

C3rBytes

PA_5, Projektarbeit, INF-P-AT005, BE1

Dokumentation Gruppe 4:



Jérémie Equey, Olaf Schmidt, Mersid Hazbiu

Bachelor of Science in Informatik 2017.BE1

Herbstsemester 2020/21

Abgabe am 23.12.2020



Dokumentenversionierung

Version	Datum	Änderungen	Autor
0.1	02.09.2020	Initiale Version Format Aufbereitung	O. Schmidt
0.2	20.09.2020	Neue Version mit 2. Fassung des Projektauftrages	J. Equey
0.3	30.10.2020	Komplettiert bis zur Konzeption	Alle
0.4	08.11.2020	Anpassungen an Konzeption	O.Schmidt, J. Equey
0.5	15.12.2020	Testing	J. Equey
1.0	21.12.2020	Finale Fassung	Alle

Inhaltsverzeichnis

1. PROJEKTMANAGEMENT	8
1.1. PROJEKTAUFRAG	8
1.1.1. Ausgangslage	8
1.1.2. Projektziele und Inhalte	8
1.1.3. Rahmenbedingungen / Restriktionen / Technische Ressourcen	10
1.1.4. Eckwerte	10
1.1.5. Grundlagen	12
1.1.6. Begründeter Lösungsansatz	12
1.1.7. Risiken	16
1.1.8. Problembeschreibungen	17
1.1.9. Risikomatrix	17
1.1.10. Problemlösung inkl. Wertung	18
1.1.11. Lösungsentscheid	19
1.1.12. Probleme bzw. Risikofälle	20
1.1.13. Messbare Lieferobjekte	20
1.2. PROJEKTPLANUNG	21
1.2.1. Projektstrukturplan	21
1.2.1. Terminplanung	23
1.2.2. Ressourcenplanung	24
1.2.3. Arbeitspakete	25
2. SOFTWARE-ENGINEERING	46
2.1. PRODUKTEINSATZ	46
2.1.1. Anwendungsbereiche	46
2.1.2. Zielgruppen	46
2.1.3. Betriebsbedingungen	46
2.2. STAKEHOLDERS	47
2.3. PRODUKTANFORDERUNGEN	48
2.3.1. Funktionale Anforderungen	48
2.3.2. Nicht funktionale Anforderungen	48
2.3.3. Qualitätsanforderungen	48
2.4. QUALITÄTSZIELBESTIMMUNG	49
2.5. USE CASES	50
2.5.1. Set master password	51
2.5.2. Login	52
2.5.3. add new items	54
2.5.4. Modify item	55
2.5.5. Change master password	57
2.5.6. Delete user account	58
2.5.7. Search item	60
2.5.8. Open URL	61
2.5.9. Generate password	62
2.5.10. Copy password	63
2.5.11. Delete item	64
2.5.12. Login with 2 Step Authentication	65
2.5.13. Logout	66
2.5.14. Set Master Passphrase	67
2.5.15. Change Master Passphrase	68
2.6. POC's	69

2.7.	NUTZWERTANALYSE KOMPONENTEN	69
2.7.1.	<i>Verschlüsselungsalgorithmen.....</i>	69
2.7.2.	<i>Datenbanksystem-Vergleich für unsere Applikation.....</i>	72
2.7.3.	<i>Two-Factor / Two-Step Authentication.....</i>	75
2.7.4.	<i>GUI.....</i>	78
2.7.5.	<i>Morphologischer Kasten</i>	80
2.8.	TECHNISCHE KONZEPTION	81
2.8.1.	<i>Systemidee</i>	81
2.8.2.	<i>Coderichtlinien.....</i>	81
2.8.3.	<i>Allgemeine Richtlinien.....</i>	81
2.8.4.	<i>Packages</i>	81
2.8.5.	<i>Klassen / Interfaces.....</i>	81
2.8.6.	<i>Methoden.....</i>	81
2.8.7.	<i>Variablen.....</i>	82
2.9.	ARCHITEKTURPATTERN	82
2.10.	KONTEXTSICHT	83
2.11.	DOMÄHENMODEL	84
2.12.	KOMPONENTENSICHT.....	86
2.13.	SCHNITTSTELLENBESCHREIBUNG	87
2.13.1.	<i>Schnittstelle Datenbankeintrag</i>	87
2.14.	KLASSENMODEL	87
2.14.1.	<i>Gesamtstruktur</i>	88
2.14.2.	<i>Packages</i>	89
2.14.3.	<i>Main.....</i>	89
2.14.1.	<i>Controller</i>	90
2.14.2.	<i>Crypto</i>	91
2.14.3.	<i>Utils.....</i>	92
2.14.4.	<i>DAO.....</i>	93
2.14.5.	<i>DatabaseEntry</i>	95
2.14.6.	<i>Connection</i>	96
2.15.	AKTIVITÄTSIDIAGRAMM	97
2.15.1.	<i>Zugriffssicherheit</i>	97
2.15.2.	<i>Registrierung</i>	98
2.15.3.	<i>Login</i>	100
2.15.4.	<i>Neues Profil anlegen</i>	101
2.15.5.	<i>Profil löschen</i>	102
2.16.	SEQUENZIDIAGRAMME	102
2.16.1.	<i>Registrierung</i>	103
2.16.2.	<i>Login</i>	105
2.16.3.	<i>Eintrag erstellen</i>	107
2.16.4.	<i>Eintrag löschen</i>	108
2.16.5.	<i>Eintrag ändern</i>	109
2.16.6.	<i>Master-Passwort ändern</i>	110
2.16.7.	<i>Master-Passphrase ändern</i>	111
2.17.	VERTEILUNGSSICHT	112
2.18.	DATENBANKARCHITEKTUR	112
2.18.1.	<i>Entity–relationship Diagramm</i>	112
2.18.2.	<i>Was ist Derby?.....</i>	114
2.18.3.	<i>Wie ist Derby in C3rBytes konfiguriert?</i>	115
2.18.4.	<i>Ablauf beim Erstellen eines neuen Kontos in C3rBytes :.....</i>	116
2.19.	BENUTZEROBERFLÄCHE	118

2.19.1.	<i>Login View</i>	118
2.19.2.	<i>Main View</i>	119
2.19.3.	<i>Add New Item View</i>	120
2.19.4.	<i>Generate Password</i>	121
2.20.	QUALITÄTSSICHERUNG / TESTS	122
2.20.1.	<i>Manuelle Testszenarien</i>	122
2.20.2.	<i>Testfälle / Testergebnisse</i>	123
2.20.3.	<i>JUnit Tests</i>	133
2.20.4.	<i>Testergebnisse</i>	136
2.21.	INSTALLATIONSANLEITUNG	137
2.21.1.	<i>Installation via GitLab</i>	137
2.21.2.	<i>Erstellung der Jar-Datei</i>	141
2.21.3.	<i>Wrapping die jar für Windows (.exe)</i>	142
2.21.4.	<i>Wrapping die jar für macOS (.app)</i>	143
2.21.5.	<i>C3rBytes starten (als .app- oder .exe-Datei)</i>	145
2.22.	BETRIEBSANLEITUNG	147
2.22.1.	<i>Einleitung</i>	147
2.22.2.	<i>Kapitel 1: Start</i>	147
2.22.3.	<i>Kapitel 2: Hauptmenü</i>	149
2.22.4.	<i>Kapitel 3: Profil</i>	150
2.22.5.	<i>Kapitel 4: Kontoeinstellungen ändern</i>	155
2.23.	EVALUIERUNGSBERICHT	157
2.24.	VERWENDETE LITERATUR	158
3.	EINVERSTÄNDNISERKLÄRUNG / SELBSTSTÄNDIGKEITSERKLÄRUNG	159
4.	ABBILDUNGSVERZEICHNIS	160
5.	TABELLENVERZEICHNIS	163
6.	ANHANG	164
6.1.	STATUSBERICHTE	164
6.1.1.	<i>Statusbericht 1</i>	164
6.1.2.	<i>Statusbericht 1 – Reflexion</i>	170
6.1.3.	<i>Statusbericht 2</i>	173
6.1.4.	<i>Statusbericht 2 – Reflexion</i>	180

Glossar

Nr.	Begriff	Bedeutung
1	Account	Ist gleichbedeutend wie Profil.
2	AES	Advanced Encryption Service. Aktuell sicherer symmetrischer Verschlüsselungsalgorithmus.
3	Anwender	Anwender ist gleichbedeutend wie Nutzer.
4	Applikation / App	Mit Applikation ist das Programm namens C3rBytes gemeint.
5	Brute-Force Angriffe	Ein Angreifer versucht durch Probieren aller Möglichkeiten an das gesuchte Passwort zu gelangen.
6	Credentials	Darunter sind die Login-Daten wie Benutzername und Passwort von Profilen zu verstehen.
7	DAO	Data Access Object: Das DAO-Modell (Data Access Object) ist ein strukturelles Modell, das die Anwendungs-/Geschäftsschicht von der Persistenzschicht isoliert.
8	DB / RDB	Die relationale Datenbank.
9	Event	Eine Aktion des Benutzers mit der GUI löst einen Event aus, welcher genutzt werden kann, um auf Benutzereingaben zu reagieren.
10	GUI	Graphical User Interface. Darunter ist eine grafische Benutzerschnittstelle zu verstehen.
11	Hostsystem	Das System, an dem der USB-Stick mit der Applikation eingebunden wird.
12	iv	Der Initialisierungsvektor ist Teil im AES-Verschlüsselungsalgorithmus.
13	JAVA-VM / JVM	Die virtuelle Maschine von JAVA.
14	JDK	Das Java Development Kit (JDK) ist eine Implementierung einer der beiden Java-Plattformen, die von der Oracle Corporation in Form eines Binärprodukts veröffentlicht wird und sich an Java-Entwickler auf Solaris, Linux, macOS oder Windows richtet. Das JDK enthält eine private JVM und einige andere Ressourcen, um die Entwicklung einer Java-Anwendung abzuschließen. (wikipedia.org)
15	Konto	Die Nutzerinformationen, die mit der Datenbank verbunden sind
16	Laufwerk mounten	Das Anschließen eines Speichermediums zum Zugriff auf dessen Daten
17	Login-Account	Account auf einer anderen Plattform
18	Maintained	Software, die vom Hersteller gewartet und aktualisiert wird.
19	Masterpassphrase	Der Schlüssel, welcher das Passwort zur Verschlüsselung und Entschlüsselung von Profil-Passwörtern verschlüsselt und entschlüsselt
20	Masterpassword	Das Passwort, welches die Datenbank als solche verschlüsselt
21	MPP	Abkürzung von Master Passphrase.
22	MPW	Abkürzung von Master Passwort.
23	Nutzer	Endbenutzer des Systems.
24	Profil	Darunter sind die verschiedenen Informationen zu verstehen, die mit einem Login-Account verbunden sind.
25	salt	Zum besseren Schutz gegen brute-force Angriffe kann ein Passwort mit einem salt "gesalzen" werden. Es wird dem Passwort hinzugefügt.
26	SHA3	Aktuell sicherer Hashalgorithmus.



27	String	Eine Zeichenkette bestehend aus mehreren Zeichen.
28	System	Die Applikation namens C3rBytes.
29	User	User ist gleichbedeutend wie Nutzer.
30	Zwischenablage	Die Zwischenablage (englisch Clipboard) ist ein Puffer, also ein Zwischenspeicher, für das kurzzeitige Speichern und Abrufen von Daten.

1. Projektmanagement

1.1. Projektauftrag

1.1.1. Ausgangslage

1.1.1.1. Ist-Situation

Täglich erfahren wir von spezialisierten Websites, dass Datenlecks mit persönlichen Informationen ins Netz gestellt werden. Häufig enthalten diese Daten Benutzernamen, E-Mail-Adressen sowie Passwörter. Diese im Web verloren gegangenen Daten stellen eine Gefahr für die Besitzer dieser Passwörter dar, da ihre Informationen nicht mehr sicher sind und somit kompromittiert werden könnten.

Diese gesammelten Informationen können in einem Wörterbuch (engl. wordlist) landen und werden unter anderem zur Durchführung von Brute-Force-Angriffen verwendet.

Die Konsequenzen eines solchen Datenverlusts können für die betroffenen Benutzer katastrophal sein. Ein Hacker könnte z.B. das Passwort eines E-Mail-Kontos ändern, so dass der ursprüngliche Benutzer keinen Zugriff mehr auf seine Daten hat.

Auch wenn Unternehmen vor einigen Jahren begonnen haben, Strategien zur Sicherung von Konten (2- oder Multi-Faktor-Authentifizierung) zu implementieren, ist es immer noch wichtig, eine gute Hygiene mit sensiblen Daten zu haben. Ein Passwort sollte nur zur einmaligen Verwendung gelangen. Zudem sollte es hinreichend komplex sein. Komplexität bedeutet jedoch für einen Menschen nicht dasselbe wie für einen Rechner. *Bz54!_@* ist vielleicht für Menschen kompliziert, jedoch nicht für einen Rechner. Außerdem ist seine Entropie äußerst gering. Auf der anderen Seite ist *@/_I-am-A-PINK-Licorne-3012_#* ein Passwort mit einer höheren Entropie aber für den Benutzer schwieriger zu merken.

Daher ist die Benutzung eines Passwort-Managers heutzutage eine Pflicht, wenn man sein digitales Leben schützen und nicht riskieren will, alle seine Daten durch das Wiederverwenden alter Passwörter zu verlieren.

1.1.1.2. Definition Passwortmanager

Was versteht man unter einem Passwort-Manager?

Ein Passwort-Manager ist eine Art Software, die es einem Benutzer ermöglicht, seine Passwörter zu verwalten, entweder durch Zentralisierung aller seiner Identifikationen und Passwörter in einer Datenbank (Portfolio) oder durch deren Berechnung durch einen Algorithmus. **Der Passwort-Manager ist mindestens durch ein eindeutiges Passwort geschützt**, so dass man sich nicht alle anderen Passwörter merken muss.

Der Benutzer kann daher auch kompliziertere (und robustere) wählen und für jedes Konto oder jedes Dokument ein anderes Passwort einsetzen. (Sollte eines der Passwörter abgefangen worden sein, sind die anderen Konten oder Dokumente davon nicht betroffen). ([Wikipedia.org](#))

1.1.2. Projektziele und Inhalte

1.1.2.1. Projektziele

In diesem Modul wollen wir, Jérémie Equey, Olaf Schmidt und Mersid Hazbiu einen Passwort-Manager entwickeln. Der Passwort-Manager muss seine Daten in erster Linie schützen und zudem über die Grundfunktionen verfügen, die von dieser Art von Software erwartet werden (wie z.B. einen Passwortgenerator). Eine detailliertere Liste der Anforderungen ist in Kapitel 2.4 Produktanforderungen

zu finden. Diese Software wird von Grund auf entwickelt, getestet und dokumentiert. Projektende ist am 23.12.2020 vorgesehen.

In diesem Projekt wird es unser Ziel sein, die Sicherheit (durch Produktdesign) zu respektieren. Wir wollen unsere Daten nicht aufgrund eines Designfehlers exponieren. Viele Fragen sind noch offen, werden aber im Laufe dieser Arbeit geklärt werden.

Darüber hinaus möchten wir die Vorgaben aus der Arbeitsmappe weitestgehend erfüllen, um den Lerneffekt dieses interdisziplinären Moduls vollumfänglich umsetzen zu können.

1.1.2.2. Name des Produktes

Unser Passwort-Manager heisst C3rBytes. Es handelt sich um eine Verkettung von Cerber und Bytes. Das Resultat (cerbytes) wird nach den Regeln von Leet Speak (L33t Speak) transformiert, d.h. die Vokale werden durch Zahlen ersetzt. Wir heben auch die Konsonanten C und B hervor, um die Grenze zwischen den beiden Wörtern zu markieren: C3rBytes.

In der griechischen Mythologie ist Cerberus (im Altgriechischen Κέρβερος / Kérberos) der dreiköpfige Hund mit Schlangenschwanz, der den Eingang zur Hölle bewacht und verhindert, dass die Toten aus der Höle des Hades entkommen und die Lebenden kommen, um einige der Toten zu holen. Eine schöne Metapher für die Sicherheit unseres Produkts. ([Wikipedia.org](#))

Das Byte ist eine Einheit digitaler Information, die aus acht Bits besteht. Dies ist ein Verweis auf die Funktionen unserer Software, die digitale Informationen transformiert und dadurch absichert.

1.1.2.3. Minimum viable Product

Keep it simple but make it great, wird unsere Maxime sein. Wir wollen keinen Passwort-Manager wie eine Gasfabrik entwickeln. Ein guter Passwort-Manager weiß, wie er seine Datenbank schützen kann (kein Zugriff ohne Eingabe der korrekten Zugangsdaten) und wie er dem Benutzer komplexe Passwörter vorschlagen kann. Das war's. Der Rest ist Beilage.

Insbesondere sollen folgende Ziele realisiert werden:

- Intuitive Bedienung des Systems.
- Entwicklung eines Passwort-Generators.
- Sichern der Daten mit einem Master-Passwort.
- Zusätzliche Sicherheit durch Eingabe einer Master-Passphrase.
- Fähigkeit, diese Zugangsdaten zu ändern.

Kurzum wollen wir eine sichere App für Profil-Einträge entwickeln.

1.1.2.4. Grobanforderungen

- Das System hat eine Schnittstelle (GUI). Die Software kann mit der Maus gesteuert werden.
- Das System ist durch ein Master-Passwort geschützt.
- Passwörter werden im Passwort-Manager gespeichert und geschützt (kein Zugang ohne das Master-Passwort und/oder Master-Passphrase).
- Das System kann starke Passwörter erzeugen (Passwort-Generator-Funktion).
- Das System kann vom Passwort-Manager aus Weblinks starten bzw öffnen.
- Das Zugangsdaten können geändert werden.
- Das System bietet keine Password share Funktion (ausser copy-paste).

1.1.2.5. Abgrenzung

- Das System ist nicht Multiuser-fähig.

- Das System kann nur auf einem «normalen» Computer genutzt werden (keine IOS oder Android Devices).
- Das System muss nicht auf Deutsch zu Verfügung stehen.

1.1.3. Rahmenbedingungen / Restriktionen / Technische Ressourcen

1.1.3.1. Technische Vorgaben (FFHS)

Als Programmiersprache ist hauptsächlich Java zu verwenden. Die Sprachmittel der objektorientierten Programmierung sind zu beachten. Die Nutzung von weiteren Sprachen ist möglich, für unser Projekt wird SQL verwendet, sofern wir eine Datenbank implementieren werden.

1.1.3.2. Technische Ressourcen

Für den Betrieb der Software ist ein Rechner mit einem modernen Betriebssystem wie Windows 10, Mac OS 10.15 oder Linux 5.7.6 erforderlich.

Die Version der Java Runtime Environment (JRE) muss mindestens in der Version 8 und das Java Developper Kit (JDK) mindestens in der Version 11.0.1 vorliegen (empfohlenen: Version 15.0), damit unsere Anwendung optimal funktioniert.

1.1.3.3. Restriktionen

Wir garantieren nicht, dass die Software mit einer früheren Version der genannten Betriebssysteme oder des Java Runtime Environments funktioniert.

1.1.4. Eckwerte

1.1.4.1. Projektdauer

Das Projekt startet offiziell am 21.08.2020 aufgrund der aktuellen Pandemielage (Corona) mit einem Online-Kick-Off und soll an die für unsere Gruppe verantwortlichen Dozenten gesendet werden. Daher wird unser Team in den Wochen bis zum 23.12.2020 den Passwort-Manager C3rBytes entwickeln.

1.1.4.2. Aufwand

Das Projekt muss von allen drei Teammitgliedern in einer Gesamtarbeitszeit von ca. 450 Stunden abgeschlossen werden. Dies entspricht 25 Stunden Arbeit pro Woche für die gesamte Gruppe, d.h. (ca.) 8 Stunden pro Person. Dies ist nicht zu vernachlässigen, wenn man bedenkt, dass jedes Teammitglied anderen Modulen folgt und zu 70% bis 80% beruflich tätig ist. Zwei Teammitglieder haben zudem kleine Kinder.

1.1.4.3. Phasentermine

Dieses Projekt verwendet das Phasen-Vorgehensmodell des Lehrmittels «Handbuch Projektmanagement» vom Springer Verlag 2011. Dabei werden die Phasen sinnvoll auf das Projekt angepasst.

Für die Softwareentwicklung hat sich der Projektausschuss für das Wasserfallmodell entschieden. Es ist ein lineares (nicht iteratives) Vorgehensmodell, dass sich für dieses Projekt mit vorgegebenen Anforderungen und Leistungen präzise beschreiben lässt.



Tabelle 1: Phasen

Nr.	Phase	Beschreibung
1	Initialisierung	Definition der Aufgaben, Ziele und der Zielgruppe. Grobanforderungen definieren, Ergebnisse formulieren und Meilensteine planen.
2	Vorstudie	Erstellung einer Grobplanung für die Umsetzung der Projektidee. Definition der Ablaufplanung, Projekt-Organisation und des Projektauftrags.
3	Konzeption	Gesamtkonzept fertigstellen, Lösungsansätze (Varianten) aufzeigen und bewerten. Detaillierte Projektlösung planen und erarbeiten.
4	Realisierung	Umsetzung des Detailplans. Einführung planen, Controlling durchführen und Abweichungen kommunizieren
5	Abschluss	Übergabe organisieren, Abschlussbericht und Schlussrechnung erstellen. Projektdokumentation ergänzen, Projektbeurteilung und «Lessons Learned» verarbeiten. Evaluierungsbericht erstellen.

Tabelle 2: Phasenbeschreibungen

MS	Name	Start	Ende
1	Initialisierung	01.09.2020	10.09.2020
2	Vorstudie	11.09.2020	18.09.2020
3	Konzeption	27.09.2020	26.10.2020
4	Realisierung	27.10.2020	13.12.2020
5	Projektabchluss	14.12.2020	23.12.2020

Tabelle 3: Meilensteine

1.1.4.4. Rollenverteilung

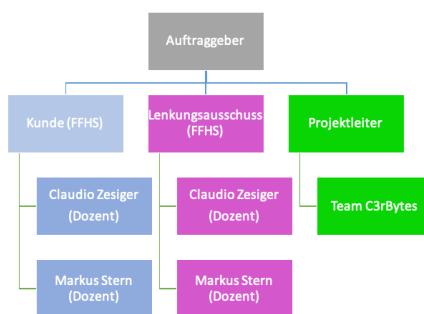


Abbildung 1: Rollenorganisation

In unserem Projekt sehen wir die folgenden Funktionen vor: Projektmanager, Entwickler, Architekt, Entwickler/Architekt, Requirement Engineer und Tester. Die erste Rollenverteilung fand am 21.08.2020 statt.

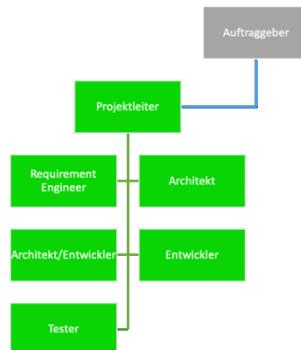


Abbildung 2: Team C3rBytes

Jedes Teammitglied übernimmt mindestens einmal die Rolle des Projektleiters. Die Zuteilungen der Rollen sind als Richtlinie zu verstehen und können je nach Stärken des Teammitgliedes variieren.

Vorname	Name	Rolle	bis	Phase
Jérémie	Equey	Projektleiter	19.09.2020	Initialisierung, Vorstudie
Olaf	Schmidt	Requirements Engineer	19.09.2020	Initialisierung, Vorstudie
Mersid	Hazbiu	Requirements Engineer	19.09.2020	Initialisierung, Vorstudie
Jérémie	Equey	Architekt/Entwickler	20.10.2020	Vorstudie, Konzept
Olaf	Schmidt	Projektleiter	20.10.2020	Vorstudie, Konzept
Mersid	Hazbiu	Architekt/Entwickler	20.10.2020	Vorstudie, Konzept
Jérémie	Equey	Architekt/Entwickler Tester	14.12.2020	Realisierung, Abschluss
Olaf	Schmidt	Architekt/Entwickler Tester	14.12.2020	Realisierung, Abschluss
Mersid	Hazbiu	Projektleiter	14.12.2020	Realisierung, Abschluss
Jérémie	Equey	Projektleiter	23.12.2020	Realisierung, Abschluss
Olaf	Schmidt	Entwickler	23.12.2020	Realisierung, Abschluss
Mersid	Hazbiu	Entwickler	23.12.2020	Realisierung, Abschluss

Tabelle 4: Rollenverteilung

1.1.5. Grundlagen

Es gibt keine bereits geleistete Arbeit, auf die wir aufbauend oder wiederholend zurückgreifen könnten. Wir erstellen eine Applikation von Grund auf, von A bis Z. Die drei Mitglieder des Teams haben ihre Spezialisierung in Computersicherheit an der FFHS begonnen und verfügen bereits über theoretische Kenntnisse in Sicherheit und Kryptographie.

1.1.6. Begründeter Lösungsansatz

1.1.6.1. Musskriterien

Gemäss unseren Grobanforderungen haben wir die folgenden Musskriterien definiert, die für die Auswahl der drei zu vergleichenden Varianten nötig sind.

Nr.	Beschreibung
MZ01	Das System muss eine grafische Nutzeroberfläche (GUI) bereitstellen.
MZ02	Das System muss dem Nutzer erlauben, sich nach der initialen Anmeldung mittels seinen Master-Zugangsdaten einzuloggen.
MZ03	Das System muss dem Nutzer erlauben sein Master-Zugangsdaten zu ändern.
MZ04	Das System muss dem Nutzer erlauben Daten eines Profils hinzuzufügen.
MZ05	Das System muss dem Nutzer erlauben Daten eines Profils zu ändern.
MZ06	Das System muss dem Nutzer erlauben Daten eines Profils zu löschen.
MZ07	Das System muss dem Nutzer erlauben, mittels Suchfunktion seine Profile zu durchsuchen.
MZ08	Das System muss die Datenbank mit den Master-Zugangsdaten schützen.
MZ09	Das System muss die Passwörter der Profile mit einem unbekannten Passwort sichern.
MZ10	Das System muss dem Nutzer erlauben Weblinks mit dem Default-Browser zu öffnen.
MZ11	Das System muss dem Nutzer die Funktionalität bieten starke Passwörter nach seinen Vorgaben zu generieren.
MZ12	Das System muss dem Nutzer ermöglichen, das Passwort zu kopieren und es nach 10 Sekunden aus der Zwischenablage zu löschen.
MZ13	Das System muss das kopierte Passwort nach 10 Sekunden automatisch aus der Zwischenablage entfernen
MZ14	Das System muss dem Nutzer die Löschung seines Master-Accounts ermöglichen.
MZ15	Das System muss es dem Nutzer ermöglichen sich ordnungsgemäss auszuloggen.

Vordergründig müssen wir uns über die Sicherheit des Entwurfs erkundigen, damit wir bei der Verwendung der Software keine entscheidenden Informationen preisgeben.

Um einen Passwort-Manager zu entwickeln, bieten sich drei Varianten an. Diese nachfolgenden Varianten können allesamt unsere Musskriterien vollumfänglich abdecken.

1. Eine Software (Standalone App): Der Benutzer kann auf seine Daten zugreifen, indem er die Software startet. Alle Daten werden auf seinem Computer gespeichert. Es erfolgt kein Austausch über ein Netzwerk.
2. Einen Webdienst (eine Webanwendung): Der Benutzer kann seine Daten mit seinem Webbrowsert konsultieren, indem er auf die Webseite der Anwendung (im Internet) zugreift.
3. Eine Softwareapplikation, die Cloud-Funktionen verwendet, um ihre Datenbank zu synchronisieren und dem Benutzer über eine Schnittstelle (Webportal) den Zugriff zum Web ermöglicht. Der Benutzer kann seine Daten entweder auf seinem Computer mit Hilfe der Software oder mit seinem Webbrowsert konsultieren.

Daher werden die Varianten anhand der Wunschkriterien in einer Nutzwertanalyse evaluiert.

1.1.6.1. Wunschkriterien

Nr.	Beschreibung
WZ01	Datensicherheit. Die Daten sind zu jeder Zeit sicher und geschützt, weshalb keine Kommunikation über ein Netzwerk erfolgen soll.
WZ02	Investition in Zeit: das Projekt ist innerhalb der geplanten Stunden (450 h) umzusetzen.
WZ03	Das Produkt verwendet den Entwicklern bekannte Technologien.
WZ04	Die Komplexität des Produkts ist hinreichend.
WZ05	Der Passwort-Manager ist Portabel (USB-Stick).

1.1.6.1. Abgrenzungskriterien

Nr.	Beschreibung
AZ01	Das System muss keine anderweitigen Exporte irgendwelcher Art ausgeben können.
AZ02	Das System muss keine Schnittstelle zu anderen Frameworks enthalten.
AZ03	Das System soll nicht mehrbenutzertauglich sein.

Wir vergleichen unsere drei Varianten nach dem Grad ihrer Erfüllung der Wunschkriterien multipliziert mit einer Gewichtung. Es wird die Variante mit der höchsten Punktzahl ausgewählt.

1.1.6.2. Nutzwertanalyse

Variante	Beschreibung
V1	Standalone Application
V2	Webapplication
V3	Mixed Application (Web und Desktop)

1.1.6.2.1. Präferenzmatrix

Punktevergabe	
0	weniger wichtig
1	gleich wichtig
2	wichtiger

		Präferenzmatrix						
Kriterium		WZ01	WZ02	WZ03	WZ04	WZ05	Summe	Faktor %
WZ01		2	2	1	1	1	6	30
WZ02		1		1	2	0	4	20
WZ03		1	0		1	1	3	15
WZ04		0	0	1		1	2	10
WZ05		1	2	1	1		5	25
Summe							12	100

Die Gewichtung ist definiert, so können wir mit der Nutzwertanalyse fortfahren und eine Variante bestimmen.

Bewertung n	
0	Erfüllt das Kriterium nicht
1	Erfüllt das Kriterium nur teilweise oder mit Workaround (z.B. Zusatztool)
2	Erfüllt das Kriterium vollständig
3	Übertrifft das Kriterium (z.B. mit weiteren Funktionalitäten als definiert)

Lösungsvariante		V1		V2		V3	
Kriterium	Gewichtung g	n	g*n	n	g*n	n	g*n
WZ01	0.30	2.000	0.600	2.000	0.600	2.000	0.600
WZ02	0.20	2.000	0.400	2.000	0.400	1.000	0.200
WZ03	0.15	3.000	0.450	1.000	0.150	1.000	0.150
WZ04	0.10	1.000	0.100	1.000	0.100	1.000	0.100
WZ05	0.25	2.000	0.500	1.000	0.250	1.000	0.250
	1.00		2.05		1.50		1.30

1.1.6.3. Lösungswahl/Entscheid

Die Variante 1 (Stand Alone Software) wurde gewählt. Die Software erfüllt nicht nur alle unsere obligatorischen Kriterien (Musskriterien), sondern entspricht auch perfekt unseren Wunschkriterien.

Eine Stand Alone-Applikation wird es uns ermöglichen, unser Wissen über die Programmiersprache Java (Refresh) in die Praxis umzusetzen, und wir sind zuversichtlich, dass diese Variante langfristig tragfähig sein wird und es uns ermöglichen wird, unsere definierten Ziele zu erreichen. In kommenden Schritten werden wir die Varianten der Komponenten unserer Stand Alone-Applikation eruieren.

1.1.7. Risiken

Die Risikoidentifizierung und Risikobeurteilung spielt im Projektmanagement eine wichtige Rolle. Das Ziel dabei ist, sich bewusst zu werden, wo die Stärken und Schwächen des Projekts, zu dem auch das Projektteam gehört, liegen.

Die Risiken sind zahlreich, aber es handelt sich nicht unbedingt um finanzielle Konsequenzen (Universitätsprojekt).

- Keiner der Teilnehmer unseres Teams ist ein professioneller Entwickler.
- Obwohl wir von der FFHS in der Sprache JAVA ausgebildet worden sind, könnten einige Programmierfehler auftreten.
- Aufgrund des Studiensystems der FFHS und der Intensität dieses Moduls parallel zu den anderen Modulen des Semesters besteht die Gefahr, dass ein oder mehrere Mitglieder das Team verlassen.
- Alles ist sicher, bis es nicht mehr sicher ist. Es besteht ein erhebliches Risiko eines schlechten Entwurfs oder einer schlechten Umsetzung unserer Lösung, die die Achillesferse unserer Anwendung sein könnte. Besondere Aufmerksamkeit muss dem Design gewidmet werden.

Im Rahmen einer Teamsitzung wurde mittels Brainstormings eine SWOT-Analyse generiert, welche wie folgt aussieht:

		Interne Analyse	
		Stärken	Schwächen
Externe Analyse	Chancen	<ul style="list-style-type: none"> • Hohe Leistungsbereitschaft • Heterogenes Team • Kompetenz • Kreativität 	<ul style="list-style-type: none"> • Koordination im Team • Unzureichende Planung • Lastverteilung im Team • Vernachlässigung der Dokumentation • Wenig Erfahrung in PM und SW-Engineering
	Risiken	<ul style="list-style-type: none"> • Kompetentes Kontrollboard • Ausgereifte App • Potenzielle Marktfähigkeit der App 	<ul style="list-style-type: none"> • Nutzung fremder Libraries • Nutzung fremder Plattformen JAVA-VM • Aus- oder Wegfall von Teammitgliedern • Technische Probleme z.B. mit der Versionsverwaltung

Tabelle 5: SWOT-Analyse

1.1.8. Problembeschreibungen

Anhand der SWOT-Analyse wird deutlich, wie wir für das Projekt C3rBytes aufgestellt sind. Es stellt sich nun die Frage wie wir unsere Stärken ausbauen und unsere Schwächen verhindern oder wenigstens mindern können.

Nr.	Problem	Beschreibung
1	Koordination im Team	<ul style="list-style-type: none"> • Unklarheiten betreffend Aufträgen und Terminen • Passivität • Redundanzen
2	Unzureichende Planung	<ul style="list-style-type: none"> • Planung entspricht nicht den Vorgaben. • Planung wird vernachlässigt, folglich wird gebastelt.
3	Lastverteilung im Team	<ul style="list-style-type: none"> • Belastung im Team könnte einseitig verlaufen.
4	Vernachlässigung der Dokumentation	<ul style="list-style-type: none"> • Struktur und der Dokumentation entspricht nicht den Vorgaben. • Dokumentation, sei es in PMG, OOP, SWE oder DBS wird nicht konsistent geführt.
5	Wenig Erfahrung	<ul style="list-style-type: none"> • Keine oder wenig Erfahrung bei Projektmanagement und Software-Engineering
5	Nutzung fremder Libraries	<ul style="list-style-type: none"> • Durch die Nutzung von fremden Libraries entstehen Abhängigkeiten, auf welche man nicht wunschgemäß Einfluss ausüben kann.
6	Nutzung fremder Plattformen	<ul style="list-style-type: none"> • Die Java Run Time Environment ist auf dem Zielsystem nicht korrekt aufgesetzt. • Es gibt Versionskonflikte bei den JAVA Frameworks • Der JAVA-VM erleidet einen technischen Ausfall oder einen Konflikt.
7	Aus- oder Wegfall von Teammitgliedern	<ul style="list-style-type: none"> • Teammitglied(er) können temporär nicht am Projekt mitwirken oder fallen komplett aus.
8	Technische Probleme z.B. mit der Versionsverwaltung	<ul style="list-style-type: none"> • Kompatibilitätsprobleme mit der Versionsverwaltung.

Tabelle 6: Risikenbeschreibung

1.1.9. Risikomatrix

In der Risikomatrix werden die Risiken bzw. Probleme eingetragen. Dabei werden die Risiken nach den Kriterien Eintrittswahrscheinlichkeit x Schadensausmass bewertet. Zur besseren Übersicht wird jedes Problem in die Risikomatrix eingetragen. Der Risikomatrix ist visuell zu entnehmen, welche Probleme ernst genommen werden müssen und welchen man möglicherweise weniger Priorität zukommen lassen kann bzw. gänzlich ignorieren kann.

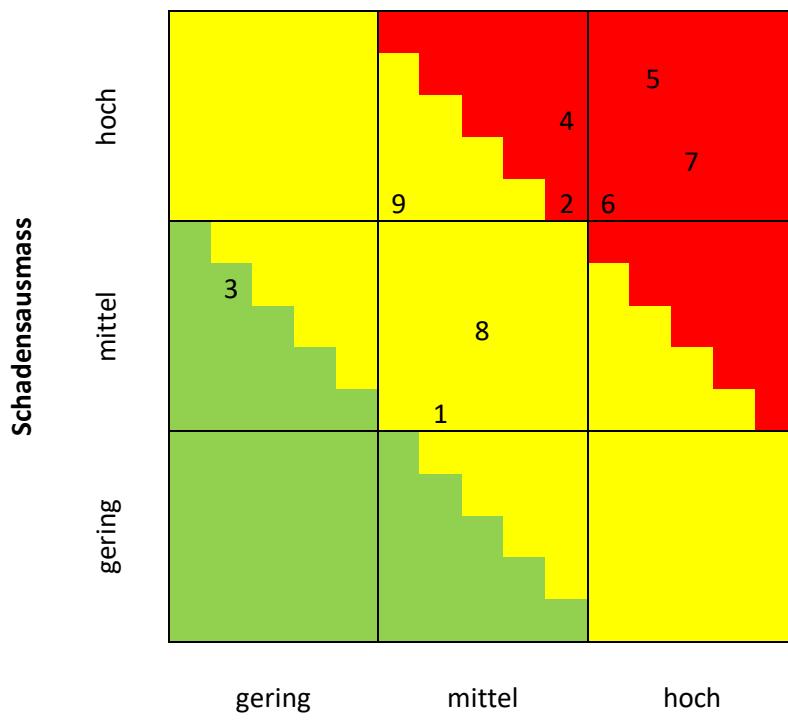


Tabelle 7: Risikomatrix

1.1.10. Problemlösung inkl. Wertung

Anhand der Risikomatrix werden die Risiken erkannt und dem entsprechend Massnahmen unter Zuhilfenahme des Problemlösungszyklus entwickelt. Als Ergebnisse der Problemlösungszyklen sind in untenstehender Tabelle zu jedem Problem Massnahmen und deren Bewertung aufgelistet.

Wertungsraster der Problemlösungen von 1 (Zu priorisieren) bis 3 (letzte Wahl)

Nr.	Problem	Lösungen	Wertung
1	Koordination im Team	<ul style="list-style-type: none"> • Klare Aufträge Zuständigkeiten definieren • Aufträge terminieren • Wöchentliche Sitzungen mit vorher mitgeteilten Traktanden • Zustellung der Sitzungsprotokolle 	1 1 2 2
2	Unzureichende Planung	<ul style="list-style-type: none"> • Genügend Zeit für die Planung vorsehen • Ist- Sollvergleich durchführen • Abweichungen abfangen • Bei Bedarf Massnahmen planen und Umsetzen 	1 2 2 1
3	Lastverteilung im Team	<ul style="list-style-type: none"> • Erfassen von IST-Belastungen • Vergleich von Ist- mit Soll-Belastung • Resultate des Ist- Sollvergleichs auswerten 	1 1 1

		<ul style="list-style-type: none"> • Gespräch im Team suchen • Funktionalitäten streichen • Überstunden 	1 1 2
4	Vernachlässigung Dokumentation	<ul style="list-style-type: none"> • Kontinuierliche Dokumentation (mind. 1x wöchentlich) • Recherche in anderen Quellen zur Ergänzung und Erweiterung der Dokumentation • Reviews durch andere Teammitglieder • Feed-back des Kontrollboards ummünzen 	2 1 1 1
5	Wenig Erfahrung	<ul style="list-style-type: none"> • Erfahrene Leute zur Beratung dazuziehen • Internetrecherche • Literatur verarbeiten 	1 2 2
6	Nutzung fremder Libraries	<ul style="list-style-type: none"> • Fremde Libraries möglichst vermeiden • Eigene Libraries implementieren und verwenden 	2 2
7	Nutzung fremder Plattformen	<ul style="list-style-type: none"> • Immer mit der aktuellsten, stabilen Version der JAVA-VM arbeiten 	1
8	Aus- oder Wegfall von Teammitgliedern	<ul style="list-style-type: none"> • Änderungsantrag erstellen und einreichen 	1
9	Technische Probleme z.B. mit der Versionsverwaltung oder der Entwicklungsumgebung	<ul style="list-style-type: none"> • Neuaufsetzen des Projekts • Kompatibilitätsprobleme lösen 	1 1

Tabelle 8: Problemlösung inkl. Wertung

1.1.11. Lösungsentscheid

In einer Sitzung, an der alle Teammitglieder anwesend waren, wurde beschlossen, dass mindestens alle mit 1 gewerteten Lösungen durchgesetzt werden sollten. Lösungen mit Wertung 2 werden als optional, wobei Lösungen mit Wertung 3 vorerst vernachlässigt werden können, wobei aber immer auch situationsbezogen reagiert werden soll.

Nachkontrolle erfolgt kontinuierlich nach erfolgter Ausführung.

1.1.12. Probleme bzw. Risikofälle

Sollten sich Problem- bzw. Risikofälle entwickeln, werden wir unter diesem Abschnitt die Einzelheiten wie Problembeschreibung, betroffene Ressourcen, Massnahmen und Nachkontrolle tabellarisch aufführen.

Problem	Name	Massnahme	Nachkontrolle / Status
Eine Woche Rückstand	Alle	Wunschkriterien wurden weggelassen.	erfüllt
Unfall (Schulterbruch, 2 Wochen)	Olaf Schmidt	Aufgaben wie Dokumentation schreiben und programmieren wurden verschoben, andere Aufgaben, die mit einer Hand lösbar waren, wie Diagramme zeichnen, wurden an ihn übergeben.	erfüllt
Krankheitsfall (Seit November)	Mersid Hazbiu	Aufgaben wurden im Krankheitsfall an andere Teammitglieder übergeben, sodass Tage der Arbeitsunfähigkeit möglichst geringe Auswirkung auf das Projekt hatten.	erfüllt
Ungleiche Lastverteilung im Team	Alle	Überstunden, da ansonsten kein auslieferbares Produkt hätte abgeliefert werden können.	erfüllt

Tabelle 9: Probleme und Massnahmen

1.1.13. Messbare Lieferobjekte

Im Rahmen dieses Projekts planen wir die wichtigsten Lieferobjekte. Diese und weitere werden später bei der Projektplanung näher erläutert:

- Projektauftrag
- Projektstrukturplan
- Statusberichte 1 und 2
- Präsentationen der Statusberichte 1 und 2
- Fachanforderungen
- Nichtfunktionale Anforderungen
- Funktionale Anforderungen
- Technische Konzeption
- Testkonzept
- Quellcode
- Unit-Test
- Testprotokolle
- Java Dokumentation
- Produkt auf USB-Stick
- Installationsanleitungen
- Benutzerhandbuch
- Projektdokumentation

1.2. Projektplanung

1.2.1. Projektstrukturplan

Das Phasenmodell gibt vor wie das Projekt und dessen Lieferobjekte aufgebaut sind. Der Projektstrukturplan ist in tabellarischer Form aufgeführt. Er gibt Aufschluss über die Gliederung des Projektes in die Phasen, sowie den zugehörigen und wichtigsten Arbeitspaketen. Ein detaillierter Projektstrukturplan, bei dem jedes Paket einzeln aufgeführt hätte, hätte den Rahmen der Dokumentation gesprengt.

Initialisierung	Vorstudie	Konzeption	Realisierung	Einführung
Projektziele				
Nutzwertanalyse Projekt				
Grobanforderungen				
Meilensteine planen				
1. Rollenverteilung				
Teamsitzungen				
	Projektantrag			
	Planung (Initial)			
	Risiken Identifizieren und bewerten			
	Pflichtenheft erstellen			
	Proof of Concept			
	Projektstrukturplan			
	Statusbericht 1			
	Präsentation			
	Reflektion			
	Teamsitzungen			
	2. Rollenverteilung			
		Use Cases erstellen		
		Nutzwertanalysen Verschlüsselung		
		Nutzwertanalyse Datenbank		
		Nutzwertanalyse GUI		
		Nutzwertanalyse Two-Step Authentication		
		PoC Apache Derby		
		UI-Konzept		
		Qualitätskriterien definieren		
		Loginalgorithmus definieren/ Aktivitätsdiagramm		
		Domänenmodell/ Fachklassenmodell erstellen		
		Kontextdiagramm erstellen		
		Klassendiagramme erstellen		
		Sequenzdiagramm erstellen		

		Verteilungsdiagramm erstellen		
		EER Diagramm		
		Testkonzept erstellen		
		Teamsitzungen		
		Implementierung Crypto-Package		
		Implementierung Utils-Package		
		Graphical User Interface (GUI)		
		Implementierung Controller		
		Implementierung Datenbank / Datenbankanbindung		
		Komponentenintegration		
		Unit-Tests		
		Manuelle Tests / Testprotokolle		
		JavaDoc erstellen		
		Refactoring / Bug fixing		
		Packaging (Jar-Datei)		
		Installationsanleitung		
			Benutzerhandbuch	
			Projektdokumentation	
			Abschlussbericht	
			Fertigstellung Dokumente	
			Projektdruck und Rollout	
			Projektversand	
			Verteidigung vorbereiten	
			Präsentation	
			Nachbesprechung Projektarbeit	

Tabelle 10: Projektstrukturplan

1.2.1. Terminplanung

Der Terminplan wird mit dem Gratis-Programm "GanttProject" erstellt. Das Gantt-Diagramm stellt den kritischen Pfad schraffiert dar und wurde mit ursprünglich geplanten Terminen erstellt. Die grünen Balken stellen die laufenden Teamsitzungen dar.

Wir gingen davon aus, dass jede Woche acht Stunden pro Person für die verschiedenen Arbeitspakete investiert werden konnten. Dies ergab das folgende Gantt-Diagramm, welches den Optimalfall für das Projekt dargestellt hätte. Aufgrund von nicht termingerecht abgeschlossenen Arbeitspaketen hat sich die Planung im September um eine Woche verzögert, was Auswirkungen auf den Rest der Planung hatte.

Aufgrund des limitierten Zeitrahmens für das Projekt wurden mehrere Arbeitspakete, vor allem in der Vorstudie und Konzeption, parallel zueinander ausgeführt, um mehr Zeit für die Realisierung zu lassen. Dies führte auch, dass Arbeitspakete des Abschlusses vorgezogen wurden. Vor allem das Bugfixen überschritt die geplanten Arbeitsdaten.

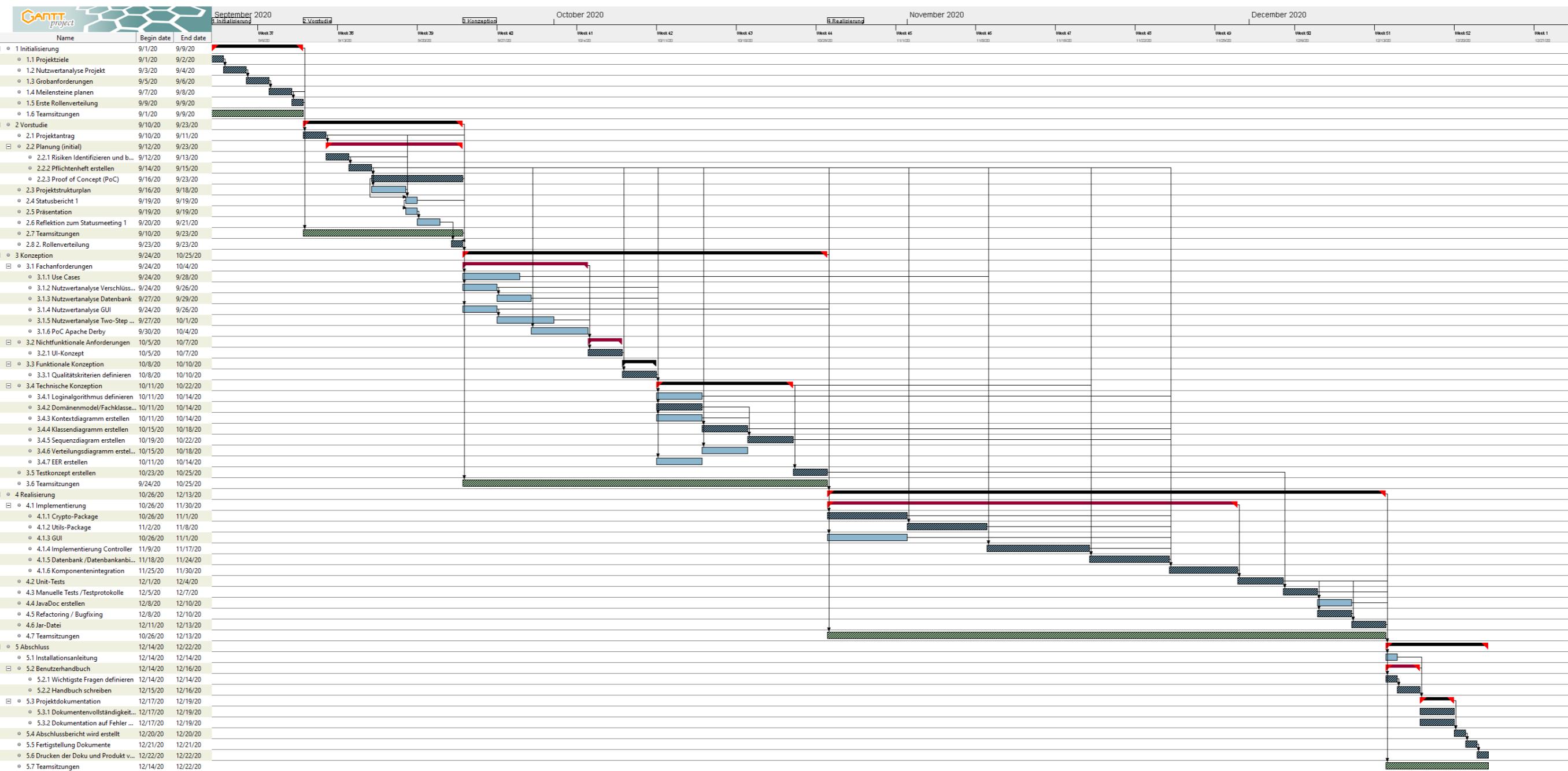


Abbildung 3: Terminplanung

1.2.2. Ressourcenplanung

Die Ressourcenplanung dient der Zuteilung von Ressourcen zu den Arbeitspaketen. In diesem Fall handelt es sich ausschliesslich um menschliche Ressourcen. Jedes Team-Mitglied hat seinen eigenen Rechner. Das Repository läuft über die von der FFHS bereitgestellte Plattform GitLab.

In der nachfolgenden Tabelle sind die Arbeitspakete inkl. den geplanten Stunden (Soll-Aufwand) für jedes Paket, sowie die Aufteilung auf jedes Teammitglied. Weiterhin sind in dieser Tabelle der aktuelle Aufwand (IST-Aufwand) in Stunden erfasst sowie die Gesamtabweichung pro Paket. Der Ressourcenplant hat sich im Verlaufe des Projektes sehr gewandelt. Aufgrund der Aufstellung erkennt man, wie schwierig die Planung war. Man erkennt, dass wir den Aufwand bei den Phasen Konzeption, Realisierung und Abschluss unterschätzt haben. Obwohl in den Phasen Initialisierung und Vorstudie «etwas Zeit gewonnen werden konnte», sind über das gesamte Projekt deutlich mehr Stunden als geplant angefallen. Hätten wir diese nicht geleistet, hätten wir keine anständig funktionierende Software ausliefern können. Es soll auch das Engagement des Teams widerspiegeln.

C3rBytes												
Projektstitel		C3rBytes	Aktualisiert am		22.12.2020							
Projektstart		19.08.2020	Version		5.2							
Projektleiter		Jérémie, Olaf, Mersid										
Nr.	Arbeitspaket	SOLL-Aufwand in Personenstunden	Soll-Aufwand Jérémie	IST-Aufwand Jérémie	Soll-Aufwand Olaf	IST-Aufwand Olaf	Soll-Aufwand Mersid	IST-Aufwand Mersid	SOLL-Aufwand in Personenstunden	Abweichung	Termine Meilensteine / Bemerkungen	Massnahmen
1	Initialisierung	24,00	18,00	16,25	3,00	5,00	3,00	5,00	26,25	2,25		
1.1	Projektziele	4,00	4,00	4,00					4,00	0,00	01.09.2020	
1.2	Nutzwertanalyse Projekt	4,00	4,00	3,00					3,00	-1,00		
1.3	Grobanforderungen	4,00	4,00	3,00					3,00	-1,00		
1.4	Meilensteine planen	2,00	2,00	1,00					1,00	-1,00		
1.5	1. Rollenverteilung	1,00	1,00	0,25					0,25	-0,75		
1.6	Teamsitzungen	9,00	3,00	5,00	3,00	5,00	3,00	5,00	15,00	6,00		
2	Vorstudie	82,00	20,00	23,00	41,00	45,00	21,00	30,00	98,00	16,00		
2.1	Projektantrag	2,00	2,00	4,00					4,00	2,00	Verlängert auf 26.09.2020	
2.2	Planung (Initial)	37,00	1,00	1,00					47,00	10,00		
2.2.1	Risiken identifizieren und bewerten	8,00			8,00	8,00			8,00	0,00		
2.2.2	Pflichtenheft erstellen	8,00			8,00	8,00			8,00	0,00		
2.2.3	Proof of Concept (POC)	20,00			20,00	24,00			6,00	30,00	Technisch schwieriger umzusetzen als Erwartet	
2.3	Projektstruktur-/Termin-/Ressourcen	16,00					16,00	20,00	20,00	4,00	Es wurden Fehler gefunden und korrigiert und Anpassungen nach Änderungsanträgen gemacht.	Mersid korrigierte in die Pläne während er Überstunden leistete.
2.4	Statusbericht 1	5,00	5,00	6,00					6,00	1,00		
2.5	Präsentation	2,00	2,00	2,00					2,00	0,00		
2.6	Reflektion	4,00	4,00	4,00					4,00	0,00		
2.7	Teamsitzungen	15,00	5,00	5,00	5,00	5,00	5,00	4,00	14,00	-1,00		
2.8	2. Rollenverteilung	1,00	1,00	1,00					1,00	0,00	26.09.2020	
3	Konzeption	115,00	45,00	61,50	54,00	56,00	16,00	18,00	135,50	17,50		
3.1	Fachanforderungen	26,00							33,00	7,00		
3.1.1	Use Cases	10,00	10,00	16,00					16,00	6,00	Viele Detailarbeiten.	
3.1.2	Nutzwertanalyse Verschlüsselungsalgorithme	4,00	4,00	4,00					4,00	0,00		
3.1.3	Nutzwertanalyse Datenbank	4,00	4,00	4,00					4,00	0,00		
3.1.4	Nutzwertanalyse GUI	4,00							4,00	0,00		
3.1.5	Nutzwertanalyse Two-Step Authentication	4,00							4,00	5,00	1,00	
3.1.6	Proof of Concept Apache Derby	12,00	12,00	15,00					15,00	3,00	Learning by doing.	
3.2	Nichtfunktionale Anforderungen	4,00							6,00	2,00		
3.2.1	UI-Konzept	4,00	4,00	4,00					2,00	6,00		
3.3	Funktionale Konzeption	2,00							2,00	0,00		
3.3.1	Qualitätskriterien definieren	2,00			2,00	2,00			2,00	0,00		
3.4	Technische Konzeption	39,00							48,50	9,50		
3.4.1	Logialgorithmus definieren / Aktivitätsdiagramm	8,00			8,00	8,00			8,00	0,00		
3.4.2	Domenenmodell / Fachklassenmodell erstellen	4,00			4,00	4,00			4,00	0,00		
3.4.3	Kontextdiagramme erstellen	2,00			2,00	2,00			2,00	0,00		
3.4.4	Klassendiagramm erstellen	8,00			3,00	8,00	8,00		11,00	3,00		
3.4.5	Sequenzdiagramme erstellen	12,00				12,00	14,00		14,00	2,00		
3.4.6	Verteilungsdiagramm erstellen	2,00				2,00	2,00		2,00	0,00		
3.4.7	EER erstellen	3,00	3,00	7,50					7,50	4,50	Es war nicht möglich, unser Schema aus MySQL Workbench in Derby zu importieren. Suche nach Lösungen.	
3.5	Testkonzept erstellen	8,00							8,00	0,00		
3.6	Teamsitzungen	24,00	8,00	8,00	8,00	8,00	8,00	7,00	23,00	-1,00		

Abbildung 4: Ressourcenplanung

4	Realisierung	235,00	94,50	137,00	82,50	108,50	58,00	56,00	301,50	66,50		
4.1	Implementierung	158,00							181,00	23,00		
4.1.1	Crypto-Package	25,00			25,00	25,00			25,00	0,00		
4.1.2	Utils-Package	10,00		1,00	8,00	8,00	2,00	2,00	11,00	1,00		
4.1.3	GUI	32,00				4,00	32,00	32,00	36,00	4,00	Neue Technologie (Mehraufwand)	Anpassungen wurden mithilfe von Überstunden erledigt während der Code geschrieben wurde.
4.1.4	Implementierung Controller	46,00	16,00	20,00	20,00	20,00	10,00	8,00	48,00	2,00		
4.1.5	Datenbank / Datenbankanbindung	30,00	30,00	50,00		5,00			55,00	25,00	Viel Troubleshooting und Bugs der DB selber.	Olaf half mit, während Jérémie Überstunden leistete um die Datenbank zum laufen zu bringen.
4.1.6	Komponentenintegration	15,00	7,50			7,50	6,00		6,00	-9,00		
4.2	Unit-Tests	14,00	7,00	7,00	7,00	8,00			15,00	1,00		
4.3	Manuelle Tests / Testprotokolle	9,00	9,00	9,00		1,00			10,00	1,00		
4.4	JavaDoc erstellen	8,00	4,00	4,00	4,00	3,50			7,50	-0,50		
4.5	Refactoring / Bug Fixing	18,00	10,00	30,00		18,00	8,00	8,00	56,00	38,00	Viel mehr Bugs als erwartet. Nach Bug-fixing traten wieder neue Bugs auf	
4.6	Jar-Datei	10,00	5,00	10,00	5,00	4,00			14,00	4,00		
4.7	Teamsitzungen	18,00	6	6,00	6,00	6,00	6,00	6,00	18,00	0,00		
5	Abschluss	49,00	8,00	14,00	4,00	15,50	37,00	44,00	73,50	24,50		
5.1	Installationsanleitung	2,00	2,00	3,00					3,00	1,00		
5.2	Benutzerhandbuch	5,00							6,50	1,50		
5.2.1	Wichtigste Fragen Definieren	1,00							1,00	1,00	0,00	
5.2.2	Handbuch schreiben	4,00							4,00	5,00	5,50	1,50
5.3	Projektdokumentation	24,00							40,50	16,50		
5.3.1	Dokumentationsvollständigkeit überprüfen	12,00		4,00		4,00	12,00	14,00	22,00	10,00	Elemente im Dokument fehlten und Dokumentenstruktur wurde angepasst.	Arbeit wurde an den Wochenenden verrichtet um die Termine einzuhalten.
5.3.2	Dokumentation nach Fehlern überprüfen	12,00				5,50	12,00	13,00	18,50	6,50		
5.4	Abschlussbericht wird erstellt	3,00	1,00	1,00	1,00	1,50	1,00		2,50	-0,50		
5.5	Fertigstellung Dokumente	4,00				1,00	4,00	8,00	10,00	6,00		
5.6	Drucken der Doku und Produkt versar	2,00	2,00	2,00					2,00	0,00		
5.7	Teamsitzungen	9,00	3,00	3,00	3,00	3,00	3,00	3,00	9,00	0,00		
Ferien/Absenzen									-			
Jérémie Equey		185,5	251,75						251,75			
Olaf Schmidt				184,50	230,00				230,00			
Mersid Habibu						135,00	153,00		153,00			
Total									634,75			
Projektende												
SOLL									IST			
Gesamtaufwand in Personenstunden		505,00							634,75			

Abbildung 5: Ressourcenplanung

1.2.3. Arbeitspakete

In diesem Kapitel werden die Arbeitspakete im Detail phasenweise für jedes wichtigere Paket aufgeführt.

1.2.3.1. Initialisierung

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.1
Arbeitspaket Name	Projektziele
Phase	Initialisierung
Input / Voraussetzung	Kick-off
Tätigkeiten	Ideen sammeln
	Zielformulierung des Projektes
Output	Projektziele
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	02.09.2020
Effektive Terminierung	02.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.2
Arbeitspaket Name	Nutzwertanalyse Projekt
Phase	Initialisierung
Input / Voraussetzung	1.1 Projektziele
Tätigkeiten	Identifizierung von MUSS-Kriterien (MK) Bestimmung der WUNSCH-Kriterien Identifizierung von potenziellen Kandidaten (Varianten), die die Kriterien erfüllen und die als Softwarelösung umgesetzt werden kann
Output	Wahl einer Variante, die als Software Lösung entwickelt wird.
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	Gemäss Inputs vom Statusmeeting 1
Planmässige Terminierung	04.09.2020
Effektive Terminierung	20.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.3
Arbeitspaket Name	Grobanforderungen
Phase	Initialisierung
Input / Voraussetzung	1.2 Projektziele
Tätigkeiten	Grobbestimmung der System- und Programmanforderungen
Output	Grobanforderungen
Aufwand in Std.	4
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	06.09.2020
Effektive Terminierung	06.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.4
Arbeitspaket Name	Meilensteine planen
Phase	Initialisierung
Input / Voraussetzung	1.3 Grobanforderungen
Tätigkeiten	Erste Planung der Meilensteine
Output	Grobplanung Projektablauf
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	Auf der Grundlage dieser ersten Planung wird eine detaillierte Planung in das PSP eingebaut.
Planmässige Terminierung	08.09.2020
Effektive Terminierung	08.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.5
Arbeitspaket Name	Rollenverteilung
Phase	Initialisierung
Input / Voraussetzung	1.4 Teamsitzung
Tätigkeiten	Bestimmung der neuen Rollen im Team Bestimmung des neuen Projektleiters neue Rollenverteilung wird in der Dokumentation eingefügt
Output	Protokoll der Sitzung / Dokumentation (Statusbericht 1 und 2)
Aufwand in Std.	1
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	09.09.2020
Effektive Terminierung	09.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	1.6
Arbeitspaket Name	Teamsitzung
Phase	Initialisierung
Input / Voraussetzung	Das Team wird zu einer Teamsitzung einberufen.
Tätigkeiten	Besprechung gemäss der Tagesordnung der Sitzung
Output	Protokoll der Sitzung
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	-
Effektive Terminierung	18.08.2020, 04.09.2020, 09.09.2020

1.2.3.2. Vorstudie

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.1
Arbeitspaket Name	Projektantrag
Phase	Vorstudie
Input / Voraussetzung	1.4 Meilensteine planen
Tätigkeiten	Erstellung des Projektauftrages mit: <ul style="list-style-type: none"> - Ausgangslage - Projektziele und Inhalte - Rahmenbedingungen/Restriktion/Technische Ressourcen (Gemäss Arbeitsmappe)
Output	Der Projektauftrages ist erstellt und verfügbar in einem Worddokument
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	Nach der ersten Statussitzung wurden Anpassungen vorgenommen.
Planmässige Terminierung	11.09.2020
Effektive Terminierung	15.09.2020, 28.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.2.1
Arbeitspaket Name	Risiken identifizieren und bewerten
Phase	Vorstudie
Input / Voraussetzung	2.1 Projektantrag
Tätigkeiten	Risiken identifizieren SWOT-Analyse erstellen Problembeschreibungen formulieren Risikomatrix generieren Problemlösungen definieren
Output	Risikenanalyse inkl. Massnahmenpaket
Aufwand in Std.	16
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	13.09.2020
Effektive Terminierung	13.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.2.2
Arbeitspaket Name	Pflichtenheft erstellen
Phase	Vorstudie
Input / Voraussetzung	Auftrag verabschiedet Projektauftrag
Tätigkeiten	Stakeholder identifizieren Muss-, Wunsch- und Abgrenzungskriterien eruieren Produkteinsatz klären Produktumgebung festlegen Produktfunktionen definieren Anforderungen definieren Qualitätskriterien festlegen
Output	Dokumentarischer Inhalt in Form eines Pflichtenhefts und Teil des Hauptdokumentes
Aufwand in Std.	16
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	15.09.2020
Effektive Terminierung	15.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.2.3
Arbeitspaket Name	PoC's / Machbarkeit
Phase	Vorstudie
Input / Voraussetzung	Anforderungen und Definitionen im Pflichtenheft
Tätigkeiten	<p>Erarbeiten und prüfen der Machbarkeit einer 2 Schritt-Authentifizierung, wobei es um den zweiten Schritt geht.</p> <p>Erarbeiten und prüfen der Machbarkeit für die Ver- und Entschlüsselung eines FileContents mittels symmetrischen Verschlüsselungsverfahren</p> <p>Erarbeiten und prüfen der Machbarkeit eines DataAccessObject für die Kommunikation zwischen Datenbank und C3rBytes</p> <p>Erarbeiten und prüfen der Machbarkeit eines Passwortgenerator mit Optionen für Länge und Zeichensatz</p> <p>Erarbeiten und prüfen der Machbarkeit, um einen Link eines Eintrags mit dem Defaultbrowser öffnen zu können</p>
Output	<p>Ergebnisse im Hauptdokument hinzufügen</p> <p>Dieses Arbeitspaket soll zeigen, welche Anforderungen realistisch sowie umsetzbar sind und welche nicht oder nur mit erheblichem Mehraufwand umgesetzt werden können.</p> <p>Umsetzbare Konzepte können in die Realisation übernommen werden</p>
Aufwand in Std.	42
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	23.09.2020
Effektive Terminierung	23.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.3
Arbeitspaket Name	Projektstrukturplan
Phase	Vorstudie
Input / Voraussetzung	<p>2.2.1 Risiken identifizieren und bewerten</p> <p>2.2.2 Pflichtenheft</p> <p>2.2.3 PoC's / Machbarkeit</p>
Tätigkeiten	<p>Erstellen des Projektstrukturplans</p> <p>Definierung der Arbeitspakete</p> <p>Beschreibung der Arbeitspakete</p> <p>Terminplanung</p> <p>Resourcenplanung</p>
Output	Projektstrukturplan
Aufwand in Std.	16
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	18.09.2020
Effektive Terminierung	23.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.4
Arbeitspaket Name	Statusbericht 1
Phase	Vorstudie
Input / Voraussetzung	2.1 Projektauftrag 2.2.1 Risiken identifizieren und bewerten 2.2.2 Pflichtenheft erstellen 2.2.3 PoC's / Machbarkeit 2.3 Projektstrukturplan
Tätigkeiten	Verfassung des Berichts für das Statusmeeting 1
Output	Bericht des Statusmeeting1
Aufwand in Std.	5
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	18.09.2020
Effektive Terminierung	19.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.5
Arbeitspaket Name	Präsentation zum Statusbericht 1
Phase	Vorstudie
Input / Voraussetzung	2.4 Statusbericht 1
Tätigkeiten	Verfassung der Präsentation für das Statusmeeting 1
Output	Präsentation zum Statusmeeting1
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	18.09.2020
Effektive Terminierung	19.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.6
Arbeitspaket Name	Reflexion zum Statusmeeting 1
Phase	Vorstudie
Input / Voraussetzung	2.4 Statusbericht 1 2.5 Präsentation zum Statusbericht 1
Tätigkeiten	Verfassung eines Protokolls des Treffens (Statusmeeting)
Output	Reflexion zum Statusmeeting1
Aufwand in Std.	4
Verantwortlich	Jérémie Equey, Mersid Hazbiu, Olaf Schmidt
Bemerkungen	Das Dokument muss nach dem Statusmeeting 1 hochgeladen werden.
Planmässige Terminierung	19.09.2020
Effektive Terminierung	19.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.7
Arbeitspaket Name	Teamsitzungen
Phase	Vorstudie
Input / Voraussetzung	Das Team wird zu wöchentlichen Teamsitzung einberufen.
Tätigkeiten	Besprechung gemäss der Tagesordnung der Sitzung
Output	Protokoll der Sitzung
Aufwand in Std.	5
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	
Effektive Terminierung	11.09.2020, 18.09.2020, 19.09.2020, 23.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	2.8
Arbeitspaket Name	Rollenverteilung
Phase	Vorstudie
Input / Voraussetzung	2.6 Reflexion zum Statusmeeting 1 2.7 Teamsitzung
Tätigkeiten	Bestimmung der neuen Rollen im Team Bestimmung des neuen Projektleiters Neue Rollenverteilung wird in der Dokumentation eingefügt Anpassungen gemäss Bemerkungen während des Status Meeting 1
Output	Protokoll der Sitzung / Dokumentation (Statusbericht 1 und 2)
Aufwand in Std.	1
Verantwortlich	alle
Bemerkungen	
Planmässige Terminierung	23.09.2020
Effektive Terminierung	23.09.2020

1.2.3.3. Konzept

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.1
Arbeitspaket Name	Use Cases
Phase	Konzeption
Input / Voraussetzung	2.1 Projektauftrag 2.2.2 Pflichtenheft 2.4 Statusbericht 1
Tätigkeiten	Bestimmung der Use Cases Erstellung der Diagramme der Use Cases Beschreibung der Use Cases
Output	Use Case Diagramm im UML-Format Use Case Beschreibung pro Use Cases in der Dokumentation
Aufwand in Std.	10
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	28.09.2020
Effektive Terminierung	28.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.2
Arbeitspaket Name	Nutzwertanalyse Verschlüsselungsalgorithmus
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele
Tätigkeiten	Identifizierung von MUSS-Kriterien Bestimmung der WUNSCH-Kriterien Identifizierung von potenziellen Kandidaten endgültige Wahl eines Kandidaten (ausgewählte Variante)
Output	Nutzwertanalyse betreffend des Verschlüsselungsalgorithmus in Form eines Dokuments. Dokument wird in der Dokumentation eingefügt.
Aufwand in Std.	4
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	26.09.2020
Effektive Terminierung	26.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.3
Arbeitspaket Name	Nutzwertanalyse eines Datenbanksystems
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele
Tätigkeiten	Identifizierung von MUSS-Kriterien Bestimmung der WUNSCH-Kriterien Identifizierung von potenziellen Kandidaten eines Datenbanksystems endgültige Wahl eines Kandidaten (ausgewählte Variante)
Output	Nutzwertanalyse betreffend des Datenbanksystems in Form eines Dokuments. Dokument wird in der Dokumentation eingefügt.
Aufwand in Std.	4
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	29.09.2020
Effektive Terminierung	29.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.4
Arbeitspaket Name	Nutzwertanalyse GUI
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele
Tätigkeiten	Muss-Kriterien bestimmen Wunsch-Kriterien bestimmen Kandidaten bestimmen Kandidaten auswählen
Output	Nutzwertanalyse für das GUI, welche in der Dokumentation eingefügt wird
Aufwand in Std.	4
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	26.09.2020
Effektive Terminierung	26.09.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.5
Arbeitspaket Name	Nutzwertanalyse Two-Step Authentication
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele
Tätigkeiten	Muss-Kriterien bestimmen Wunsch-Kriterien bestimmen Kandidaten bestimmen Kandidaten auswählen
Output	Nutzwertanalyse für die Two-Step Authentication, welche in der Dokumentation eingefügt wird
Aufwand in Std.	4
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	01.10.2020
Effektive Terminierung	01.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.1.6
Arbeitspaket Name	Proof of Concept für Apache Derby
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.1.2 Nutzwertanalyse Verschlüsselungsalgorismus
Tätigkeiten	Informationssammlung betreffend Apache Derby Konfigurationstest von Apache Derby (IntelliJ) Erarbeiten und prüfen der Machbarkeit eines DataAccessObject für die Kommunikation zwischen Datenbank und C3rBytes Debugging
Output	Briefing an einer Teamsitzung, Erstellung von Installations- und Konfigurationsanweisungen für andere Mitglieder
Aufwand in Std.	15
Verantwortlich	Jérémie
Bemerkungen	
Planmässige Terminierung	04.10.2020
Effektive Terminierung	04.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.2.1
Arbeitspaket Name	UI-Konzept
Phase	Konzeption
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.1.1 Use Cases
Tätigkeiten	Erstellung von Mockups (GUI) Erstellung von Mockups für die wichtigsten Views
Output	Mock Up im Format jpg oder png werden erstellt
Aufwand in Std.	4
Verantwortlich	Jérémie Equey, Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	07.10.2020
Effektive Terminierung	07.10.2020

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	3.3.1	
Arbeitspaket Name	Qualitätskriterien definieren	
Phase	Konzept	
Input / Voraussetzung	2.1 Projektantrag 2.2.2 Pflichtenheft, Anforderungen, Ziele	
Tätigkeiten	Ermitteln der Qualitätskriterien anhand des Projektauftrags und des Pflichtenheftes	
Output	Qualitätskriterien in tabellarischer Form	
Aufwand in Std.	2	
Verantwortlich	Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	10.10.2020	
Effektive Terminierung	10.10.2020	

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	3.4.1	
Arbeitspaket Name	Loginalgorithmus definieren	
Phase	Konzept	
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.1.2 Nutzwertanalyse Verschlüsselungsalgorithmus	
Tätigkeiten	Ermitteln der Logik für einen sicheren Login-Prozess sowie für den sicheren Zugriff auf die Daten der Software	
Output	UML-Aktivitätsdiagramm Register und Login	
Aufwand in Std.	12	
Verantwortlich	Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	14.10.2020	
Effektive Terminierung	14.10.2020	

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	3.4.2	
Arbeitspaket Name	Domänenmodel / Fachklassenmodel erstellen	
Phase	Konzept	
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele	
Tätigkeiten	Identifizieren von Objekten und Klassen Festlegen des Verhaltens der Objekte und Klassen Identifizieren von Beziehungen zwischen den Klassen Festlegen der Schnittstellen zwischen den Klassen	
Output	Fachklassenmodel bzw. Domänenmodel in Form eines UML-Diagramms	
Aufwand in Std.	4	
Verantwortlich	Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	14.10.2020	
Effektive Terminierung	14.10.2020	

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.4.3
Arbeitspaket Name	Kontextdiagramm erstellen
Phase	Konzept
Input / Voraussetzung	Nutzwertanalyse 2.2.2 Pflichtenheft, Anforderungen, Ziele
Tätigkeiten	Systemabgrenzung definieren Kontextabgrenzung definieren
Output	Kontextabgrenzung UML-Diagramm
Aufwand in Std.	2
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	14.10.2020
Effektive Terminierung	14.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.4.4
Arbeitspaket Name	Klassendiagramm erstellen
Phase	Konzept
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.4.2 Domänenmodell
Tätigkeiten	Klassen identifizieren Identifikation der Attribute der Klasse Identifikation der Methoden der Klasse Beziehungen zwischen Klassen identifizieren
Output	UML-Klassendiagramm bzw. Bausteinsicht der Software
Aufwand in Std.	8
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	18.10.2020
Effektive Terminierung	18.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.4.5
Arbeitspaket Name	Sequenzdiagramme erstellen
Phase	Konzept
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.1.1 Use Cases 3.4.2 Domänenmodell 3.4.3 Kontextdiagramm 3.4.4 Klassenmodell
Tätigkeiten	Ermitteln der Kommunikation zwischen den Klassen bei wichtigen Sequenzen wie Initialer Login, Login, Eintrag erstellen, Eintrag löschen Eintrag ändern, Master-Passwortändern, Master-Passphrase
Output	Ablaufdiagramme in UML-Form
Aufwand in Std.	16
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	22.10.2020
Effektive Terminierung	22.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.4.6
Arbeitspaket Name	Verteilungsdiagramm erstellen
Phase	Konzept
Input / Voraussetzung	2.2.2 Pflichtenheft, Anforderungen, Ziele 3.4.3 Kontextdiagramm
Tätigkeiten	Ermitteln der Hardware- und Softwarekomponenten
Output	Software Sicht UML-Verteilungsdiagramm
Aufwand in Std.	2
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	18.10.2020
Effektive Terminierung	18.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.4.7
Arbeitspaket Name	EER Diagramm
Phase	Konzeption
Input / Voraussetzung	3.1.3 Nutzwertanalyse Datenbanksystem, Anforderungen
Tätigkeiten	Erstellung des Datenbankschemas mit MySQL Workbench Visuelle Darstellung von Datenbanktabellen Erstellung der SQL-Datei Import vom Schema in Derby
Output	Eine SQL-Konfigurationsdatei wird erstellt sowie eine visuelle Darstellung der Beziehungen zwischen den Tabellen Die SQL-Konfiguration-Datei kann in Derby importiert werden.
Aufwand in Std.	3
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	14.10.2020
Effektive Terminierung	14.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.5
Arbeitspaket Name	Testkonzept erstellen
Phase	Konzept
Input / Voraussetzung	2.2.2 Anforderungen und Definitionen im Pflichtenheft 3.4 Technische Konzeption
Tätigkeiten	Ermitteln der Testszenarien anhand der Anforderungen sowie der Softwaresichten Festlegen der Testfälle und deren Ablauf
Output	Auflistung der Testszenarien in tabellarischer Form
Aufwand in Std.	8
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	25.10.2020
Effektive Terminierung	25.10.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	3.6
Arbeitspaket Name	Teamsitzungen
Phase	Konzeption
Input / Voraussetzung	Das Team wird zu wöchentlichen Teamsitzung einberufen.
Tätigkeiten	Besprechung gemäss der Tagesordnung der Sitzung
Output	Protokoll der Sitzung
Aufwand in Std.	8
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	
Effektive Terminierung	25.09.2020, 02.10.2020, 09.10.2020, 16.10.2020, 25.10.2020

1.2.3.4. Realisierung

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.1
Arbeitspaket Name	Crypto-Package
Phase	Implementierung
Input / Voraussetzung	2.2.2 Anforderungen und Definitionen im Pflichtenheft 3.1.1 Use-Cases 3.4.1 Loginalgorithmus definieren/ Aktivitätsdiagramm 3.4.4 Klassendiagramm 3.4.5 Sequenzdiagramme
Tätigkeiten	Folgende Klassen müssen entwickelt werden PasswordEncrypterDecrypter, FileEncrypterDecrypter, PasswordGenerator und StringHasher
Output	"Das Package umfasst alle nötigen Klassen zum Ent- und Verschlüsseln eines Strings (AES). Zudem hat der StringHasher einen String (SHA3-512). Der PasswordGenerator generiert ein Passwort vorgegebener Länge mit vorgegebenem Zeichensatz."
Aufwand in Std.	25
Verantwortlich	Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	01.11.2020
Effektive Terminierung	01.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.2
Arbeitspaket Name	Utils-Package
Phase	Implementierung
Input / Voraussetzung	2.2.2 Anforderungen und Definitionen im Pflichtenheft 3.1.1 Use-Cases 4.1.1 Crypto-Package
Tätigkeiten	Folgende Klassen müssen entwickelt werden ClipboardHandler, CryptoUtils, FileHandler, OSBasedAction, PasswordRevealer, PasswordValidator, UrlOpener
Output	Dieses Package beinhaltet Klassen, welche zu Hilfe genommen werden, um die Funktion anderer Klassen zu erfüllen.
Aufwand in Std.	8
Verantwortlich	Olaf Schmidt
Bemerkungen	Jérémie Equey: OSBasedAction Mersid Hazbiu: PasswordRevealer
Planmässige Terminierung	08.11.2020
Effektive Terminierung	08.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.3
Arbeitspaket Name	GUI
Phase	Implementierung
Input / Voraussetzung	2.2.2 Anforderungen und Definitionen im Pflichtenheft 3.1.4 Nutzwertanalyse GUI 3.4 Softwaresichten, insbesondere Prototyp
Tätigkeiten	Folgende Klassen müssen entwickelt werden Es müssen alle grafischen Oberflächen via Scenebuilder in FXML erstellt werden. MainView, LoginViewPW, LoginViewMPP, AddNewItemView, PasswordGeneratorView, SetMPWView, SetMPPView, ChangeMPWView, ChangeMPPView, AlertView
Output	Dieses Paket stellt alle grafischen Oberflächen bereit.
Aufwand in Std.	32
Verantwortlich	Mersid Hazbiu
Bemerkungen	Olaf Schmidt: SetMPWView, SetMPPView, ChangeMPWView, ChangeMPPView Jérémie Equey: AlertView
Planmässige Terminierung	01.11.2020
Effektive Terminierung	01.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.4
Arbeitspaket Name	Implementierung Controller
Phase	Implementierung
Input / Voraussetzung	2.2.2 Anforderungen und Definitionen im Pflichtenheft 3.1.1 Use Cases, 3.4 Softwaresichten insbesondere Aktivitätsdiagramme 4.1.2 Utils-Package
Tätigkeiten	Folgende Klassen müssen entwickelt werden MainViewController, LoginViewMPW, LoginViewMPP, AddNewItemController, PasswordGeneratorController, SetMasterPWViewController, SetMasterPPViewController, ChangeMPWViewController, ChangeMPPViewController, AlertViewController
Output	Die Controller in diesem Paket setzen die Befehle des Nutzers um und kommunizieren mit dem Model
Aufwand in Std.	40
Verantwortlich	Jérémie Equey, Olaf Schmidt
Bemerkungen	Mersid Hazbiu
Planmässige Terminierung	17.11.2020
Effektive Terminierung	17.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.5
Arbeitspaket Name	Datenbank / Datenbankanbindung
Phase	Implementierung
Input / Voraussetzung	2.2.2 Use Cases, 3.4 Softwaresichten
Tätigkeiten	Folgende Klassen sind zu entwickeln DAO, DatabaseEntryDAO, DatabaseEntry, DBConnection
Output	Eine nutzbare Datenbank inkl. Erstellung des Schemas, Tables, Users und Verschlüsselung
Aufwand in Std.	40
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	24.11.2020
Effektive Terminierung	24.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.1.6
Arbeitspaket Name	Komponentenintegration
Phase	Implementierung
Input / Voraussetzung	2.2.2 Use Cases, 3.4.1 Aktivitätsdiagramm 3.4.4 Klassendiagramm 3.4.5 Sequenzdiagramm Alle Klassen vorhanden
Tätigkeiten	Loginalgorithmus umsetzen Datenbank anbinden
Output	Das Programm kommuniziert gemäss Planung und Definition, wie sie in der Architektur festgelegt worden sind.
Aufwand in Std.	15
Verantwortlich	Jérémie Equey, Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	30.11.2020
Effektive Terminierung	30.11.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.2
Arbeitspaket Name	Unit-Tests
Phase	Implementierung
Input / Voraussetzung	Alle Klassen vorhanden
Tätigkeiten	Schreiben der Unit-Tests für die wichtigsten Funktionen
Output	Unit-Tests
Aufwand in Std.	14
Verantwortlich	Jérémie Equey, Olaf Schmidt
Bemerkungen	
Planmässige Terminierung	04.12.2020
Effektive Terminierung	04.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.3
Arbeitspaket Name	Manuelle Tests / Testprotokolle
Phase	Implementierung
Input / Voraussetzung	3.5 Testkonzept Alle Klassen vorhanden 4.2 Unit-Tests erfolgreich
Tätigkeiten	Manuelles Testen gemäss Testkonzept Rückmeldung an Entwickler
Output	Vollständig ausgefülltes Testprotokoll
Aufwand in Std.	9
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	07.12.2020
Effektive Terminierung	14.12.2020

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	4.4	
Arbeitspaket Name	JavaDoc erstellen	
Phase	Implementierung	
Input / Voraussetzung	Alle Klassen vorhanden Unit-Tests erfolgreich manuelle Tests erfolgreich	
Tätigkeiten	Ergänzen der Code-Dokumentation JAVADOC Clean-up des Codes	
Output	Clean Code und eine JavaDoc der wichtigsten Funktionen	
Aufwand in Std.	8	
Verantwortlich	Jérémie Equey, Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	10.12.2020	
Effektive Terminierung	10.12.2020	

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	4.5	
Arbeitspaket	Refactoring / Bug fixing	
Phase	Realisierung	
Input / Voraussetzung	Projekt aufgesetzt und getestet	
Tätigkeiten	Refactoring von Klassen und bug fixing	
Output	Klassen wurden refactored und die bugs gefixed.	
Aufwand in Std.	10	
Verantwortlich	Jérémie Equey, Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	10.12.2020	
Effektive Terminierung	17.12.2020	

Arbeitspaketbeschreibung		Projekt: C3rBytes
Arbeitspaket Nr.	4.6	
Arbeitspaket	Jar-Datei	
Phase	Realisierung	
Input / Voraussetzung	Test cases und Junit-Tests	
Tätigkeiten	Erstellung einer Jar-Datei für Windows und Mac OS X	
Output	Die Jar-Dateien wurden erstellt	
Aufwand in Std.	10	
Verantwortlich	Jérémie Equey, Olaf Schmidt	
Bemerkungen		
Planmässige Terminierung	13.12.2020	
Effektive Terminierung	13.12.2020	

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	4.7
Arbeitspaket Name	Teamsitzungen
Phase	Realisierung
Input / Voraussetzung	Das Team wird zu wöchentlichen Teamsitzung einberufen.
Tätigkeiten	Besprechung gemäss der Tagesordnung der Sitzung
Output	Protokoll der Sitzung
Aufwand in Std.	6
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	
Effektive Terminierung	30.10.2020, 06.11.2020, 13.11.2020, 20.11.2020, 27.11.2020, 04.12.2020, 11.12.2020

1.2.3.5. Abschluss

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.1
Arbeitspaket Name	Installationsanleitung
Phase	Abschluss
Input / Voraussetzung	Test cases und Junit-Tests JavaDoc
Tätigkeiten	Erstellung einer Installationsbeschreibung. Falls erforderlich, unterscheiden zwischen Windows- und MacOS-Betriebssystemen.
Output	Installationsanleitung
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	14.12.2020
Effektive Terminierung	18.12.2020, 19.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.2.1
Arbeitspaket Name	Wichtigste Fragen definieren
Phase	Abschluss
Input / Voraussetzung	4.6 Jar-Datei
Tätigkeiten	Die wichtigsten Fragen werden für das Nutzerhandbuch definiert
Output	FAQ
Aufwand in Std.	1
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	14.12.2020
Effektive Terminierung	14.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.2.2
Arbeitspaket Name	Benutzerhandbuch schreiben
Phase	Abschluss
Input / Voraussetzung	5.2.1 Wichtigste Fragen definieren
Tätigkeiten	Das Benutzerhandbuch wird geschrieben und mit Screenshots erweitert.
Output	Benutzerhandbuch
Aufwand in Std.	5
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	16.12.2020
Effektive Terminierung	16.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.3.1
Arbeitspaket Name	Dokumentationsvollständigkeit überprüfen
Phase	Abschluss
Input / Voraussetzung	Dokumentation
Tätigkeiten	Die Dokumentation wird mit der Arbeitsmappe verglichen und wird mit den fehlenden Elementen komplettiert.
Output	Dokumentation
Aufwand in Std.	12
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	19.12.2020
Effektive Terminierung	21.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.3.2
Arbeitspaket Name	Dokumentation nach Fehlern überprüfen
Phase	Abschluss
Input / Voraussetzung	Dokumentation
Tätigkeiten	Die Dokumentation wird auf orthographische und inhaltliche Fehler überprüft
Output	Dokumentation
Aufwand in Std.	12
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	19.12.2020
Effektive Terminierung	21.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.4
Arbeitspaket Name	Abschlussbericht wird erstellt
Phase	Abschluss
Input / Voraussetzung	5.2.1 Dokumentationsvollständigkeit überprüfen 5.2.2 Dokumentation auf Fehler überprüfen
Tätigkeiten	Der Abschlussbericht der Dokumentation wird geschrieben. - Projektverlauf - Projekt- und Planungsziele - Schwierigkeiten im Projekt - Positives, Negatives - Varia
Output	Abschlussbericht
Aufwand in Std.	3
Verantwortlich	alle
Bemerkungen	
Planmässige Terminierung	20.12.2020
Effektive Terminierung	20.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.5
Arbeitspaket Name	Fertigstellung Dokumente
Phase	Abschluss
Input / Voraussetzung	5.2.1 Dokumentationsvollständigkeit überprüfen 5.2.2 Dokumentation auf Fehler überprüfen 5.3 Abschlussbericht
Tätigkeiten	Die Layouts der Dokumente werden standardisiert
Output	Druckbereite Dokumente
Aufwand in Std.	4
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	22.12.2020
Effektive Terminierung	22.12.2020

Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.6
Arbeitspaket Name	Dokumente drucken und versenden
Phase	Abschluss
Input / Voraussetzung	5.4 Fertigstellung Dokumente
Tätigkeiten	Die Dokumente werden gedruckt und gebunden und werden mit der Software an die Dozenten geschickt.
Output	Gedruckte Dokumente
Aufwand in Std.	2
Verantwortlich	Jérémie Equey
Bemerkungen	
Planmässige Terminierung	22.12.2020
Effektive Terminierung	22.12.2020



Arbeitspaketbeschreibung	Projekt: C3rBytes
Arbeitspaket Nr.	5.7
Arbeitspaket Name	Teamsitzungen
Phase	Abschluss
Input / Voraussetzung	Das Team wird zu wöchentlichen Teamsitzung einberufen.
Tätigkeiten	Besprechung gemäss der Tagesordnung der Sitzung
Output	Protokoll der Sitzung
Aufwand in Std.	3
Verantwortlich	Mersid Hazbiu
Bemerkungen	
Planmässige Terminierung	
Effektive Terminierung	18.12.2020

2. Software-Engineering

2.1. Produkteinsatz

2.1.1. Anwendungsbereiche

Um den reibungslosen Betrieb des Systems sicherzustellen, unterstützt C3rBytes die Nutzer bei den folgenden Anwendungen:

- Erstellen, Verwalten und Löschen eines Master-Accounts
- Erstellen, Verwalten und Löschen von Profilen

2.1.2. Zielgruppen

Folgende Zielgruppen werden C3rBytes hauptsächlich verwenden:

Nr.	Name	Arbeiten mit dem System	Qualifikationsniveau
1	Endnutzer	<ul style="list-style-type: none"> • Master-Account Registrieren • Passphrase setzen • Einloggen • Master-Account verwalten • Profil erstellen • Profil verwalten 	Normale Anwender. Keine besonderen Qualifikationen notwendig
2	Supporter	<ul style="list-style-type: none"> • Unterstützung und Support für Endanwender 	Informatiker mit mehrjähriger Berufserfahrung
3	Entwickler	<ul style="list-style-type: none"> • Weiterentwicklung • Refactoring • Wartung 	Erfahrung als Java-Entwickler mit Security-Background

Tabelle 11: Zielgruppen

2.1.3. Betriebsbedingungen

Die App sollte idealerweise portabel (auf USB-Stick) sein. Der Zugriff auf die App sollte auch aus jeder Umgebung möglich sein, auf welcher eine aktuelle Version des JDK (enthält der Java-VM) bereitgestellt wird.

2.2. Stakeholders

Mit folgenden Anspruchsgruppen müssen die Anforderungen von C3rBytes abgestimmt werden:

Nr.	Name	Rolle im Unternehmen	Rolle im Projekt	Rolle bei Softwarenutzung
1	Markus Stern Claudio Zesiger	Dozierende, Ausschuss	Kontrollboard <ul style="list-style-type: none"> • Verabschiedung des Pflichtenhefts sowie der Meilensteine • Abnahme des Endproduktes 	<ul style="list-style-type: none"> • Test und Abnahme des Endproduktes
2	Jérémie Equey	Student	<ul style="list-style-type: none"> • Projektleiter • Requirement Engineer • Softwarearchitekt • Entwickler • Softwarearchitekt/Entwickler • Tester 	<ul style="list-style-type: none"> • Entwicklung, Wartung & Support
3	Mersid Hazbiu	Student	<ul style="list-style-type: none"> • Projektleiter • Requirement Engineer • Softwarearchitekt • Entwickler • Softwarearchitekt/Entwickler • Tester 	<ul style="list-style-type: none"> • Entwicklung, Wartung & Support
4	Olaf Schmidt	Student	<ul style="list-style-type: none"> • Projektleiter • Requirement Engineer • Softwarearchitekt • Entwickler • Softwarearchitekt/Entwickler • Tester 	<ul style="list-style-type: none"> • Entwicklung, Wartung & Support
5	Nutzer	keine	keine	<ul style="list-style-type: none"> • Beta-Tester • Eigentliche Nutzer des Systems

Tabelle 12: Stakeholders

2.3. Produktanforderungen

2.3.1. Funktionale Anforderungen

Die Funktionalen Anforderungen entsprechen weitestgehend den Musskriterien.

Nr.	Beschreibung
FA01	Das System muss eine grafische Nutzeroberfläche (GUI) bereitstellen.
FA02	Das System muss dem Nutzer bei der initialen Anmeldung erlauben ein Master-Passwort und eine Passphrase zu setzen.
FA03	Das System muss dem Nutzer erlauben, sich nach der initialen Anmeldung mittels seinen Zugangsdaten einzuloggen.
FA04	Das System muss dem Nutzer erlauben sein Master-Passwort und seine Master-Passphrase unabhängig zu ändern.
FA05	Das System muss dem Nutzer erlauben Daten eines Profils hinzuzufügen.
FA06	Das System muss dem Nutzer erlauben Daten eines Profils zu ändern.
FA07	Das System muss dem Nutzer erlauben Daten eines Profils zu löschen.
FA08	Das System muss dem Nutzer erlauben, mittels Suchfunktion seine Profile zu durchsuchen.
FA09	Das System muss die Datenbank mit dem Master-Passwort schützen.
FA10	Das System muss die Passwörter der Profile mit einem unbekannten Passwort sichern.
FA11	Das System muss dem Nutzer erlauben Weblinks mit dem Default-Browser zu öffnen.
FA12	Das System muss dem Nutzer die Funktionalität bieten starke Passwörter nach seinen Vorgaben zu generieren.
FA13	Das System muss dem Nutzer ermöglichen, das Passwort zu kopieren.
FA14	Das System muss das kopierte Passwort nach 10 Sekunden automatisch aus der Zwischenablage entfernen
FA15	Das System muss dem Nutzer die Löschung seines Master-Accounts ermöglichen.
FA16	Das System muss es dem Nutzer ermöglichen sich ordnungsgemäss auszuloggen.

2.3.2. Nicht funktionale Anforderungen

Nr.	Beschreibung
NF01	Das GUI muss im Zuge der Usability für die Endnutzer leicht und intuitiv bedienbar sein.
NF02	C3rBytes soll trotz der hohen Verschlüsselungskosten alle Abfragen unterhalb zwei Sekunden abwickeln können.

2.3.3. Qualitätsanforderungen

Nr.	Leistung	Beschreibung
QA01	Sicherheit	Das System soll die Daten des Nutzers vor unberechtigtem Zugriff (Authentizität) und vor unberechtigter Veränderung (Integrität) schützen.



2.4. Qualitätszielbestimmung

Unsere App sollte nachfolgende Qualitätskriterien erfüllen

	sehr wichtig	wichtig	weniger wichtig	unwichtig
Sicherheit	x			
Robustheit		x		
Zuverlässigkeit	x			
Korrektheit	x			
Benutzerfreundlichkeit	x			
Effizienz		x		
Portierbarkeit		x		
Kompatibilität		x		

2.5. Use Cases

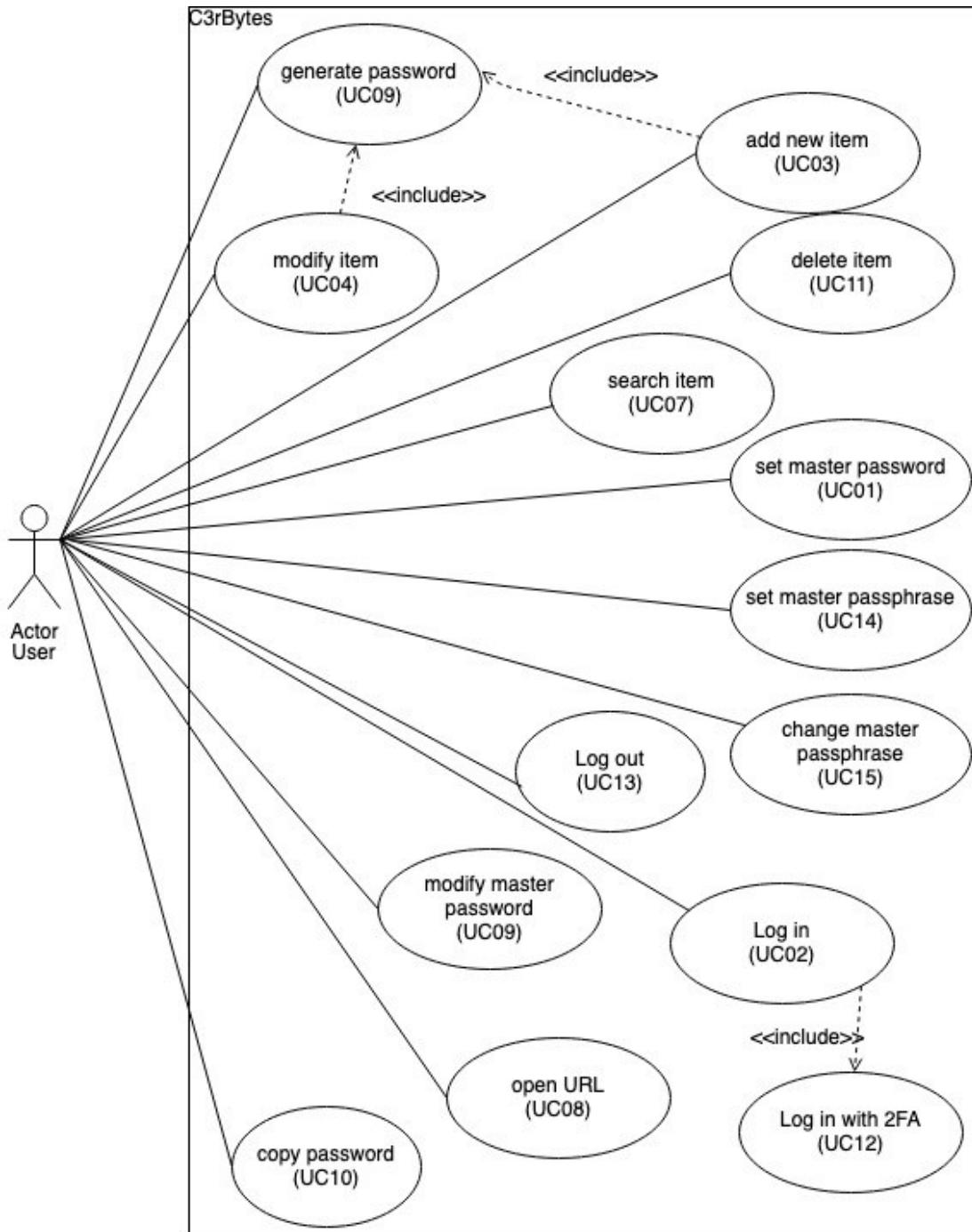


Abbildung 6: Alle Use Cases

2.5.1. Set master password

Abschnitt	Inhalt
ID	UC1
Name	Set master password
Version	2.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Jérémie Equey, Olaf Schmidt
Kurzbeschreibung	Bei der ersten Benutzung muss ein Master-Passwort gesetzt werden.
Auslösendes Ereignis	Der Benutzer öffnet die Applikation und verfügt noch über keinen Master-Account (nur das erste Mal).
Akteure	Anwender
Vorbedingung	Der Benutzer hat sich für die Verwendung der Applikation entschieden, die Applikation noch nie genutzt bzw. noch nie ein Master-Passwort gesetzt.
Nachbedingung	Der Nutzer wird aufgefordert eine Master-Passphrase zu setzen
Ergebnis	Der Benutzer hat ein Passwort zur Verschlüsselung der Datenbank gesetzt
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer öffnet die Applikation. Das System zeigt ein Feld zum Setzen eines Master-Passworts an. 2. Der Benutzer wählt ein starkes Master-Passwort. 3. Der Benutzer speichert die Änderungen 4. Das System erstellt im Hintergrund eine verschlüsselte Datenbank für diesen Benutzer und öffnet die View zum Setzen der Master-Passphrase. 5. Der Benutzer kann nun seine Passwörter speichern.
Alternativszenario	<ol style="list-style-type: none"> 1. keine
Ausnahmeszenario	Wenn der Benutzer das Verfahren vor der Registrierung unterbricht, wird die Anwendung geschlossen. Wenn das Master-Passwort nicht länger als acht Zeichen lang ist, wird eine visuelle Warnung ausgegeben (unzureichende Länge), und der Benutzer wird aufgefordert, ein längeres Master-Passwort zu erstellen.

Tabelle 13: UC01 Set master password

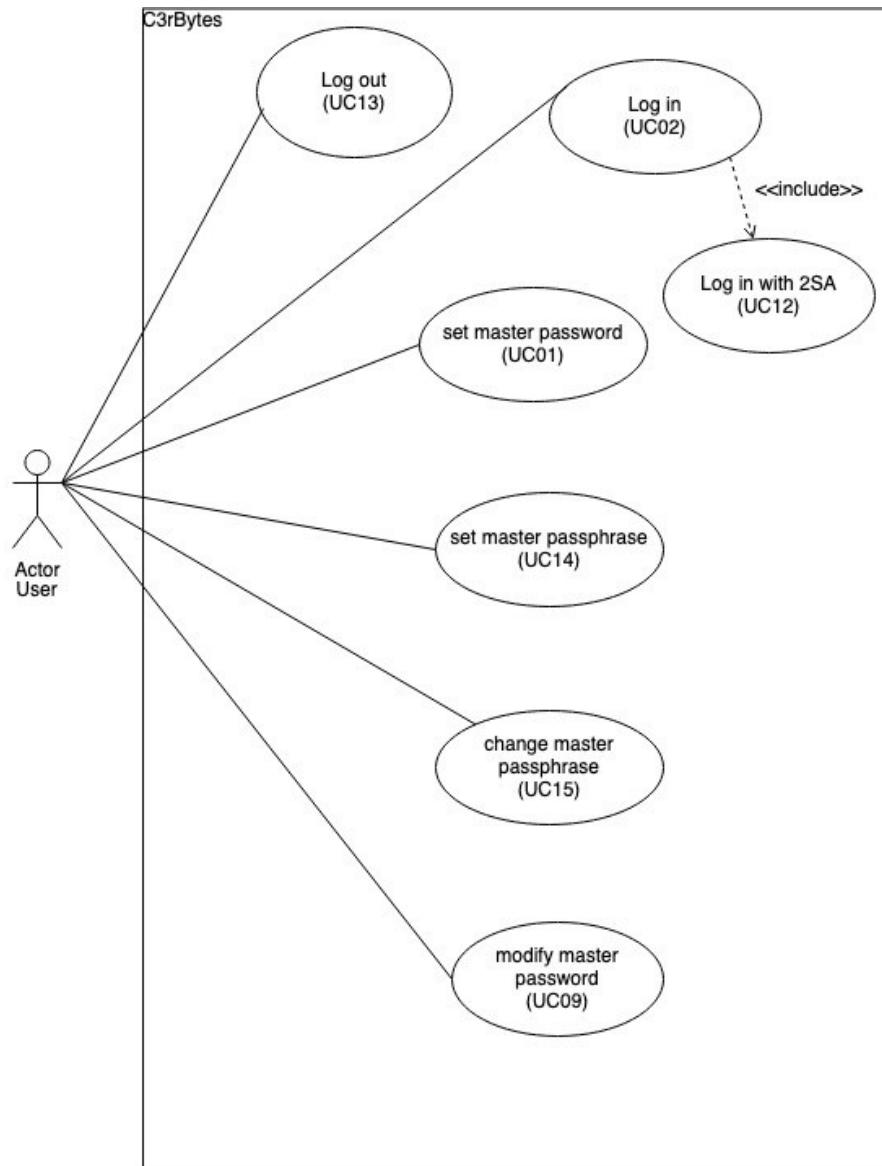


Abbildung 7: Login and Passwords management login

2.5.2. Login

Abschnitt	Inhalt
ID	UC02
Name	Login
Version	2.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Jérémie Equey, Olaf Schmidt
Kurzbeschreibung	Der Benutzer kann sich mit seinem Master-Passwort anmelden (login).

Auslösendes Ereignis	Der Benutzer hat ein Master-Passwort.
Akteure	Anwender
Vorbedingung	Der Benutzer hat sich für die Verwendung der Applikation entschieden.
Nachbedingung	Der Benutzer ist angemeldet (logged in).
Ergebnis	Der Benutzer kann durch die Applikation navigieren (login).
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer öffnet die Applikation 2. Er gibt sein Master-Passwort ein 3. Das System prüft, ob das Passwort korrekt ist 4. Wenn ja, erlaubt das System die den Zugriff auf die DB 5. Der Benutzer kann durch die Applikation navigieren
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer öffnet die Applikation 2. Er gibt sein Master-Passwort ein 3. Das System prüft, ob das Kennwort korrekt ist 4. Falls nicht, fragt das System, das Master-Passwort erneut einzugeben. 5. Hat das System den Versuch als erfolglos registriert, dann zurück zu Punkt 2.
Ausnahmeszenario	Wenn der Benutzer dreimal ein falsches Master-Passwort eingibt, dann schliesst sich die Software. Der Benutzer muss neu starten. Die Datenbank wird nicht entschlüsselt.

Tabelle 14: UC02 Login

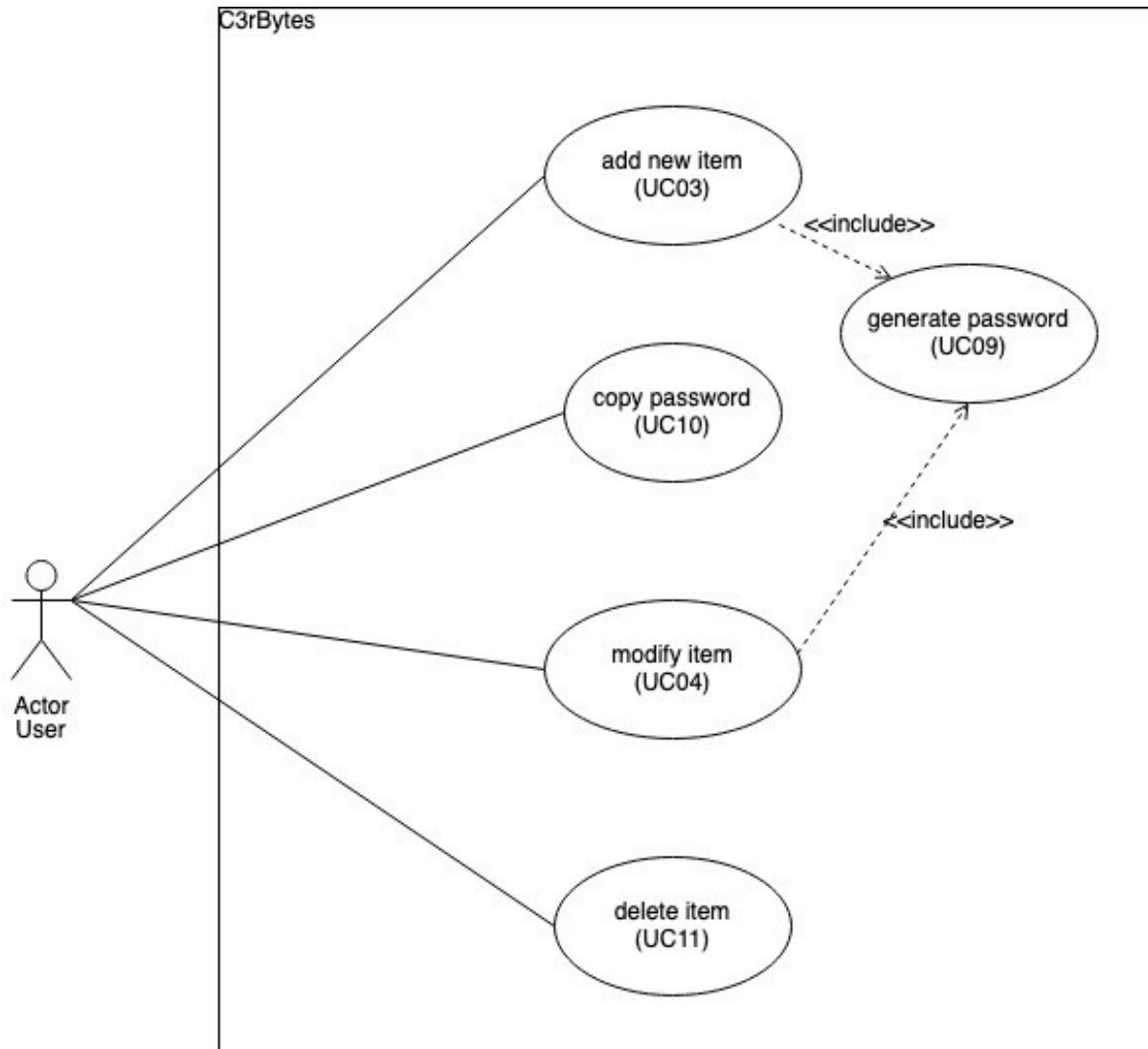


Abbildung 8: Actions related to an item.

2.5.3. add new items

Abschnitt	Inhalt
ID	UC03
Name	enter new item (z.B. password information)
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Mittel
Verantwortlicher	Jérémie Equey, Olaf Schmidt, Mersid Hasbiu
Kurzbeschreibung	Der Benutzer kann Informationen (z.B. Passwort) in der Datenbank speichern.

Auslösendes Ereignis	Drücken des Buttons "add item"
Akteure	Anwender
Vorbedingung	UC02: Der Benutzer ist eingeloggt
Nachbedingung	Das Element wird im Passwort-Manager angezeigt
Ergebnis	Ein Element wird persistent in der DB gespeichert.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer klickt auf das Symbol "+". 2. Er gibt die gewünschten Informationen einschließlich eines starken Passworts ein. 3. Er klickt auf "Speichern". 4. Das System prüft, ob das Kennwort stark genug ist. 5. Wenn dies der Fall ist, speichert das System die Informationen in der Datenbank. 6. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (*****).
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer klickt auf das Symbol "+". 2. Er gibt die gewünschten Informationen einschließlich eines starken Passworts ein. 3. Er klickt auf "Speichern". 4. Das System prüft, ob das Passwort stark genug ist. 5. Wenn nicht, fordert das System den Benutzer auf, ein stärkeres Password zu wählen und zurück zu Punkt 2. 6. Sofern das Passwort sicher ist, speichert das System die Informationen in der Datenbank. 7. Das System zeigt eine Ansicht mit den gespeicherten Einträgen an. Die Passwörter sind versteckt (*****).
Ausnahmeszenario	Der Benutzer unterbricht den Ablauf selbst. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (*****). Bei der Registrierung in der Datenbank trat ein Problem auf, und die nicht registrierten Informationen gingen verloren, dann zeigt das System eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (*****).

Tabelle 15: UC03 enter new items

2.5.4. Modify item

Abschnitt	Inhalt
ID	UC04
Name	Modify item
Version	2.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Mittel
Kritikalität	Mittel
Verantwortlicher	Jérémie Equey
Kurzbeschreibung	Der Benutzer kann Informationen (z.B. Passwort) modifizieren.
Auslösendes Ereignis	Der Benutzer wählt ein Element aus und auf «modify profile»
Akteure	Anwender
Vorbedingung	Es muss ein Element markiert sein. Der Benutzer ist eingeloggt.

Nachbedingung	Der Benutzer kann Elemente modifizieren. Der Benutzer kann in seiner Elementsammlung nach Elementen suchen.
Ergebnis	Elemente wurden geändert.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer wählt einen Eintrag aus seiner Sammlung aus. 2. Er klickt auf "Bearbeiten" / "modify profile". 3. Ein Fenster öffnet sich und zeigt die Informationen an (z.B. Benutzername und Passwort, die mit der Website verknüpft sind). 4. Der Benutzer nimmt die gewünschten Änderungen vor 5. der Benutzer klickt auf "Speichern" / "save". 6. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (*****).
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer wählt einen Eintrag aus seiner Sammlung aus. 2. Er klickt auf "Bearbeiten" / "modify profile". 3. Ein Fenster öffnet sich und zeigt die Informationen an (z.B. Benutzername und Passwort, die mit der Website verknüpft sind). 4. Der Benutzer nimmt die gewünschten Änderungen vor 5. der Benutzer klickt auf "discard" / "discard changes". 6. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an, ohne Änderungen. Die Passwörter sind versteckt (*****).
Ausnahmeszenario	Bei der Registrierung in der Datenbank trat ein Problem auf, und die nicht registrierten Informationen gingen verloren, dann zeigt das System eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (*****).

Tabelle 16: UC04 Modify item

2.5.5. Change master password

Abschnitt	Inhalt
ID	UC05
Name	Change master password
Version	1.0
Datum	03.09.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt, Jérémie Equey
Kurzbeschreibung	Der Benutzer kann sein Master-Passwort ändern.
Auslösendes Ereignis	Der Benutzer drückt auf den Button "Master-Passwort ändern"
Akteure	Anwender, System
Vorbedingung	Der Benutzer ist eingeloggt
Nachbedingung	Das neue Master-Passwort des Benutzers wird gespeichert.
Ergebnis	Die DB wird mit dem neuen Master-Passwort verschlüsselt.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer geht zu seinem Konto. 2. Er klickt auf Master-Passwort ändern. 3. Ein neues Fenster öffnet sich. 4. Der Benutzer gibt sein altes Master-Password ein und ein neues Master-Password ein. 5. Das System prüft, ob das alte Master-Password korrekt ist und ob das neue stark genug ist. <ol style="list-style-type: none"> 5.1. Wenn ja, speichert das System das neue Master-Passwort in der Datenbank. 5.2. Wenn nein, zurück zu 4.
Alternativszenario	1. keines
Ausnahmeszenario	Wenn der Benutzer das Verfahren vor Abschluss des Use Cases unterbricht, wird die Anwendung geschlossen.

Tabelle 17: UC05 Change master password

2.5.6. Delete user account

Abschnitt	Inhalt
ID	UC06
Name	Delete user account
Version	1.0
Datum	03.09.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Jérémie Equey
Kurzbeschreibung	Der Benutzer kann seinen Account löschen.
Auslösendes Ereignis	Der Benutzer drückt den Button "Account löschen"
Akteure	Anwender
Vorbedingung	Der Benutzer ist eingeloggt.
Nachbedingung	Die DB wird gelöscht und kann nicht mehr rekonstruiert werden.
Ergebnis	Die komplette DB wird gelöscht.
Hauptszenario	<ol style="list-style-type: none"> Der Benutzer navigiert zu seinem Konto. Er klickt auf "Account löschen" Ein neues Fenster wird geöffnet und der Benutzer informiert, dass die den Account gelöscht wird. Das System wird beendet.
Alternativszenario	<ol style="list-style-type: none"> Der Benutzer navigiert zu seinem Konto. Er klickt auf "Account löschen" Ein neues Fenster wird geöffnet und der Benutzer informiert, dass die den Account gelöscht wird. Der Benutzer klickt auf «discard». Das Fenster schließt sich. Der Benutzer kehrt zur Hauptansicht zurück.
Ausnahmeszenario	Wenn der Benutzer das Verfahren vor Abschluss des Use Cases unterbricht, zeigt das System dem Benutzer die Hauptansicht.

Tabelle 18: UC06 Delete user account

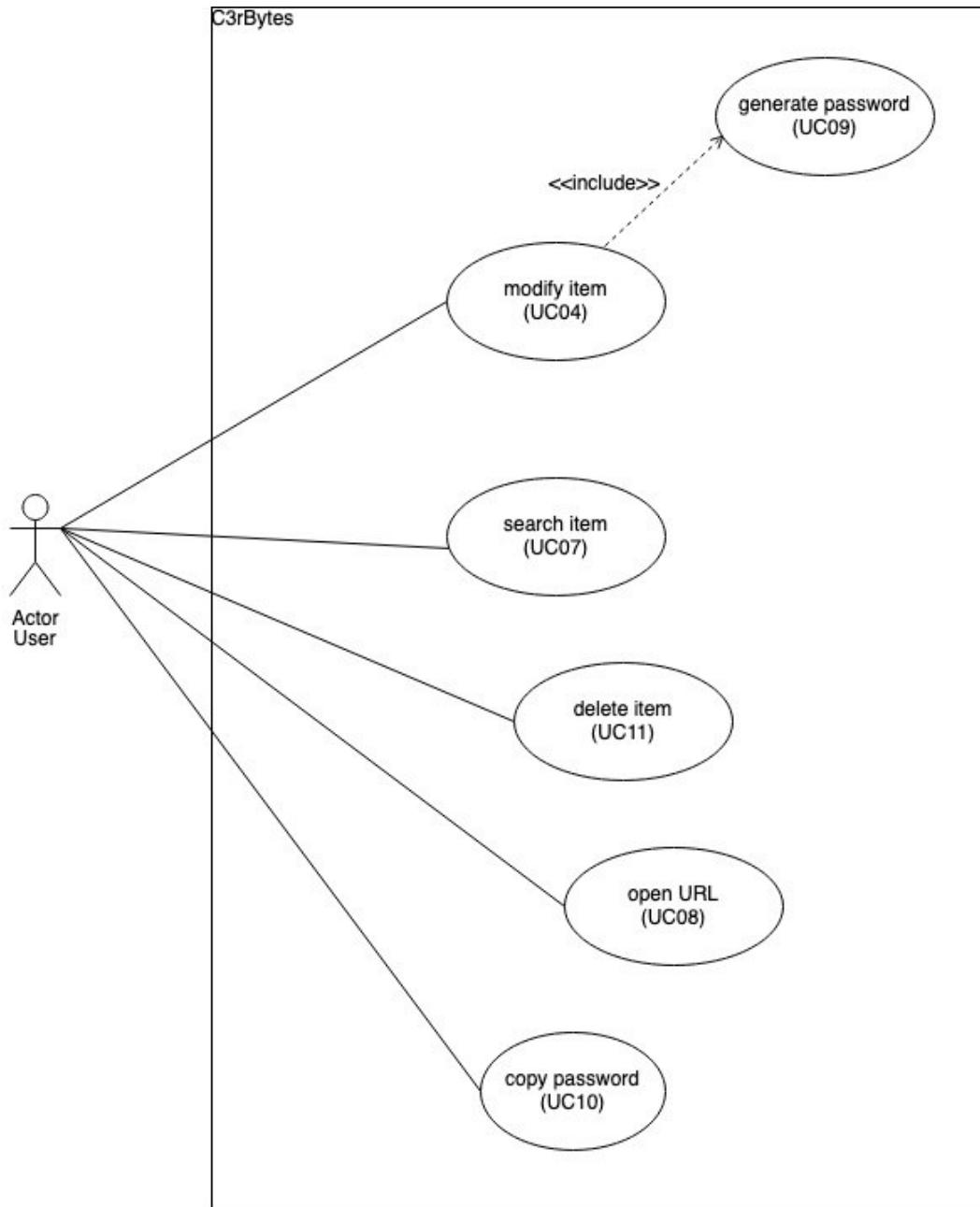


Abbildung 9: Search and related actions

2.5.7. Search item

Abschnitt	Inhalt
ID	UC07
Name	Search item
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Mittel
Kritikalität	Mittel
Verantwortlicher	Jérémie Equey
Kurzbeschreibung	Der Benutzer kann die Datenbank durchsuchen, die seine registrierten Informationen enthält (z.B. Benutzerkonto, Webseite usw.).
Auslösendes Ereignis	Der Benutzer nutzt die Suchfunktion
Akteure	Anwender
Vorbedingung	Der Benutzer ist eingeloggt. Mindestens ein Eintrag ist zuvor in der Datenbank erfasst worden.
Nachbedingung	Das System zeigt relevante Suchergebnisse in seiner Elementsammlung an.
Ergebnis	Das System zeigt dem Nutzer den gesuchten Eintrag an.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer wählt das Suchfeld aus. 2. Der Benutzer gibt seine Suchkriterien ein. 3. Der Benutzer klickt auf "Enter" (Tastaturtaste). 4. Das System liest den Wert im Suchfeld ein und legt eine Transaktion für die Datenbank an. 5. Ein oder mehrere Ergebnisse werden gefunden 6. Diese werden dem Benutzer angezeigt.
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer wählt das Suchfeld aus. 2. Der Benutzer gibt seine Suchkriterien ein. 3. Der Benutzer klickt auf "Enter" (Tastaturtaste). 4. Das System liest den Wert im Suchfeld und legt eine Transaktion für die Datenbank an. 5. Es werden keine Ergebnisse gefunden. 6. Dies wird dem Benutzer kommuniziert
Ausnahmeszenario	

Tabelle 19: UC07 Search item

2.5.8. Open URL

Abschnitt	Inhalt
ID	UC08
Name	Open URL
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Mittel
Kritikalität	Mittel
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Das System erlaubt dem Nutzer Weblinks (URLs) mit dem Default-Browser zu öffnen.
Auslösendes Ereignis	Der Benutzer klickt den Button “Open URL”.
Akteure	Anwender
Vorbedingung	Mindestens ein Eintrag ist zuvor in der Datenbank erfasst und markiert worden, welcher einen Link (URL) enthält (https://example.com).
Nachbedingung	Der Browser öffnet sich und zeigt die gewünschte URL an. Das System bleibt im Hintergrund im selben Zustand geöffnet.
Ergebnis	Im Default-Browser wird die gewünschte Seite aufgerufen.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer selektiert einen Eintrag, welcher eine gültige URL enthält. 2. Der Benutzer klickt auf den Button “Open URL”. 3. Das System startet den Standardbrowser des Betriebssystems des Benutzers und navigiert auf die gewünschte Website.
Alternativszenario	keine
Ausnahmeszenario	keine

Tabelle 20: UC08 Open URL

2.5.9. Generate password

Abschnitt	Inhalt
ID	UC09
Name	Generate password
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Das System erlaubt dem Nutzer starke Passwörter nach Wunschkriterium zu generieren.
Auslösendes Ereignis	Der Benutzer möchte ein neues Passwort registrieren, das mit einem Konto verknüpft ist. Der Benutzer klickt auf die Funktion "Passwort generieren".
Akteure	Anwender
Vorbedingung	Der Benutzer muss eingeloggt sein und nimmt einen neuen Eintrag in die Datenbank vor.
Nachbedingung	Der Benutzer kann das generierte Password als Passwort speichern oder ein neues Password generieren.
Ergebnis	Ein neues Password wurde generiert.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer möchte ein sicheres Passwort generieren. 2. Er klickt auf "Passwort generieren". 3. Er wählt seine Kriterien aus (Länge, Passphrase vs. Passwort, Sonderzeichen usw.). 4. Das System generiert ein Kennwort gemäß den Kriterien 5. Das System zeigt das Ergebnis an 6. Sobald der Benutzer zufrieden ist, speichert er das Passwort in die Zwischenablage.
Alternativszenario	keine
Ausnahmeszenario	Sollte beim Speichern in die Datenbank ein Problem auftreten, dann zeigt das System die View vor der Transaktion mit der Datenbank erneut an.

Tabelle 21: UC09 Generate password

2.5.10. Copy password

Abschnitt	Inhalt
ID	UC10
Name	Copy password
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Mittel
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Der Benutzer kann ein Passwort in den Arbeitsspeicher seines Computers zur Verwendung in einer anderen Applikation kopieren.
Auslösendes Ereignis	Der Benutzer markiert das Passwortfeld und kopiert den Inhalt mittels «copy password» oder im Kontextmenü (Rechtsklick der Maus) in die Zwischenablage
Akteure	Anwender
Vorbedingung	Mindestens ein Passwort wurde in der Datenbank registriert.
Nachbedingung	Der Benutzer kann sein Passwort aus der Zwischenablage einfügen. Nach 5 Sekunden wird das Password vom Zwischenablage gelöscht.
Ergebnis	Ein Passwort wurde kopiert.
Hauptszenario	<ol style="list-style-type: none"> Der Benutzer wählt das Passwort, welches er kopieren möchte, indem er auf die entsprechende Zeile klickt. Er macht einen Rechtsklick. Das System bietet dem Benutzer die Funktion "Passwort kopieren" an. Durch Anklicken der Option wird das Passwort (im Klartext) in den Speicher (Zwischenablage) kopiert. Der Benutzer kann das Ergebnis dann an anderer Stelle einfügen.
Alternativszenario	1. keine
Ausnahmeszenario	Wenn die Aktion "Kopieren" nicht funktioniert hat, muss der Benutzer die Aktion noch einmal wiederholen.

Tabelle 22: UC10 Copy password

2.5.11. Delete item

Abschnitt	Inhalt
ID	UC11
Name	Delete item
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Mittel
Kritikalität	Mittel
Verantwortlicher	Jérémie Equey
Kurzbeschreibung	Der Benutzer kann Informationen (z.B. Passwort) löschen.
Auslösendes Ereignis	Der Benutzer wählt ein Element aus und drückt «Delete profile»
Akteure	Anwender
Vorbedingung	Es muss ein Element markiert sein.
Nachbedingung	Der Benutzer kann Elemente löschen. Der Benutzer kann in seiner Elementsammlung nach Elementen suchen.
Ergebnis	Elemente wurden geändert.
Hauptszenario	<ol style="list-style-type: none"> Der Benutzer wählt einen Eintrag aus seiner Sammlung aus. Er klickt auf "Löschen" / "delete profile". Das System fordert den Benutzer auf, diese Aktion zu bestätigen, indem es ihm ein Pop-up-Fenster präsentiert: "The entry will be deleted. Are you sure?": answers (okay, cancel) Der Benutzer bestätigt die Aktion durch klicken auf "Okay". Das System löscht das Element dauerhaft. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (****).
Alternativszenario	<ol style="list-style-type: none"> Der Benutzer wählt einen Eintrag aus seiner Sammlung aus. Er klickt auf "Löschen" / "delete profile". Das System fordert den Benutzer auf, diese Aktion zu bestätigen, indem es ihm ein Pop-up-Fenster präsentiert: The entry will be deleted. Are you sure?: answers (ok, cancel) Der Benutzer bestätigt die Aktion durch klicken auf "Cancel". Die Aktion wird annulliert. Das System zeigt eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (****).
Ausnahmeszenario	Bei der Registrierung in der Datenbank trat ein Problem auf, und die nicht registrierten Informationen gingen verloren, dann zeigt das System eine Ansicht mit den gespeicherten Artikeln an. Die Passwörter sind versteckt (****).

Tabelle 23: UC11 Delete item

2.5.12. Login with 2 Step Authentication

Abschnitt	Inhalt
ID	UC12
Name	Login with 2SA
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Der Benutzer kann sich mit seinem Master-Passwort anmelden (login). Wenn der erste Schritt erfolgreich ist, muss er noch eine Master-Passphrase eingeben. Erst wenn das MasterPassword, sowie die Master-Passphrase korrekt ist, gelangt der User zur Hauptansicht.
Auslösendes Ereignis	Der Benutzer hat ein Master-Passwort sowie ein Master-Passphrase konfiguriert.,
Akteure	Anwender
Vorbedingung	Der Benutzer hat sich für die Verwendung der Applikation entschieden.
Nachbedingung	Der Benutzer ist korrekt angemeldet (logged in).
Ergebnis	Der Benutzer kann durch die Applikation navigieren (logged in).
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer öffnet die Applikation 2. Er gibt sein Master-Passwort ein 3. Das System prüft, ob das Passwort korrekt ist 4. Wenn ja, 5. Der Benutzer muss seine Master-Passphrase eingeben. 6. Das System prüft, ob der eingegebene String gültig ist. 7. Wenn das der Fall ist, erlaubt das System den Zugriff auf die DB 8. Der Benutzer kann nun durch die Applikation navigieren
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer öffnet die Applikation 2. Er gibt sein Master-Passwort ein 3. Das System prüft, ob das Passwort korrekt ist 4. Wenn ja, 5. Der Benutzer muss seine Master-Passphrase eingeben. 6. Das System prüft, ob der eingegebene String gültig ist. 7. Wenn das nicht der Fall ist, muss der Benutzer Schritt 5 wiederholen.
Ausnahmeszenario	Wenn der Benutzer drei falsche Master-Passphasen eingibt, dann schliesst sich die Software. Der User muss neu starten.

Tabelle 24: UC12 Login with 2SA

2.5.13. Logout

Abschnitt	Inhalt
ID	UC13
Name	Logout
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Der Benutzer kann sich abmelden (log out).
Auslösendes Ereignis	Der Benutzer hat ein Master-Passwort.
Akteure	Anwender, System
Vorbedingung	Der Benutzer hat sich für die Verwendung der Applikation entschieden.
Nachbedingung	Der Benutzer ist abgemeldet (logged out).
Ergebnis	Die Anwendung zeigt ein Fenster zur Eingabe eines Master-Passworts an (wenn logged out).
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer ist angeloggt und klickt ausloggen (log out) 2. Das System loggt den User aus 3. Das System verschlüsselt die Dateien 4. Die Fenster werden geschlossen. 5. Das System wurde geschlossen. (exit)
Alternativszenario	1. keine
Ausnahmeszenario	keine

Tabelle 25: UC13 Logout

2.5.14. Set Master Passphrase

Abschnitt	Inhalt
ID	UC14
Name	Set master passphrase
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Hoch
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Bei der ersten Benutzung muss ein Master Passphrase gesetzt werden.
Auslösendes Ereignis	Das System prüft, ob die Datei c3r.c3r auf dem System des Users vorhanden ist. Ist die Datei abwesend, muss der User ein Master-Passphrase setzen.
Akteure	Anwender
Vorbedingung	Der Benutzer öffnet die Applikation und verfügt noch über keinen Master-Account (nur das erste Mal). Er hat bereits ein Master Password gesetzt.
Nachbedingung	Der Nutzer wird aufgefordert eine Master-Passphrase zu setzen
Ergebnis	Der Benutzer hat ein Passwort zur Verschlüsselung der c3r.c3r Datei. Diese Datei enthält ein Schlüssel, der die Passwörter in der Datenbank verschlüsselt.
Hauptszenario	<ol style="list-style-type: none"> Der Benutzer hat die Applikation geöffnet und ein Master Password gesetzt. Das System zeigt in einem neuen Fenster ein Feld zum Setzen eines Master-Passphrase an. Der Benutzer wählt ein starkes Master-Passphrase. Der Benutzer speichert die Änderungen Das System erstellt im Hintergrund eine verschlüsselte Datei für diesen Benutzer und öffnet die Main View. Der Benutzer kann nun seine Passwörter speichern.
Alternativszenario	<ol style="list-style-type: none"> Der Benutzer hat die Applikation geöffnet und ein Master Password gesetzt. Das System zeigt in einem neuen Fenster ein Feld zum Setzen eines Master-Passphrase an. Der Benutzer klickt auf "discard". Das System löscht die Datenbank und schliesst das Programm.
Ausnahmeszenario	Wenn der Benutzer das Verfahren vor der Registrierung unterbricht, wird die Anwendung geschlossen. Wenn das Master-Passphrase nicht mindestens acht Zeichen lang ist, wird eine visuelle Warnung ausgegeben (unzureichende Länge), und der Benutzer wird aufgefordert, ein längeres Master-Passwort zu erstellen.

Tabelle 26: UC14 Set master passphrase

2.5.15. Change Master Passphrase

Abschnitt	Inhalt
ID	UC15
Name	Change master password
Version	1.0
Datum	06.10.2020
Autoren	Jérémie Equey
Priorität	Hoch
Kritikalität	Mittel
Verantwortlicher	Olaf Schmidt
Kurzbeschreibung	Der Benutzer kann sein Master-Passphrase ändern.
Auslösendes Ereignis	Der Benutzer drückt auf den Button "Master-Passphrase ändern"
Akteure	Anwender
Vorbedingung	Der Benutzer ist eingeloggt
Nachbedingung	Das neue Master-Passphrase des Benutzers wird gespeichert.
Ergebnis	Die c3r.c3r Datei wird mit dem neuen Master-Passphrase verschlüsselt.
Hauptszenario	<ol style="list-style-type: none"> 1. Der Benutzer geht zu seinem Konto. 2. Er klickt auf Master-Passphrase ändern. 3. Ein neues Fenster öffnet sich. 4. Der Benutzer gibt sein altes Master-Passphrase ein 5. Der Benutzer gibt sein neues Master-Passphrase ein 6. Das System prüft, ob das alte Master-Passphrase korrekt. <ol style="list-style-type: none"> 6.1. Wenn ja, speichert das System das neue Master-Passwort in der Datenbank. 6.2. Wenn nein, zurück zu 4.
Alternativszenario	<ol style="list-style-type: none"> 1. Der Benutzer geht zu seinem Konto. 2. Er klickt auf Master-Passphrase ändern. 3. Ein neues Fenster öffnet sich. 4. Der Benutzer gibt seine alte Master-Passphrase ein 5. Der Benutzer gibt seine neue Master-Passphrase ein <ol style="list-style-type: none"> 5.1. Der Benutzer klickt auf "discard" 6. Das System schliesst das Fenster. Das Master Passphrase wurde nicht geändert.
Ausnahmeszenario	Wenn der Benutzer das Verfahren vor Abschluss des Use Cases unterbricht, wird die Applikation geschlossen.

Tabelle 27: UC15 Change master password

2.6. POC's

Da dem gesamten Team keine professionellen Entwickler angegliedert sind, mussten alle nachfolgenden Konzepte auf deren Machbarkeit überprüft werden. Insbesondere sind dies:

- 2FA (Zwei-Faktor-Authentifizierung) mit Google Authenticator
- OpenURL
- DAO/DB-Anbindung mit JDBC-Driver
- AES Ver- und Entschlüsselung
- SHA3 Hashing
- Passwortgenerator

Um das Dokument nicht unnötig aufzublähen, ist der Quellcode im Repository unter dem Link https://git.ffhs.ch/jeremie.equey/c3rbytes/-/tree/start_over_merge/POC zu finden. Im Verlaufe des Projektes mussten wir jedoch die 2 Faktor-Authentifizierung in ein 2-Schritt-Login abändern.

2.7. Nutzwertanalyse Komponenten

2.7.1. Verschlüsselungsalgorithmen

2.7.1.1. Auswahl

Um unsere Daten zu sichern, brauchen wir einen (symmetrischen) Algorithmus, um sie zu verschlüsseln und entschlüsseln. Diese Algorithmen müssen die folgenden Kriterien erfüllen:

Musskriterien	Begriff
M01	Die Nutzung des Algorithmus ist kostenfrei.
M02	Der Algorithmus wird von Spezialisten als sicher angesehen.
M03	Der Algorithmus muss ein verbreiteter Algorithmus sein. (standardisiert)
M04	Der Algorithmus ist ein symmetrischer Verschlüsselungsalgorithmus.

Wir haben die folgenden Algorithmen ausgewählt: Advanced Encryption Standard (AES), Blowfish, Twofish, Data Encryption Standard (DES), Serpent sowie Triple Data Encryption Standard (3DES). Alle diese Algorithmen erfüllen die M03- und M04-Kriterien vollständig. Sie alle werden in der Branche üblicherweise eingesetzt. Alle sind auch öffentlich verfügbar und gebührenfrei. Für Sicherheitsaudits ist es wichtig, dass diese Algorithmen mit dem Kerckhoffschen Prinzip übereinstimmen, dass "die Sicherheit eines Kryptosystems nur auf der Geheimhaltung des Schlüssels beruhen sollte", was bedeutet, dass "andere Parameter als öffentlich bekannt vorausgesetzt werden sollten". Auch wenn das Kerckhoffsche Prinzip nicht dazu ermutigt, das Verschlüsselungssystem öffentlich zu machen, tendiert die Doktrin zu einer Tendenz, die davon ausgeht, dass Verschlüsselungssysteme umso sicherer sind, wenn sie öffentlich sind, weithin untersucht werden und kein nennenswerter Angriff bekannt ist". ([Wikipedia.org](#))

Folglich werden sich die ausgewählten Algorithmen durch das M02-Kriterium (Sicherheit) unterschieden.

2.7.1.2. AES

Musskriterien	Erfüllungsgrad
M02	Ja

Die Prozessoren unserer Computer verfügen über Anweisungen für die Kodierung und Dekodierung mit AES. Dies macht Side-Channel Angriffe sehr schwierig. Darüber hinaus gibt es bei korrekter Implementierung keine praktischen Angriffe (also keine theoretischen Angriffe), die es erlauben würden, verschlüsselte Daten ohne Kenntnis des Verschlüsselungsschlüssels zu lesen. ([Wikipedia.org](#))

2.7.1.3. Blowfish

Musskriterien	Erfüllungsgrad
M02	Nein, Blockgrösse

Im Jahr 2016 demonstrierte der SWEET32-Angriff, wie man Geburtstagsangriffe (Birthday Attacks) nutzen kann, um Klartextwiederherstellung (d.h. die Entschlüsselung von Chiffriertext) gegen Chiffren mit einer 64-Bit-Blockgröße durchzuführen. Das GnuPG-Projekt empfiehlt, dass Blowfish aufgrund seiner geringen Blockgröße nicht zum Verschlüsseln von Dateien verwendet werden sollte, die größer als 4 GB sind.

Dennoch hat der Erfinder, Bruce Schneier, empfohlen, zu seinem Blowfish-Nachfolger Twofish zu wechseln. ([Wikipedia.org](#))

2.7.1.4. Twofish

Musskriterien	Erfüllungsgrad
M02	Ja, aber

Twofish gehörte zu den fünf Finalisten des AES-Wettbewerbs, wurde aber nicht für den Standard ausgewählt. ([Wikipedia.org](#))

2.7.1.5. DES

Musskriterien	Erfüllungsgrad
M02	Nein, Schlüsselgrösse (56bits)

Der Einsatz von DES wird heute nicht mehr empfohlen, da es langsam ausgeführt wird und sein zu kleiner Schlüsselraum einen systematischen Angriff in angemessener Zeit ermöglicht. ([Wikipedia.org](#))

2.7.1.6. Serpent

Musskriterien	Erfüllungsgrad
M02	Ja

Serpent kam im Wettbewerb zusammen mit dem Advanced Encryption Standard (AES) in die Endrunde, wo sie nach Rijndael den zweiten Platz belegte. ([Wikipedia.org](#))

2.7.1.7. 3DES

Musskriterien	Erfüllungsgrad
M02	Ja

Dieser Algorithmus wird von grossen IT-Firmen wie Microsoft oder Mozilla Firefox verwendet.

2.7.1.8. Wahl der Varianten

Daher wählen wir die folgenden drei Algorithmen (Varianten) aus, auf die wir die folgenden Kriterien anwenden werden, und stellen fest, ob sie unsere Kriterien in keiner Weise erfüllen, vollständig erfüllen oder sogar übertreffen:

Varianten	
V1	AES
V2	3DES
V3	Serpent

2.7.1.9. Nutzwertanalyse

2.7.1.9.1. Wunschkriterien

Wir haben die folgenden Wunschkriterien definiert, die für unser Projekt wichtig sind:

- **W01:** Die Sicherheit des Algorithmus ist in den kommenden Jahren gewährleistet
- **W02:** Die Ver- oder Entschlüsselungsgeschwindigkeit ist hoch.
- **W03:** Die Länge der Schlüssel kann eingestellt werden
- **W04:** Die Implementierung des Algorithmus in der Sprache Java erfolgt problemlos

Wir vergleichen unsere drei Varianten nach dem Grad ihrer Erfüllung dieser Kriterien multipliziert mit der oben erhaltenen Gewichtung. Es wird die Variante mit der höchsten Punktzahl ausgewählt.

2.7.1.9.2. Präferenzmatrix

Punktevergabe	
0	weniger wichtig
1	gleich wichtig
2	wichtiger

Präferenzmatrix						
Kriterium	W01	W02	W03	W04	Summe	Faktor %
W01		0	1	1	2	0,13
W02	2		2	2	6	0,40
W03	0	0		2	2	0,13
W04	2	1	2		5	0,33
Summe					15	1,00

Die Gewichtung ist jetzt definiert, so können wir mit der Nutzwertanalyse fortfahren und eine Variante bestimmen.

Bewertung n	
0	Erfüllt das Kriterium nicht
1	Erfüllt das Kriterium nur teilweise oder mit Workaround (z.B. Zusatztool)
2	Erfüllt das Kriterium vollständig
3	Übertrifft das Kriterium (z.B. mit weiteren Funktionalitäten als definiert)

Verschlüsselungsalgorithmen		AES		3DES		Serpent	
Kriterium	Gewichtung g	n	g*n	n	g*n	n	g*n
W01	0,133	2,000	0,267	1,000	0,133	1,000	0,133
W02	0,400	2,000	0,800	2,000	0,800	0,000	0,000
W03	0,133	2,000	0,267	2,000	0,267	2,000	0,267
W04	0,333	2,000	0,667	2,000	0,667	0,000	0,000
		1	2,000		1,867		0,400

Abbildung 10: Nutzwertanalyse Verschlüsselungsalgorithmen

2.7.1.10. Lösungswahl/Entscheid

Die Variante 1 (AES) ist gewählt. Die Software erfüllt nicht nur alle unsere obligatorischen Kriterien (Musskriterien), sondern entspricht auch perfekt unseren Wunschkriterien.

Der Unterschied zwischen AES und Twofish ist sicherlich gering. Java bietet eine sehr gute Implementierung. Nichtsdestotrotz bestätigt uns die Tatsache, dass AES «DER Standard» für die kommenden Jahre darstellt, in unserer Wahl. Daneben hat Microsoft im Dezember 2018 beschlossen, 3DS für Office 365 nicht mehr zu verwenden. Die kurze Blockgröße von 64 Bit macht 3DES anfällig für Blockkollisionsangriffe, wenn es zur Verschlüsselung großer Datenmengen mit dem gleichen Schlüssel verwendet wird. Der Sweet32-Angriff zeigt, wie dies in TLS und OpenVPN ausgenutzt werden kann. OpenSSL enthält es daher seit August 2016 standardmäßig nicht mehr. Das ist kein gutes Signal. ([Wikipedia.org](#))

Was Serpent betrifft, so wird sie durch die Schwierigkeit, sie in der Java-Sprache richtig umzusetzen, stark benachteiligt.

2.7.2. Datenbanksystem-Vergleich für unsere Applikation

In unserer Applikation wollen wir ein relationales Datenbanksystem aufbauen. Der Gedanke ist einfach. Es gibt viele Elemente, die der Benutzer in die Datenbank eingeben wird, die ähnlich sein können und daher wahrscheinlich wiederholt werden. Wenn wir unser Produkt weiterentwickeln und Funktionen hinzufügen würden, mit denen Daten in der Datenbank gespeichert werden können, dann wäre ein Relationales System in der Standardform 3NF (3NF) eine gute Option.

2.7.2.1. Auswahl

Musskriterien	Begriff
M01	Die Nutzung des Datenbanksystems ist kostenfrei.
M02	Die Nutzung des Systems unterliegt einer Open-Source-Lizenz (keine proprietäre Lizenz).
M03	Das relationale Datenbankmanagementsystem kann in unser Java-Programm eingebettet werden (JAR Datei).
M04	Das relationale Datenbankmanagementsystem muss ein verbreitetes System sein. (standardisiert)

Unser System sollte sowohl auf einem Computer als auch auf einem an einen Computer angeschlossenen USB-Stick laufen können. Daher muss die Datenbank unserer Anwendung als eigenständige Anwendung funktionieren (M02). Dies sollte für den Benutzer transparent sein. Er sollte nicht selbst eine Datenbank auf seinem System installieren und diese dann mit unserer Anwendung verbinden müssen. Dieses Kriterium schränkt die Auswahl der Kandidaten ein.

Die Auswahl in der Welt der relationalen Datenbanksysteme ist riesig. Dennoch haben wir die folgende Software ausgewählt (aus dieser [Liste](#)): [H2 Database](#), [HSQLDB](#) (HyperSQL Database), [Apache Derby Database](#), [SQLite Database](#), welche unsere [Musskriterien](#) erfüllen.

M0_n	Apache Derby	HSQLDB	SQLite Database
M01	ja	ja	ja
M02	ja	ja	ja
M03	ja	ja	ja
M04	ja	ja	ja

Nichtsdestotrotz haben wir auch die folgenden Datenbanksysteme analysiert: [YugabyteDB](#), [Percona Server for MySQL](#), [MariaDB](#), [MySQL](#), [PostgreSQL](#). Aber Angesichts des Umfangs unseres Anwendungsfalls wäre es jedoch so, als würde man mit einer Kanone auf Spatzen schießen. (*Keep it simple*)

2.7.2.2. Wahl der Varianten

Daher wählen wir die drei Datenbanksysteme (Varianten) aus, auf die wir die folgenden Kriterien anwenden werden, und stellen fest, ob sie unsere Kriterien in keiner Weise erfüllen, vollständig erfüllen oder sogar übertreffen:

Varianten	
V1	Apache Derby DB
V2	HSQLDB
V3	SQLite Database

2.7.2.3. Nutzwertanalyse

2.7.2.3.1. Wunschkriterien

Wir haben die folgenden Wunschkriterien definiert, die für unser Projekt wichtig sind:

- **W01:** Das Datenbanksystem hat Sicherheitsmerkmale, d.h. es unterstützt Ver- und Entschlüsselung.
- **W02:** Die Grösse des Datenbanksystems sollte klein sein. (ohne die Datenbankdaten)
- **W03:** Das Datenbankensystem ist gut dokumentiert.
- **W04:** Die Implementierung des Datenbankensystems in der Sprache Java erfolgt problemlos
- **W05:** Die Sicherheit des Datenbanksystems ist in den kommenden Jahren gewährleistet (regelmässige Updates und Patches)

Wir vergleichen unsere drei Varianten wiederum nach dem Grad ihrer Erfüllung dieser Kriterien multipliziert mit der oben erhaltenen Gewichtung. Es wird die Variante mit der höchsten Punktzahl ausgewählt.

2.7.2.3.2. Präferenzmatrix

Punktevergabe	
0	weniger wichtig
1	gleich wichtig
2	wichtiger

		Präferenzmatrix						
Kriterium		W01	W02	W03	W04	W05	Summe	Faktor %
W01		2	2	1	2	2	7	0,35
W02	0		0	0	1	1	1	0,05
W03	1	2		0	1	1	4	0,2
W04	1	2	1		2	2	6	0,3
W05	0	1	1	0		2	2	0,1
Summe							20	1,00

Die Gewichtung ist definiert, so können wir mit der Nutzwertanalyse fortfahren und eine Variante bestimmen.

Bewertung n								
0	Erfüllt das Kriterium nicht							
1	Erfüllt das Kriterium nur teilweise oder mit Workaround (z.B. Zusatztool)							
2	Erfüllt das Kriterium vollständig							
3	Übertrifft das Kriterium (z.B. mit weiteren Funktionalitäten als definiert)							
Datenbanksystem		Apache Derby		HSQLDB		SQLite Database		
Kriterium	Gewichtung g	n	g*n	n	g*n	n	g*n	
W01	0,350	3,000	1,050	3,000	1,050	0,000	0,000	
W02	0,050	3,000	0,150	2,000	0,100	2,000	0,100	
W03	0,200	2,000	0,400	2,000	0,400	2,000	0,400	
W04	0,300	3,000	0,900	3,000	0,900	2,000	0,600	
W05	0,100	3,000	0,300	2,000	0,200	2,000	0,200	
		1,00		2,800		2,650		
							1,300	

Abbildung 11: Nutzwertanalyse Datenbanksystem

2.7.2.4. Lösungswahl/Entscheid

Die Variante 1 (Apache Derby) ist gewählt. Die Software erfüllt nicht nur alle unsere obligatorischen Kriterien (Musskriterien), sondern entspricht auch perfekt unseren Wunschkriterien.

Der Unterschied zwischen Apache Derby und HSQLDB ist sicherlich gering. Java bietet eine sehr gute Implementierung beider Systeme. Nichtsdestotrotz haben wir mit Apache Derby, das vollständig in Java implementiert ist, die Garantie für eine zu 100 % quelloffene, standardkonforme Software. Außerdem wurde sie 1997 geschaffen und kontinuierlich aktualisiert (etwa eine Version pro Jahr). Die Apache Software Foundation ist für uns eine Garantie für Seriosität.

SQLite bietet ein Verschlüsselungssystem an, allerdings in der kostenpflichtigen Version. Folglich verliert sie mit dieser Option viele Punkte und kann sich dadurch nicht behaupten.

2.7.3. Two-Factor / Two-Step Authentication

2.7.3.1. Auswahl

Die Auswahl der Two-Factor / Two-Step Authentication ist wichtig, um die Sicherheit der Software über die Verschlüsselung hinaus zu gewährleisten. Um eine Auswahl zu treffen, müssen die folgenden Musskriterien erfüllt werden:

Musskriterien	Begriff
M01	Die Authentifizierungsmethode ist kostengünstig oder kostenlos.
M02	Die Authentifizierungsmethode ist sicher.

Die folgenden Authentifizierungsmethoden wurden ausgewählt:

Google Authenticator, Authy, Email und Passphrase werden als Two-Factor Authentication-Methoden (2FA) analysiert und bei der Two-Step-Authetication eine Verschlüsselung der Datenbank mit verschlüsselten Kennwörtern mit Schlüssel in einem separaten File. Google Authenticator und Authy sind dedizierte Apps, die auf gängigen Smartphones verfügbar sind, und periodisch neue Authentifizierungscodes erstellen, mit denen man sich anmelden kann. SMS sind auch verfügbar, sind aber etwas unsicherer, da sie keine spezielle Tokens verwenden, um die Codes auf dem Gerät selbst zu generieren. Die Passphrase gilt als zweites Passwort und wird vom Benutzer gesteuert. Im Grunde genommen ist der Austausch einer 2-FA nur umzusetzen

2.7.3.1.1. Google Authenticator

Musskriterium	Erfüllungsgrad
M01	Ja

Google Authenticator erfüllt alle Kriterien und ist somit als Vergleichskandidat angenommen.

2.7.3.1.2. Authy

Musskriterium	Erfüllungsgrad
M01	Ja

Authy trifft ebenfalls alle Kriterien und ist somit als Vergleichskandidat angenommen.

2.7.3.1.3. Passphrase

Musskriterium	Erfüllungsgrad
M01	Ja.

Passphrase erfüllt die Kriterien.

2.7.3.1.4. SMS

Musskriterium	Erfüllungsgrad
M01	Die Authentifizierungsmethode ist kostengünstig oder kostenlos.

SMS ist auch als Variante verfügbar, und kostet nur [\\$0.069 pro SMS](#) ausserhalb der Vereinigten Staaten. Die SMS-Variante ist an eine Telefonnummer gebunden, was den Sicherheitsstandard erfüllt, da die Nummer nicht in mehreren Geräten gleichzeitig verfügbar ist. Die neueste Sicherheitsforschung ermutigt uns jedoch, uns anderen Authentifizierungsmethoden zuzuwenden.

Variantenauswahl

Varianten	
V1	Google Auth
V2	Authy
V3	Passphrase

2.7.3.2. Nutzwertanalyse

2.7.3.2.1. Wunschkriterien

Wir haben diese Wunschkriterien zur Verfeinerung der Analyse ausgewählt:

- **W01:** Ein hoher Sicherheitsstandard ist gewährleistet
- **W02:** Multi-Device
- **W03:** Es ist nicht an spezielle Hardware gebunden
- **W04:** Backups sind verfügbar
- **W05:** Cloud-Access ist verfügbar
- **W06:** Der Implementationsaufwand ist gering
- **W07:** Es ist Kontounabhängig

Wir vergleichen die Varianten anhand dieser Kriterien und bestimmen anhand der höchst erreichten Punktzahl welche für die Two-Factor Authentication-Methode in Frage kommt.

2.7.3.2.2. Präferenzmatrix

Punktevergabe	
0	weniger wichtig
1	gleich wichtig
2	wichtiger

Kriterium	W01	W02	W03	W04	W05	W06	W07	Summe	Faktor
W01	2	2	2	2	2	2	2	12,000	29%
W02	0	2	1	1	1	2	7,000	17%	
W03	0	0	0	1	0	2	3,000	7%	
W04	0	1	2	1	0	1	5,000	12%	
W05	0	1	1	1	0	1	4,000	10%	
W06	0	1	2	2	2	2	9,000	21%	
W07	0	0	0	1	1	0	2,000	5%	
Summe								42,000	100%

Mit der Gewichtung kann die Nutzwertanalyse bestimmt werden.

Bewertung n	
0	Erfüllt das Kriterium nicht
1	Erfüllt das Kriterium nur teilweise oder mit Workaround (z.B. Zusatztool)
2	Erfüllt das Kriterium vollständig
3	Übertrifft das Kriterium (z.B. mit weiteren Funktionalitäten als definiert)

2SA		Google Authenticator		Authy		Passphrase	
Kriterium	Gewichtung g	n	g*n	n	g*n	n	g*n
W01	0,250	3	0,750	2	0,500	2	0,500
W02	0,125	1	0,125	2	0,250	3	0,375
W03	0,054	2	0,107	2	0,107	3	0,161
W04	0,089	1	0,089	2	0,179	0	0,000
W05	0,071	0	0,000	2	0,143	0	0,000
W06	0,161	2	0,321	2	0,321	3	0,482
W07	0,036	2	0,071	1	0,036	2	0,071
	0,79		1,464		1,536		1,589

Abbildung 12: Nutzwertanalyse Two-Step Authentication

2.7.3.3. Auswahl

Es wird Variante 3, die Passphrase, ausgewählt, da es die Kriterien am besten erfüllt und mit dieser der Implementationsaufwand am geringsten ist, was in einem stark zeitlich begrenzten und umfangreichen Projekt sehr gelegen kommt.

2.7.4. GUI

Das Graphical User Interface, GUI, ist von hoher Bedeutung, da es den Nutzern hilft, sich auf einfache Weise zurechtzufinden, mögliche Fehlerquellen von Seiten der Nutzer zu minimieren und das Streamlining von Befehlen zu ermöglichen. Die folgenden Kriterien sind:

Musskriterien	Begriff
M01	Das Framework ist den Entwicklern bekannt.
M02	Das Framework folgt dem MVC-Pattern.

2.7.4.1. Auswahl

Die folgenden UI-Frameworks wurden ausgewählt:

JavaFX, Swing und AWT. Alle drei Varianten erfüllen die Voraussetzungen, wobei angemerkt werden muss, dass AWT grösstenteils in Swing übernommen wurde.

2.7.4.2. JavaFX

Musskriterien	Erfüllungsgrad
M01	Erfüllt
M02	Erfüllt.

JavaFX ist mindestens einem der Entwickler bekannt und folgt dem MVC-Pattern.

2.7.4.3. Swing

Musskriterien	Erfüllungsgrad
M01	Erfüllt
M02	Erfüllt

Swing ist zwei Entwicklern bekannt und folgt dem MVC-Pattern.

2.7.4.4. AWT

Musskriterien	Erfüllungsgrad
M01	Das Framework ist den Entwicklern nur bedingt bekannt.
M02	Das Framework folgt nicht dem MVC-Pattern.

AWT erfüllt nur knapp das erste Muss-Kriterium, und verfehlt das zweite vollständig. Dies würde normalerweise zu einem sofortigen Ausschluss führen, aber es wird der Interesse Halber trotzdem weiter verglichen, obwohl es nur im unwahrscheinlichsten Fall den Score von 1 erreichen wird.

Variantenauswahl

Varianten
V1
V2
V3

2.7.4.5. Nutzwertanalyse

2.7.4.5.1. Wunschkriterien

Wir haben nachfolgende Wunschkriterien zur Verfeinerung der Analyse ausgewählt:

- **W01:** Das Framework ist noch immer maintained.
- **W02:** Das Framework verfügt über die nötigen UI-Features.
- **W03:** Das Framework verfügt über einen Screen-Builder.

Wir vergleichen die Varianten anhand dieser Kriterien und bestimmen dann anhand der höchst erreichten Punktzahl, welches GUI-Framework in Frage kommt.

2.7.4.5.2. Präferenzmatrix

Punktevergabe	
0	weniger wichtig
1	gleich wichtig
2	wichtiger

Präferenzmatrix					
Kriterium	W01	W02	W03	Summe	Faktor %
W01		2	1	3	50,00
W02	0		1	1	16,67
W03	1	1		2	33,33
Summe				6	1,00

Mit der Gewichtung kann die Nutzwertanalyse bestimmt werden.

Bewertung	
0	Erfüllt das Kriterium nicht
1	Erfüllt das Kriterium nur teilweise oder mit Workaround (z.B. Zusatztool)
2	Erfüllt das Kriterium vollständig
3	Übertrifft das Kriterium (z.B. mit weiteren Funktionalitäten als definiert)

UI-Framework		JavaFX		Swing		AWT	
Kriterium	Gewichtung g	n	g*n	n	g*n	n	g*n
W01	0,500	2	1,000	1	0,500	0	0,000
W02	0,167	2	0,333	2	0,333	2	0,333
W03	0,333	2	0,667	0	0,000	0	0,000
	1,00		2,000		0,833		0,333

Abbildung 13: Nutzwertanalyse UI-Framework

2.7.4.6. Auswahl

Variante 1, JavaFX hebt sich als klarer Sieger hervor, da dieses Framework die nötigen Features aufweist und mithilfe des Scene-Builders die Arbeit vereinfacht, was den potenziellen Zeitaufwand weiter reduziert. AWT wird gänzlich abgelehnt, da dies knapp die nötigen Features aufweist und aufgrund dessen Alters nicht mehr wirklich maintained wird bzw. in Swing aufgenommen wurde.

2.7.5. Morphologischer Kasten

Die Systemelemente und Eigenschaften von C3rBytes wurden ausgiebigen Evaluierungen unterzogen. Im nachfolgenden morphologischen Kasten sind die Optionen, sowie die effektive Auswahl (grün) markiert.

	Varianten					
Anwendungstyp	Desktop-App	WebApp	Hybrid			
Programmiersprache	Java					
Verschlüsselungsalgorithmen	AES	Blowfish	Twofish	DES	Serpent	3DES
Grafische Oberfläche	JavaFX	Swing	AWT			
Persistenz	Datei	RDB				
Hostsystem	OS (Workstation)	USB-Stick				

Tabelle 28: Morphologischer Kasten

Somit ist C3rBytes eine Standalone, bzw. eine Desktop-Anwendung, welche in Java geschrieben wurde. Aus kryptologischer Sicht soll der AES Algorithmus zum Einsatz kommen. Die grafische Oberfläche soll

mit JavaFX umgesetzt werden. Zudem wird die Persistierung mittels einer relationalen Datenbank erfolgen. Als Zusatz läuft die Anwendung wahlweise auf einem normalen Betriebssystem oder auf einem USB-Stick.

2.8. Technische Konzeption

2.8.1. Systemidee

Wir bauen ein Desktop-Programm, welches erlaubt Profile sicher zu speichern und zu verwalten. Die Nutzer sollen in der Lage sein, eine gesicherte Datenbank zu entschlüsseln und zu verschlüsseln. Die Datenbank ist mit einem Master-Passwort und einer Master-Passphrase in zwei Stufen gesichert. Der Nutzer kann Profile in der Datenbank anlegen, in denen man Nutzernamen, Passwörter, Profiltypen, URL's und Notizen anlegen kann. Diese Profile sind veränderbar und lösbar. Das Master-Passwort und die Master-Passphrase des Nutzerkontos können unabhängig verändert werden und das Konto kann gesamthaft gelöscht werden.

Das Programm verfügt über keine Möglichkeit mit dem Internet zu kommunizieren, ausser dem Öffnen von Webseiten. Somit ist eine Synchronisation über mehrere Geräte nicht möglich.

Des Weiteren wird kein Multi-Threading in Erwägung gezogen. Es würde eine Komplexität einfügen, die unser Projekt nur unnötig aufblasen würde. Die Komplexität würde stark steigen und der Performanzgewinn würde, wenn überhaupt, nur gering sein. Im "worst case" würde die Performanz sogar darunter leiden. Ebenso verhält es sich mit der Normalisierung der Datenbank. Diese wird nicht normalisiert. Nähere Erläuterungen dazu finden Sie unter Kapitel 2.18.

2.8.2. Coderichtlinien

Um zu garantieren, dass der Code korrekt und sauber programmiert wird, und um die Wartbarkeit zu erhöhen, haben wir uns auf die folgenden Coderichtlinien geeinigt und halten diese im Verlauf der Entwicklung ein.

2.8.3. Allgemeine Richtlinien

- Der Code wird in Englisch geschrieben.
- Jede Datei enthält nur eine Klasse oder Interface, d.h. keine Nested Classes.
- Auskommentierter Code wird gelöscht.

2.8.4. Packages

- Gruppen von Klassen werden in separaten Packages unterteilt.

2.8.5. Klassen / Interfaces

- Klassen- und Interfacenamen beginnen mit Grossbuchstaben. Camel-Case wird für Klassen- und Interfacenamen verwendet. Interfaces beginnen mit dem Grossbuchstaben I.
- Ausnahme bilden die View-Controller. Diese beginnen mit einem Kleinbuchstaben.

2.8.6. Methoden

- Methodennamen beginnen mit Kleinbuchstaben. Camel-Case wird für Methodennamen verwendet.
- Methodennamen sollen eindeutig gewählt werden, sodass deren Zweck einfach ersichtlich ist.
- Code muss wo nötig mit JavaDoc dokumentiert werden.

2.8.7. Variablen

- Variablennamen beginnen mit Kleinbuchstaben. Camel-Case wird für Variablennamen verwendet.
- Variablennamen sollen eindeutig gewählt werden, sodass keine Verwechslung entstehen kann.

2.9. Architekturpattern

Bei der Architektur von C3rBytes wird weitestgehend auf das Model-View-Controller Pattern (MVC-Pattern) gesetzt.

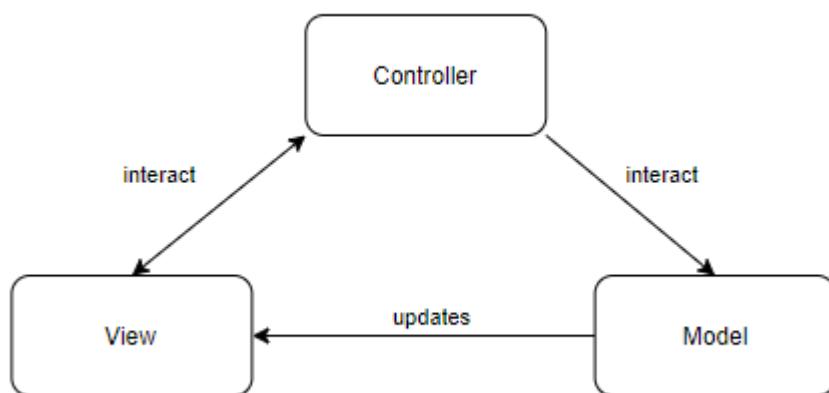


Abbildung 14: Architektur-Pattern MVC

Die View enthält deshalb keine Logik, sondern besteht lediglich aus der Benutzeroberfläche. Im Grunde genommen bestehen die Views allesamt aus den FXML-Dateien. Die View gibt die Benutzereingaben an die Controller weiter.

Die Controller erhalten von den Views die Benutzereingaben. Die Controller kommunizieren je nach Aktion mit den entsprechenden Packages, wie z.B. Crypto, wenn um Verschlüsselungsfragen geht oder dem DAO, sofern die Datenpersistenz oder abgerufen werden sollen.

Das Model stellt unsere Daten dar und erhält von Controllern den Input und gibt je nach Aktion den Inhalt der Daten an die View weiter. Diese aktualisiert sich dementsprechend und der Kreislauf schliesst sich.

Die nachfolgenden Architekturdokumente wurden laufend aktualisiert und spiegeln das Endresultat wider.

2.10. Kontextsicht

Der Kontext von C3rBytes besteht aus dem System selber, welches zur Persistierung an eine relationale Datenbank angebunden ist.

C3rBytes wird auf einem Hardwaresystem, wie z.B. einem USB-Drive oder anderen Speichermedien ausgeliefert, welche aber noch ein Betriebssystem benötigen, indem das Speichermedium gemounted bzw. eingebunden werden kann.

C3rBytes kann alternativ auch direkt auf dem Betriebssystem installiert werden.

Auf dem Betriebssystem muss zwingend eine JVM laufen.

Zum User als eigentlicher Nutzniesser besteht ebenfalls eine grafische Schnittstelle.

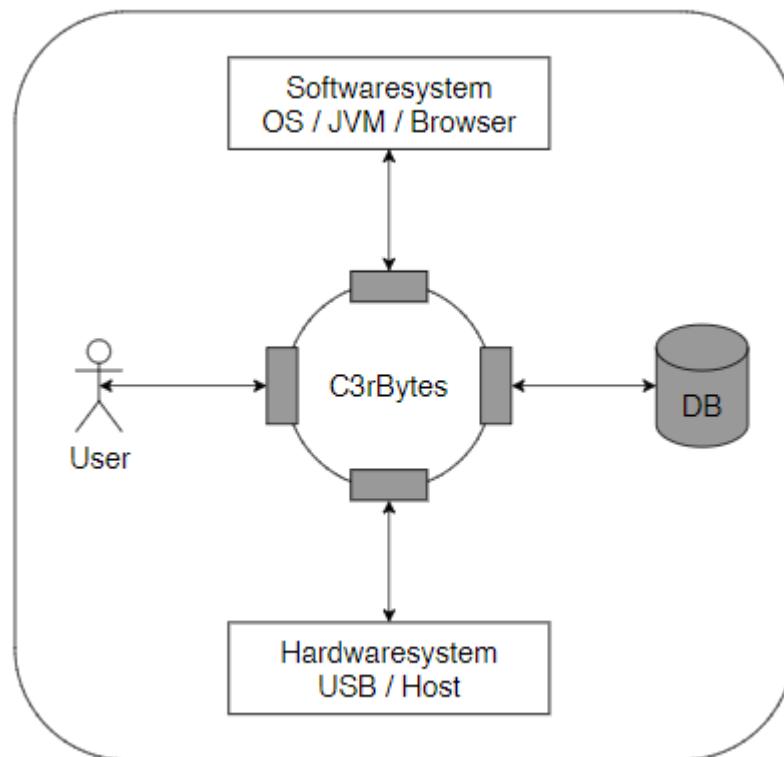


Abbildung 15: Kontextdiagramm



2.11. Domänenmodel

Als ein zentrales Element im Domänenmodel haben wir den User identifiziert, welcher ein Master-Passwort und eine Master-Passphrase besitzt bzw. setzt und mit der grafischen Benutzerschnittstelle interagiert.

Die grafische Benutzerschnittstelle interagiert je nach Nutzereingaben mit dem Crypto Package und/oder der relationalen Datenbank.

Das Master-Passwort verschlüsselt und entschlüsselt die relationale Datenbank.

Die relationale Datenbank beinhaltet die ganzen Profile.

Die Profile bestehen ihrerseits aus den Elementen User name, Password, Type, URL, Notes und Attributes.

Die Master-Passphrase wird benutzt, um ein vom Passwort-Generator erzeugtes Passwort zu verschlüsseln oder zu entschlüsseln.

Dieses generierte Passwort vorgegebener Länge sowie vorgegebenem Zeichensatz wird in eine Datei geschrieben.

Mit diesem generierten Passwort lassen sich wiederum die Passwörter der Profile verschlüsseln oder entschlüsseln.

Das nachfolgende Domänenmodel soll die zentralen Elemente sowie das fachliche Zusammenspiel und deren Kardinalität untereinander aufzeigen.

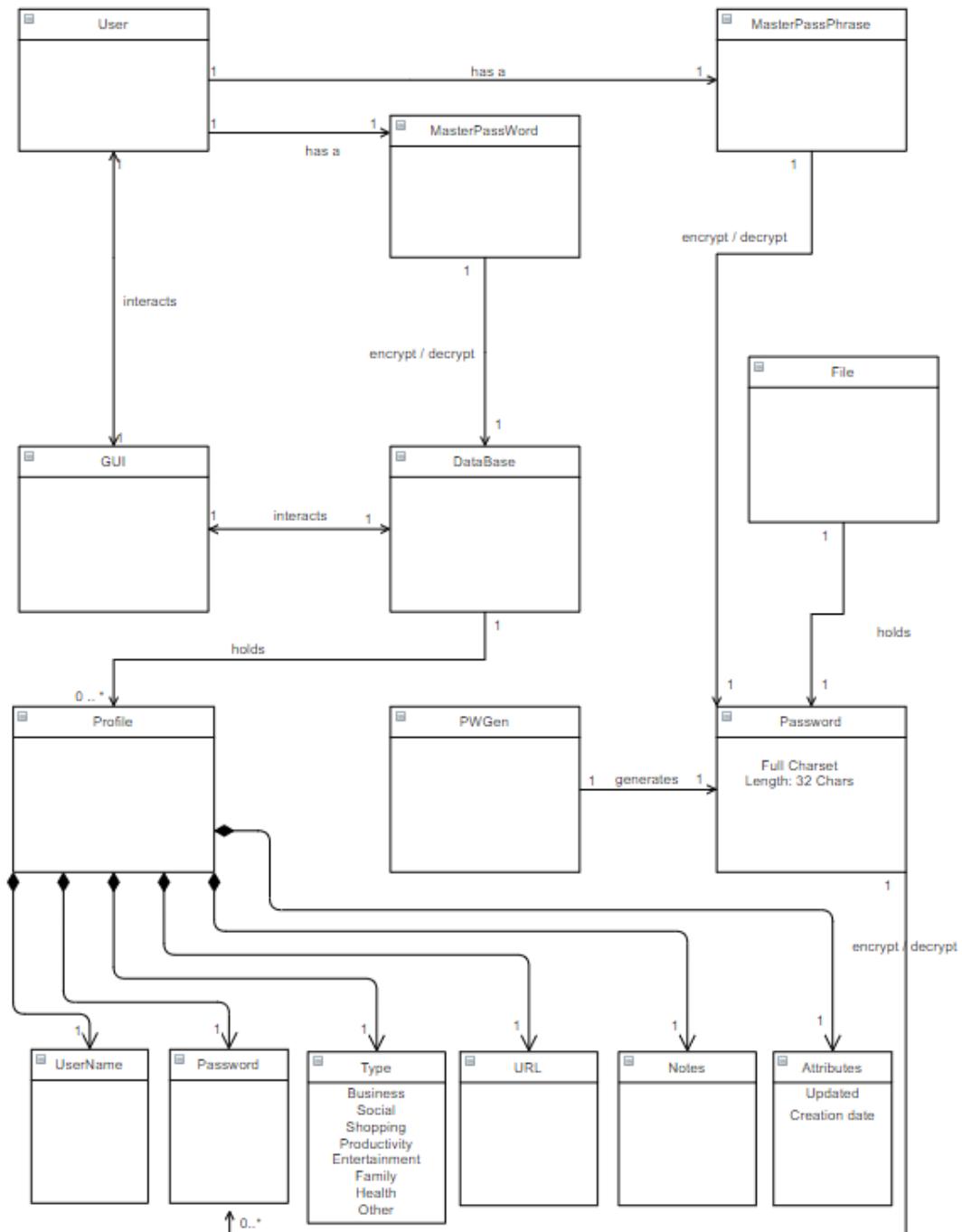


Abbildung 16: Domänenmodell

2.12. Komponentensicht

In nachstehender Abbildung sind die Pakete (Packages) sowie deren Komponenten dargestellt, welche zur Strukturierung der Software von Bedeutung ist.

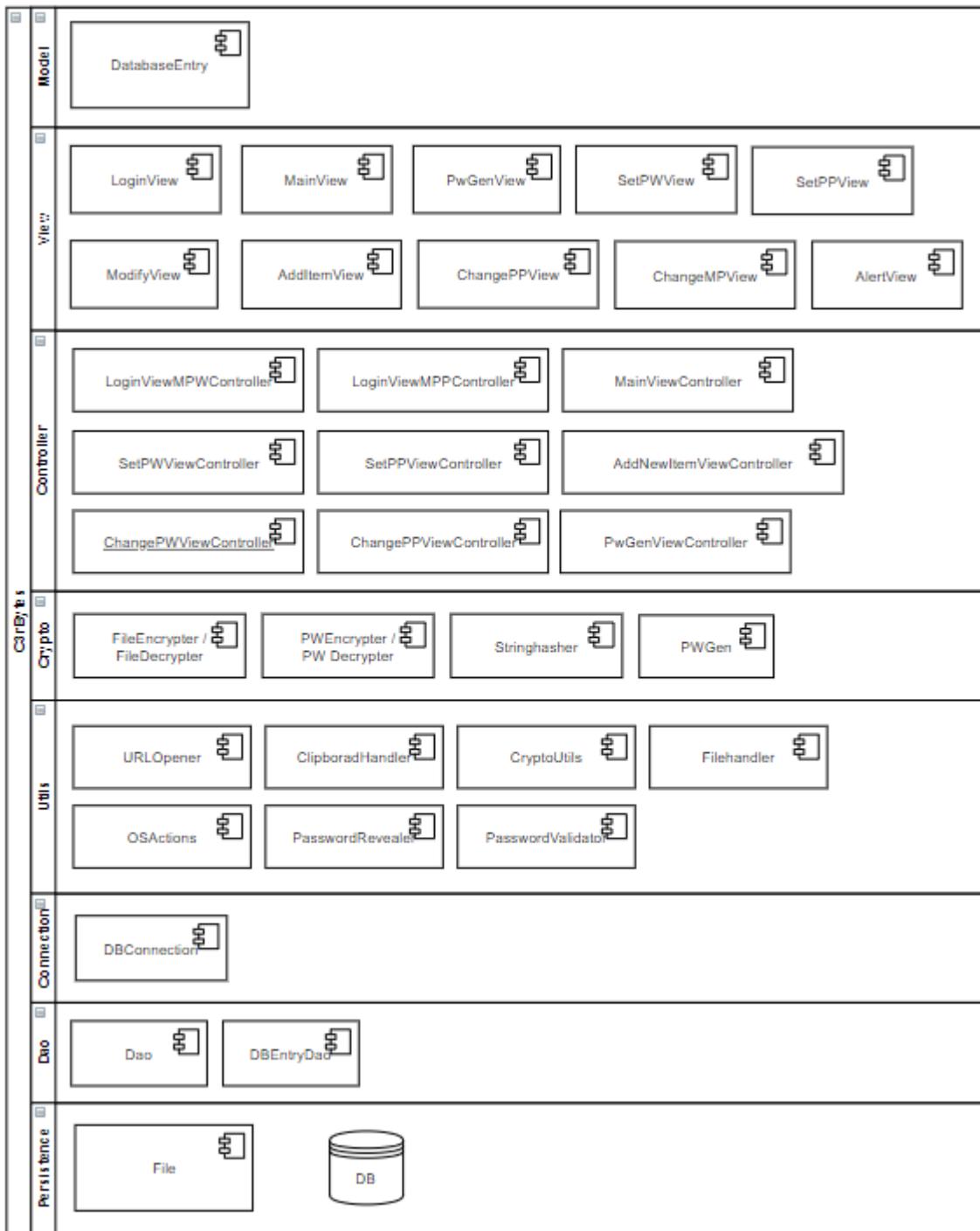


Abbildung 17: Komponentendiagramm

2.13. Schnittstellenbeschreibung

2.13.1. Schnittstelle Datenbankeintrag

Jeder Eintrag der Datenbank besteht aus sieben Werten:

- ID (user_id): Dies ist die numerische und eindeutige ID des Eintrags. Sie wird automatisch bei der Erstellung des Eintrags generiert und wird als Integer gespeichert.
- User Name (username): Der Nutzernname des Profils für die definierte Software oder Website, es wird als Varchar gespeichert.
- Password (password_text): Dies ist das Passwort des Profils für die definierte Software oder Website. Dieses ist normalerweise verschleiert und kann nach dem Entschlüsseln der Datenbank in die Zwischenablage kopiert werden. Das Passwort ist ein Varchar und kann rein aus Buchstaben oder auch alphanumerisch mit Sonderzeichen bestehen.
- Type (description): Profile sind zwischen Sozialen, Shopping-, Business- und privaten Profilen etc. unterteilt, es wird als Varchar gespeichert.
- URL (url_content): Dies enthält die URL der jeweiligen Website und ist optional. Es wird als Varchar gespeichert.
- Note (note): Dies enthält Notizen, die der Nutzer für dieses Profil erstellt hat und sind optional. Es wird als CLOB (Character Large Object) gespeichert.
- Date created (date_creation): Dies ist das Datum mit Zeitangaben, das dem Nutzer zeigt, wann das Profil erstellt wurde, es wird als Varchar gespeichert.
- Last Updated (date_update): Dies ist ein Datum mit Zeitangaben, das dem Nutzer zeigt, wann das Profil das letzte Mal aktualisiert wurde, es wird als Varchar gespeichert.

Eintrag erstellen

- Um einen Eintrag zu generieren, wird eine View erstellt, die mithilfe eines Controllers die oben genannten Werten erstellt. Der Nutzer gibt die Werte ein und speichert dann den Eintrag als Objekt in der Datenbank. Das Passwort kann entweder direkt eingegeben werden oder wird automatisch basierend auf verschiedenen Parametern in einem neuen Fenster generiert und so im Eintrag gespeichert.

Item View – Generator View

- Passwort – Ein alphanumerischer String mit Sonderzeichen kann generiert werden, wenn die Password Generator View aufgerufen wird. Das Passwort wird dann entweder vom Nutzer gespeichert oder verworfen und das Fenster wird geschlossen und man gelangt zurück zur Add Item View.

2.14. Klassenmodel

Das Klassenmodel unter Punkt Gesamtstruktur stellt die Hauptkomponenten sowie die Interaktionen untereinander in groben Zügen dar. Die Klassen sind nach den Prinzipien von geringer Kopplung und höher Kohäsion entwickelt worden. Des Weiteren wurde auf Information Hiding gesetzt und auf kleine und saubere Schnittstellen geachtet. Zudem haben wir Wert daraufgelegt, dass der Scope der Variablen, welche auf Objekte zeigen, so klein wie möglich und so gross wie nötig ist.

Detaillierte Informationen zu den Klassen, ihren Feldern und Methoden erhalten Sie in den nachfolgenden Unterkapiteln sowie in der JavaDoc-Dokumentation.

2.14.1. Gesamtstruktur

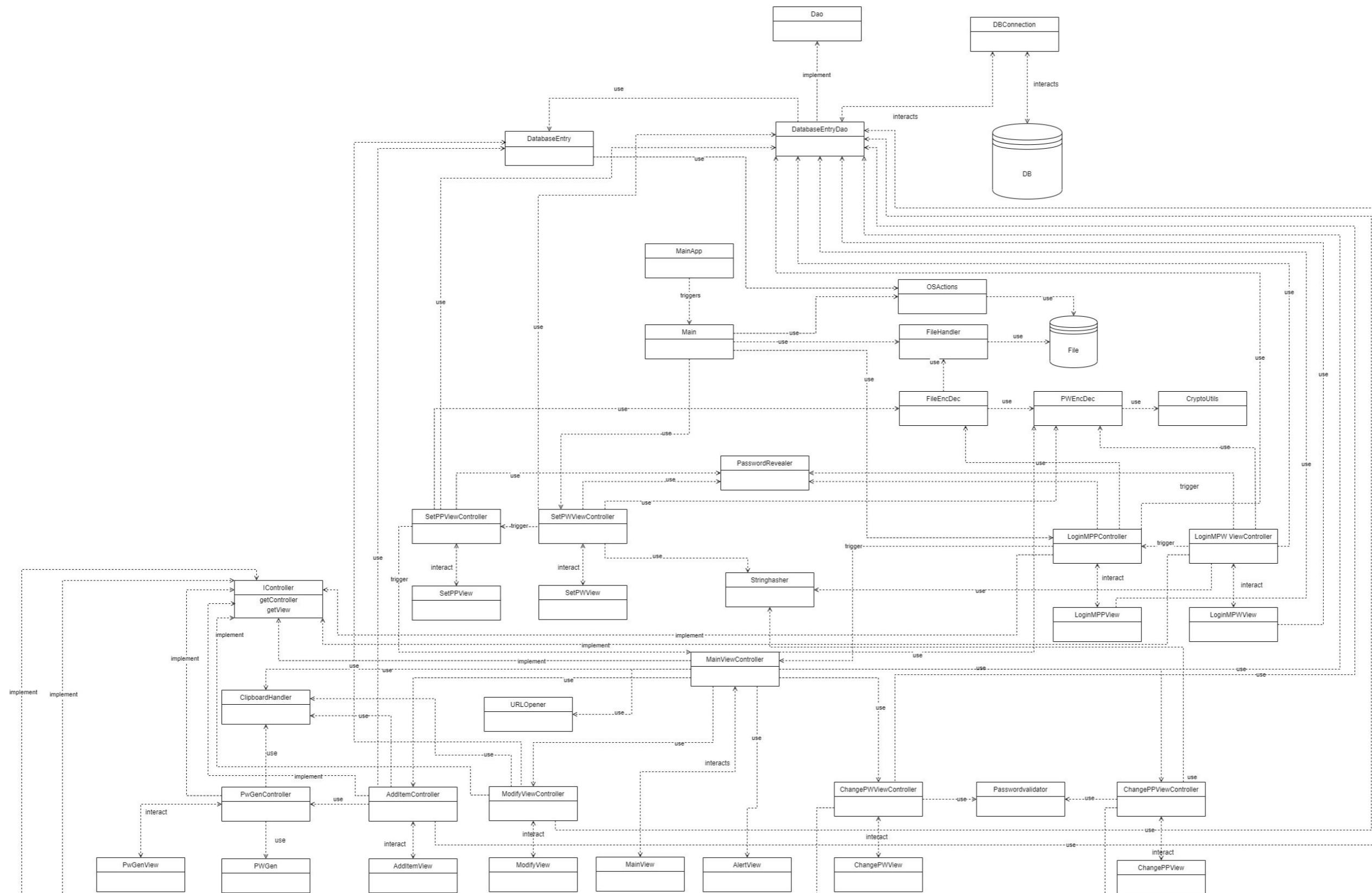


Abbildung 18: Klassendiagramm Gesamtstruktur

2.14.2. Packages

Die aus dem Domänen- und Komponentenmodell gewonnenen Erkenntnisse spiegeln sich in den Paketen des Source-Codes wider.



Abbildung 19: Package-Struktur

2.14.3. Main

Das Package Main besteht aus zwei Klassen.

Bei der Paketierung von JavaFX-Applikationen im Zusammenspiel mit der Javaversion 11.0.6 gab es Probleme in Form von Nullpointern. Die JavaFX-Loader, sowie die FXML-Ressourcen konnten nicht gefunden werden.

Aus diesem Grund mussten wir mit einem Workaraound arbeiten, indem wir die Klasse MainApp hinzugefügt haben. Der Eintrittspunkt in das Programm führt über die Main-Methode der MainApp-Klasse. Diese Main-Methode zeigt im Wesentlichen nur auf die Main-Methode der Main-Klasse. Die Main-Methode startet ihrerseits die Methode start, welche die grafische Benutzeroberfläche lädt und anzeigt.

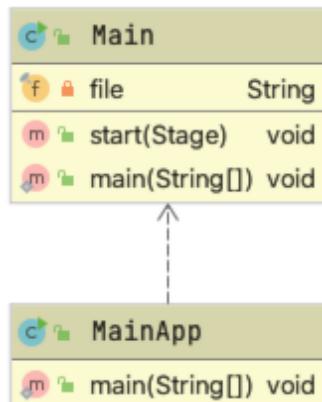
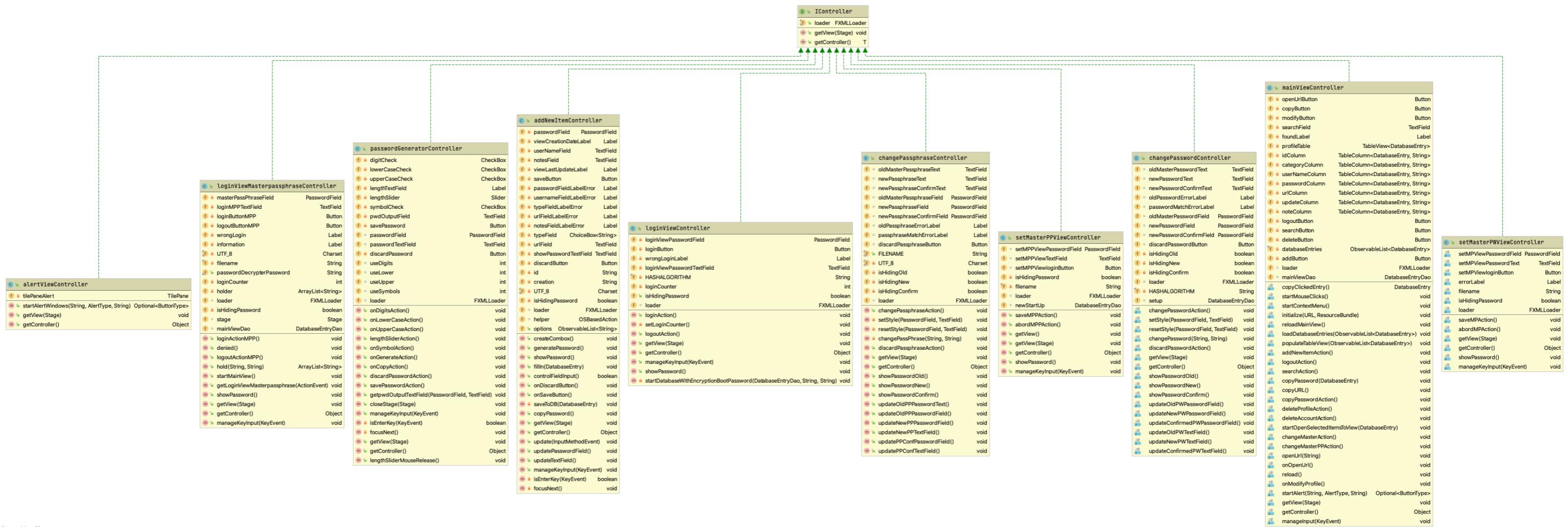


Abbildung 20: Klassenmodell
Main-Package

2.14.1. Controller

Die Controller nehmen die Benutzereingaben entgegen, welche meist in Form von Events (Mausklick- oder Tastatureingaben) auftreten.

Aufgrund besserer Wartbarkeit hat jede View ihren eigenen Controller. Somit kann bei Bedarf jeder Controller und jede View individuell angepasst werden. Alle Controller implementieren zudem das Interface `IController`, was bedeutet, dass jeder Controller seine View in Form seiner FXML-Ressource und seinen Controller zur Verfügung stellt. Somit können andere Klassen die View sowie den Controller für weitere Verarbeitungsschritte abrufen und nutzen. Weiterhin wäre es zur besseren Entkopplung in Zukunft relativ einfach möglich einen `ViewManager` einzubinden, welcher die gewünschte View und/oder den gewünschten Controller zur Verfügung stellen könnte.



Powered by yhris

2.14.2. Crypto

Das Package Crypto ist, wie der Name vermuten lässt, für die Entschlüsselung und die Verschlüsselung von Inhalten zuständig. Das Package beinhaltet nachfolgende Klassen.

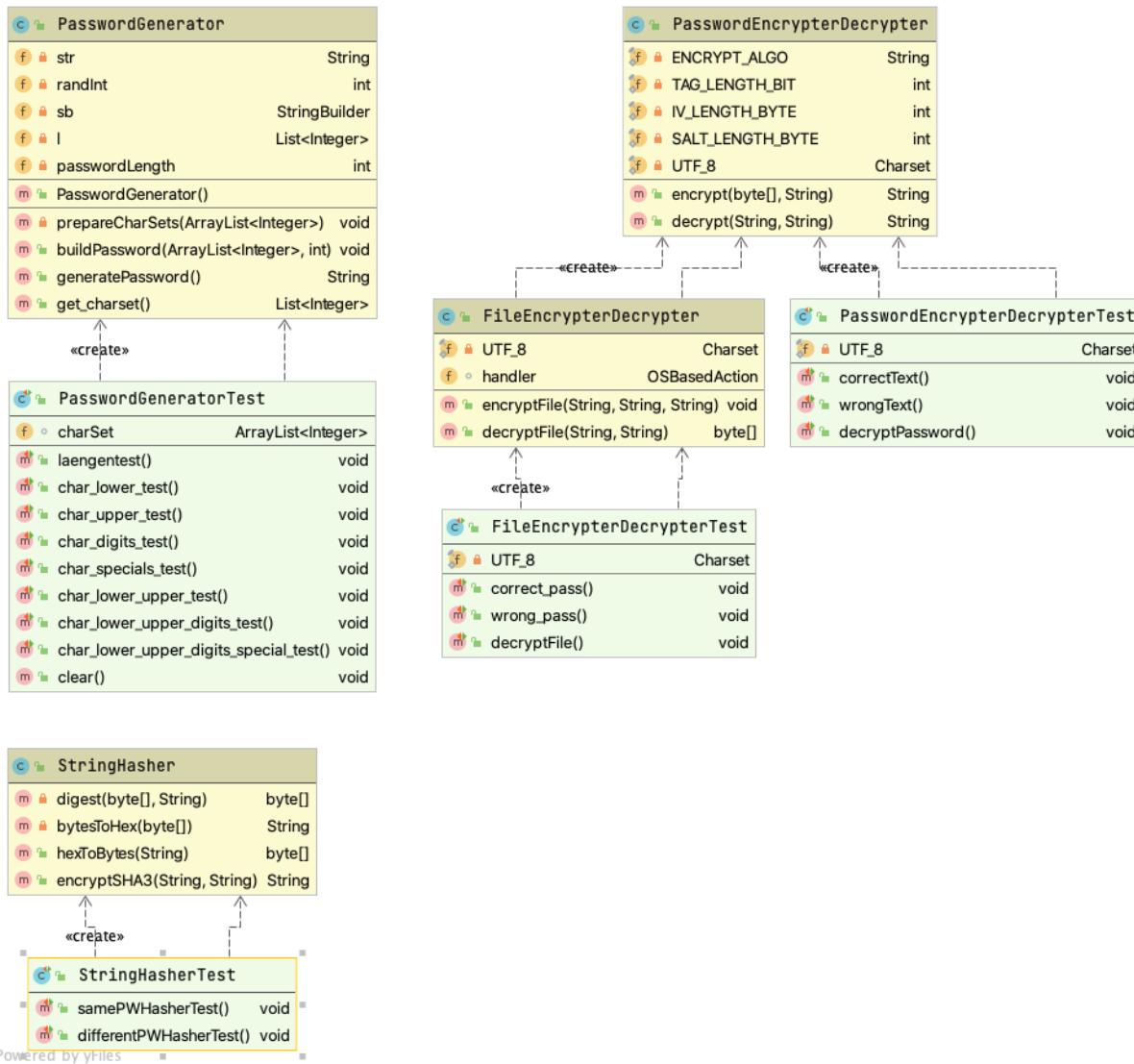


Abbildung 23: Klassenmodell Crypto-Package

Der FileEncrypterDecrypter entschlüsselt und verschlüsselt in Kombination mit dem PasswordEncrypterDecrypter ein vom PasswordGenerator generiertes Passwort mit der Passphrase, welcher der Nutzer beim initialen Login gesetzt hat und speichert es in einem File. Diese Klassen kommen ebenso beim Setzen und Einloggen der Master-Passphrase zum Einsatz.

Zudem steht ein Stringhasher (SHA3-512) bereit um das Master-Passwort zu hashen, welches zur Verschlüsselung der Datenbank genutzt wird. Diese Klasse wird vorwiegend genutzt, wenn es sich um Funktionen für das Master-Passwort wie das Setzen, Ändern und ebenso für das Login handelt.

2.14.3. Utils

Das Utils-Package beinhaltet Hilfsklassen, welche anderen Klassen zur Verfügung stehen, damit diese ihre Aufgabe erfüllen können.

Im Folgenden sind die Klassen schematisch dargestellt.

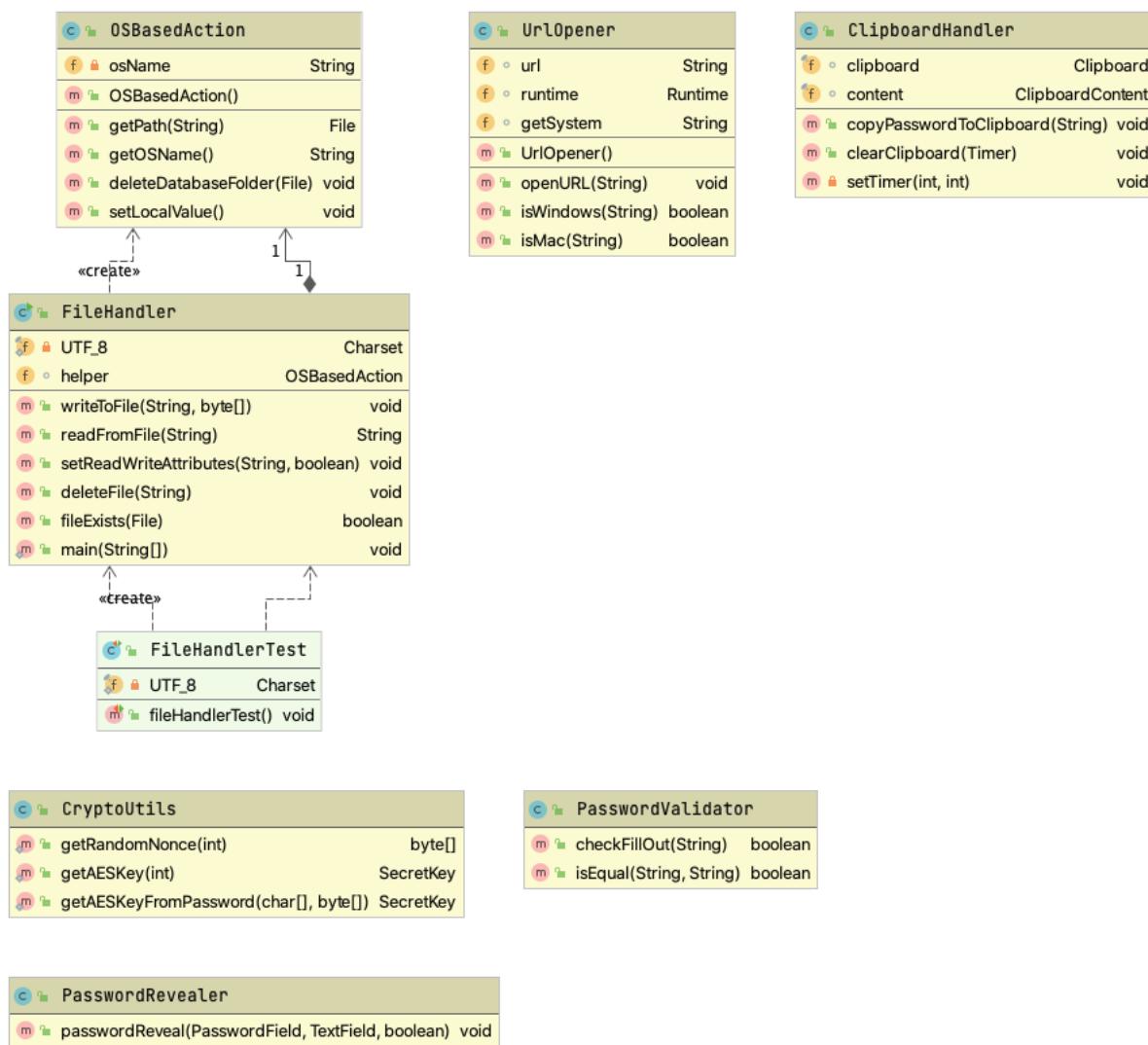


Abbildung 24: Klassenmodell Utils-Package

Die Klasse OSBasedAction stellt Methoden zur Verfügung, welche mit dem Betriebssystem interagieren, beispielsweise das Löschen einer Datei.

Der URL-Opener öffnet die selektierte URL im Standardbrowser des Betriebssystems, sofern diese den Vorgaben entspricht. (www.testurl.ch)

Mit dem ClipboardHandler lässt sich das selektierte Passwort für zehn Sekunden in die Zwischenablage kopieren. Nach Ablauf der zehn Sekunden wird der Zwischenspeicher wieder geleert und steht für den Einfüge-Vorgang nicht mehr zur Verfügung.

Der Filehandler schreibt, liest und ändert Attribute der Dateien.

Die CryptoUtils stellen für den PasswordEncrypterDecrypter diverse Methoden zur Verfügung. Die getRandomNonce liefert mit dem "salt" und dem "iv" eine sichere und zufällige Byte-Folge. Die getAESKeyFromPassword-Methode liefert einen geheimen Schlüssel, welcher vom Password abgeleitet wird. Beide Methoden werden benötigt, um den AES-Algorithmus korrekt und sicher zu implementieren.

Der PasswordValidator überprüft, ob die übergebenen Strings valide sind. Konkret werden Methoden angeboten, die prüfen, ob die Strings nicht leer sind und ob sie identisch sind. Diese Klasse wird von den Controllern benutzt, um zu überprüfen, ob beim Wechseln eines Passworts die fraglichen Felder nicht leer bzw. identisch sind.

Der PasswordRevealer implementiert die Funktion zum Anzeigen und Verbergen von Passwortfeldern.

2.14.4. DAO

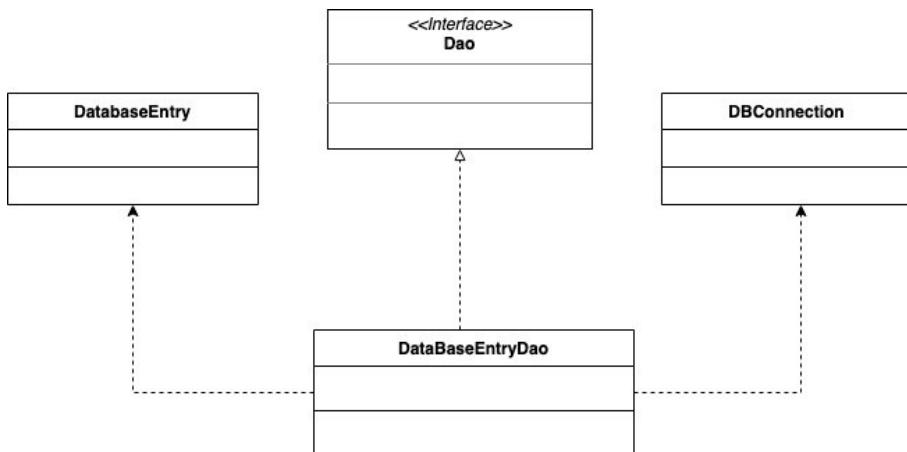


Abbildung 25: Klassenmodell DAO

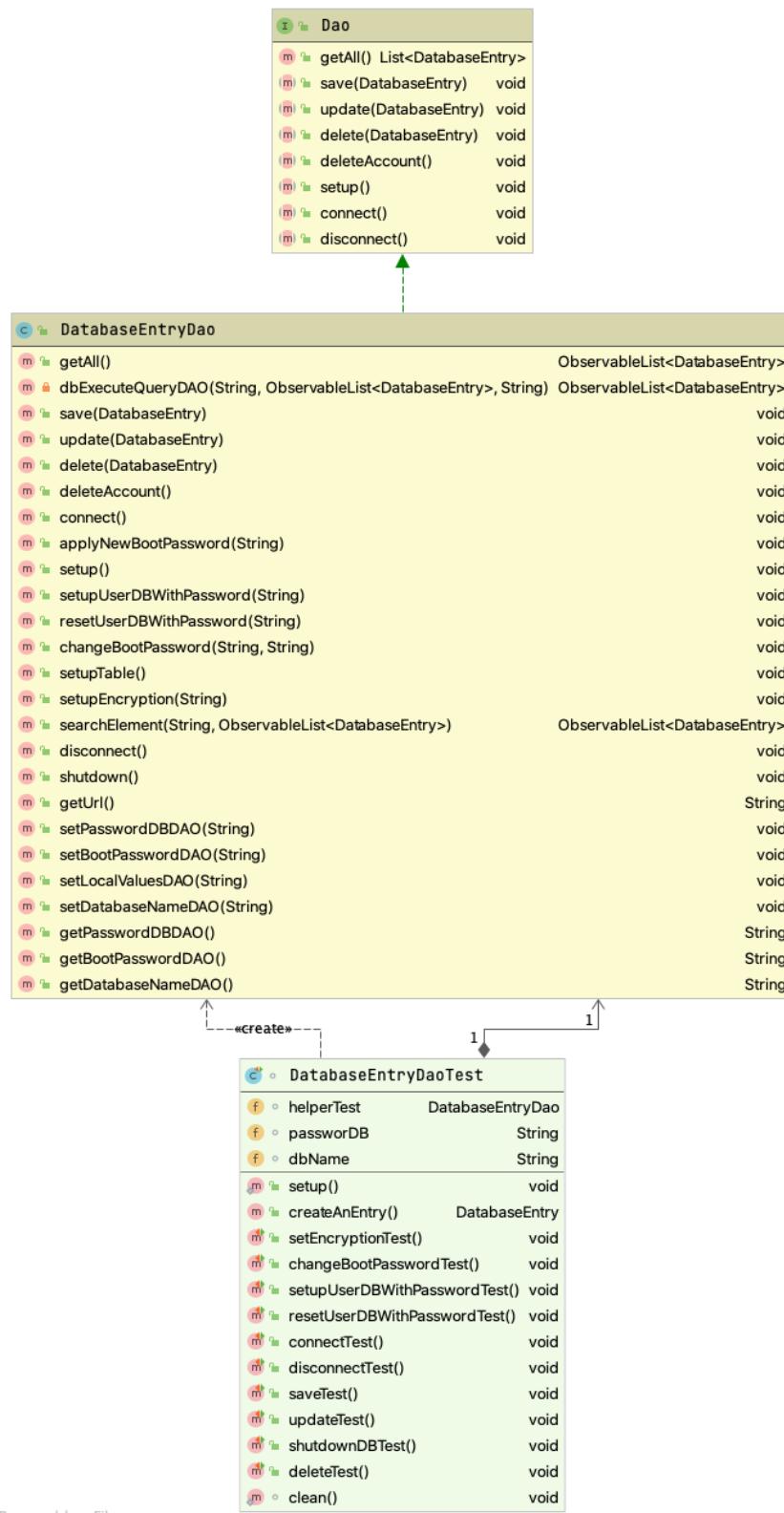
Die Kommunikation mit der Datenbank erfolgt über das Data Access Object (DAO) Pattern. Das DAO-Pattern sorgt dafür, dass die Komponenten sauber voneinander entkoppelt sind.

Wir haben eine DatabaseEntry-Klasse, welche ein einfacher Container für Benutzerdaten ist.

Aufgrund des hohen Abstraktionsniveaus, dass das Interface (Dao) bietet, ist es einfach, eine konkrete, feingranulare Implementierung zu erstellen, die mit DatabaseEntry-Objekten arbeitet.

Die Klasse DatabaseEntryDao implementiert die gesamten Funktionalitäten, die zum Abrufen, Aktualisieren und Entfernen von DatabaseEntry-Objekten erforderlich sind.

Sowohl die DatabaseEntry- als auch die DatabaseEntryDao-Klasse existieren innerhalb derselben Anwendung unabhängig voneinander. Die wichtigste Facette dieses Prozesses ist, wie DatabaseEntryDao alle Low-Level-Details, welche die Objekte persistiert, aktualisiert und löscht, vor der Applikation verbirgt. Nur das DAO kommuniziert mit der DBConnection-Klasse, die für die Kommunikation mit der Datenbank (und die Persistenz) verantwortlich ist.



Powered by yFiles

Abbildung 26: Klassenmodell DatabaseEntryDao

2.14.5. DatabaseEntry

Die Daten, die in der Datenbank gespeichert werden oder aus der Datenbank abgefragt werden, werden in Objekte injiziert, die durch den Konstruktor der Klasse DatabaseEntry.java erzeugt werden. Letztere hat auch eine ganze Reihe von Methoden, um ihre Instanzvariablen zu extrahieren.

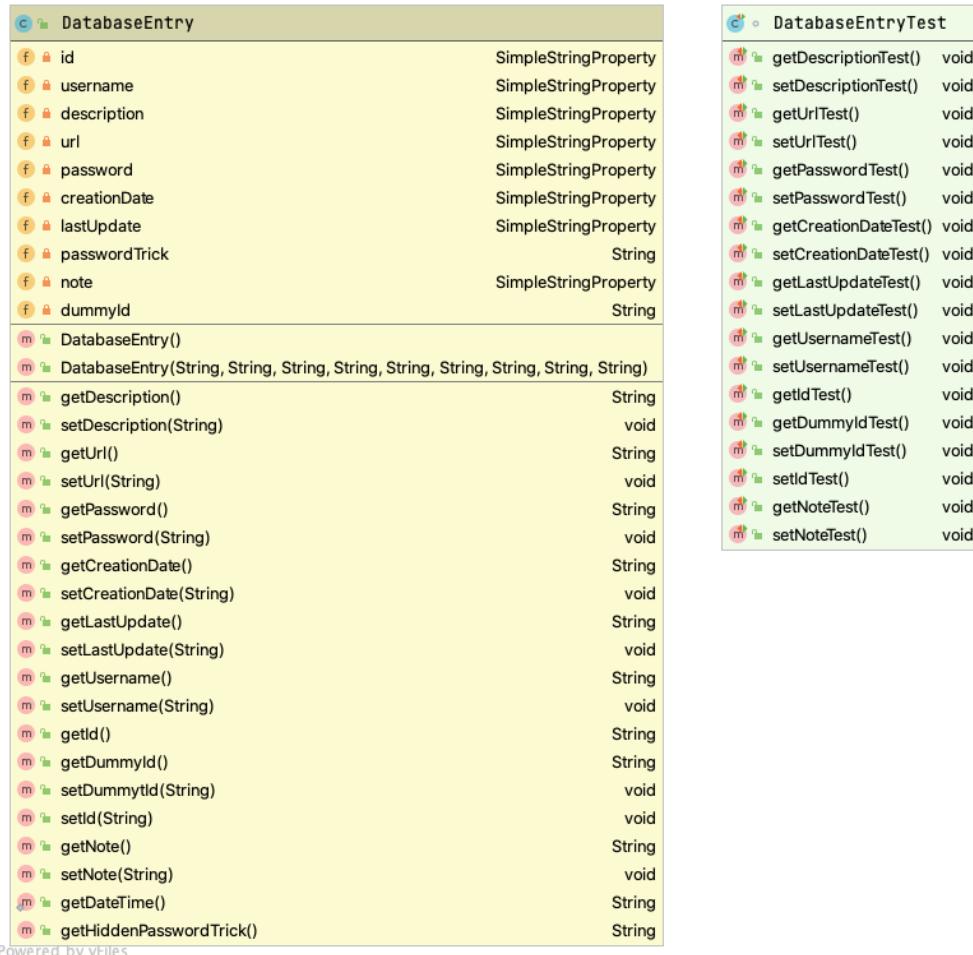


Abbildung 27: Klassenmodell DatabaseEntry

2.14.6. Connection

Die Klasse DBConnection.java enthält alle Felder für eine erfolgreiche Verbindung mit der Derby-Datenbank (oder anderen). Es unterstützt die Funktionen zum Verbinden und Trennen der Verbindung, aber auch die Verwaltung der Verbindungselemente (Änderung des Passworts, Änderung des Bootpassworts). Sie tauscht sich mit der Datenbank aus, um Informationen zu erfassen, abzufragen und zu suchen. Alle Informationen, die es außerhalb der Datenbank erhält, stammen von der Klasse DatabaseEntryDao.java. Anschließend überträgt sie die Ergebnisse (von Anfragen, Suchen usw.) an die Klasse DatabaseEntryDao.java.

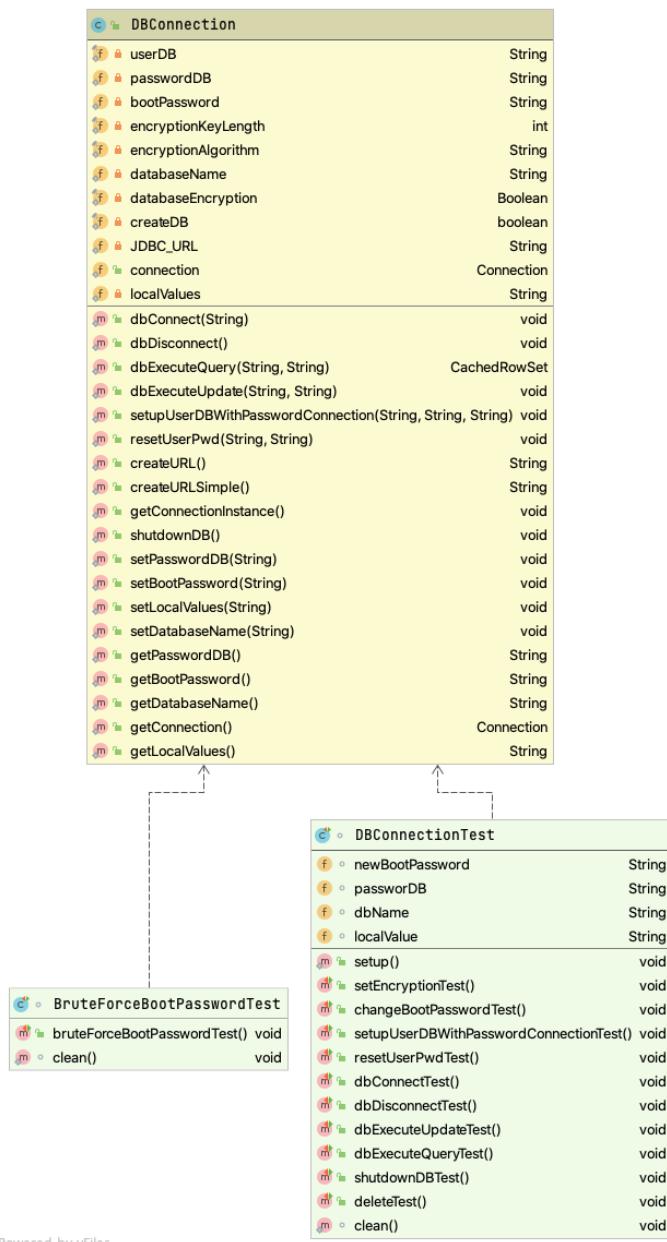


Abbildung 28: Klassenmodell DBConnection

2.15. Aktivitätsdiagramm

2.15.1. Zugriffssicherheit

Die Zugriffssicherheit ist das Herzstück der Applikation. Mit der Sicherheit des Logins steht oder fällt die Sicherheit des Passwort-Manager. Die Zugriffssicherheit ist in zwei Stufen aufgebaut. Die nachfolgende Abbildung veranschaulicht die Zugriffsmodalitäten.

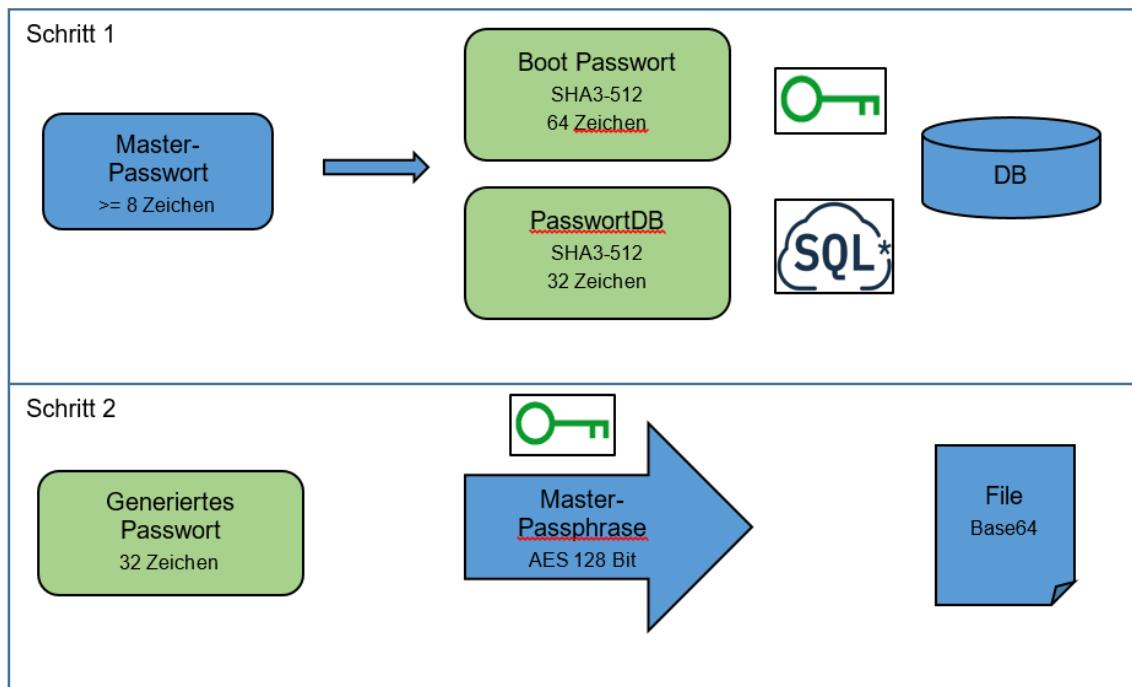


Abbildung 29: Zwei-Schritt-Zugriffssicherheit

Bei der Anmeldung muss als erstes ein Masterpassword gesetzt bzw. eingegeben werden. Dieses umfasst mindestens acht Zeichen. Die Wahl der Zeichen steht dem Nutzer frei. Es empfiehlt sich jedoch, ein komplexeres Passwort (Gross- und Kleinbuchstaben, Ziffern von 0 – 9 sowie Sonderzeichen aller Art) von hinreichender Länge (je länger desto besser der Schutz) zu setzen. Im Hintergrund wird die Datenbank mit dem gehaschten (SHA3-512) Master-Passwort mit AES 256-Bit verschlüsselt. Das gehashte Bootpasswort hat somit eine Länge von 64 Zeichen.

Damit der Nutzer Anfragen an die Datenbank anbringen kann, muss er sich vorgängig mit einem User-Passwort, dem PasswortDB, authentifizieren. Dieses User-Passwort wird vom Master-Passwort abgeleitet. Der Hash des Master-Passwortes wird noch einmal gehasht. Für das User-Passwort gelangen lediglich die letzten 32 Zeichen zur Anwendung. Von dieser Sequenz bekommt der Nutzer allerdings nichts mit. Es ist ein verborgenes Sicherheitsfeature von C3rBytes.

Anschliessend kann der Nutzer eine Master-Passphrase setzen bzw. eingeben. Wie der Name vermuten lässt, sollte es sich dabei um einen komplexen Satz bzw. Teilstücken von Buchstaben davon handeln. Die Master-Passphrase ist nicht obligatorisch, es existieren demnach keine Restriktionen. C3rBytes ist so aufgebaut, dass auch ohne die Eingabe einer Master-Passphrase das Passwort verschlüsselt im File abgelegt wird. Ob mit oder ohne Eingabe einer Master-Passphrase kommt zur Erhöhung der Sicherheit ein zufällig generierter «salt» sowie ein Initialisierungsvektor "iv" zur Anwendung. Es wird dennoch dringlichst empfohlen einen möglichst komplexen Satz zur Verschlüsselung zu verwenden.



Sobald die Master-Passphrase gesetzt ist, wird C3rBytes ein komplexes Passwort mit 32 Zeichen generieren, welches mit der Master-Passphrase verschlüsselt und in einem versteckten, schreibgeschützten File abgespeichert wird. Dieses generierte Passwort verschlüsselt alle Passwörter der Profile, welche später in C3rBytes eingetragen werden. Der Nutzer hat keine Kenntnis des generierten Passworts in der Datei. Er kennt lediglich seine Master-Passphrase, welche das Password verschlüsseln und entschlüsseln kann.

Sollte jemand trotzdem die erste Barriere durchbrechen, indem die Datenbank (mit dem Hash des Boot- und Userpasswordes) entschlüsselt wird, gibt es eine weitere Hürde. Die Passwörter jedes Profils in der Datenbank sind ebenfalls verschlüsselt und können nur durch den Schlüssel in der Datei entschlüsselt werden, und dieser ist durch die Master-Passphrase geschützt. Somit sind die Passwörter der Profile mindestens doppelt geschützt.

2.15.2. Registrierung

Das Aktivitätsdiagramm zeigt den erstmaligen Zugriff auf C3rbytes auf und entspricht den Schilderungen im Kapitel Zugriffssicherheit.

Es sind die Schritte, welche durchlaufen müssen, um die Software soweit aufzusetzen, damit alle Settings erstellt werden und das Tool in den betriebsbereiten Zustand übergeht.

Sofern diese Schritte ordnungsgemäss durchlaufen worden sind, ist C3rbytes bereit, um erste Einträge zu speichern. Falls nicht, wird C3rBytes alle Schritte zurücksetzen, damit die Registrierung in einem nächsten Anlauf sauber ausgeführt werden kann.

Anmerkung: Diese Aktivitäten werden ebenfalls durchlaufen, falls man seinen Masteraccount gelöscht hat und einen Neuen anlegen möchte.

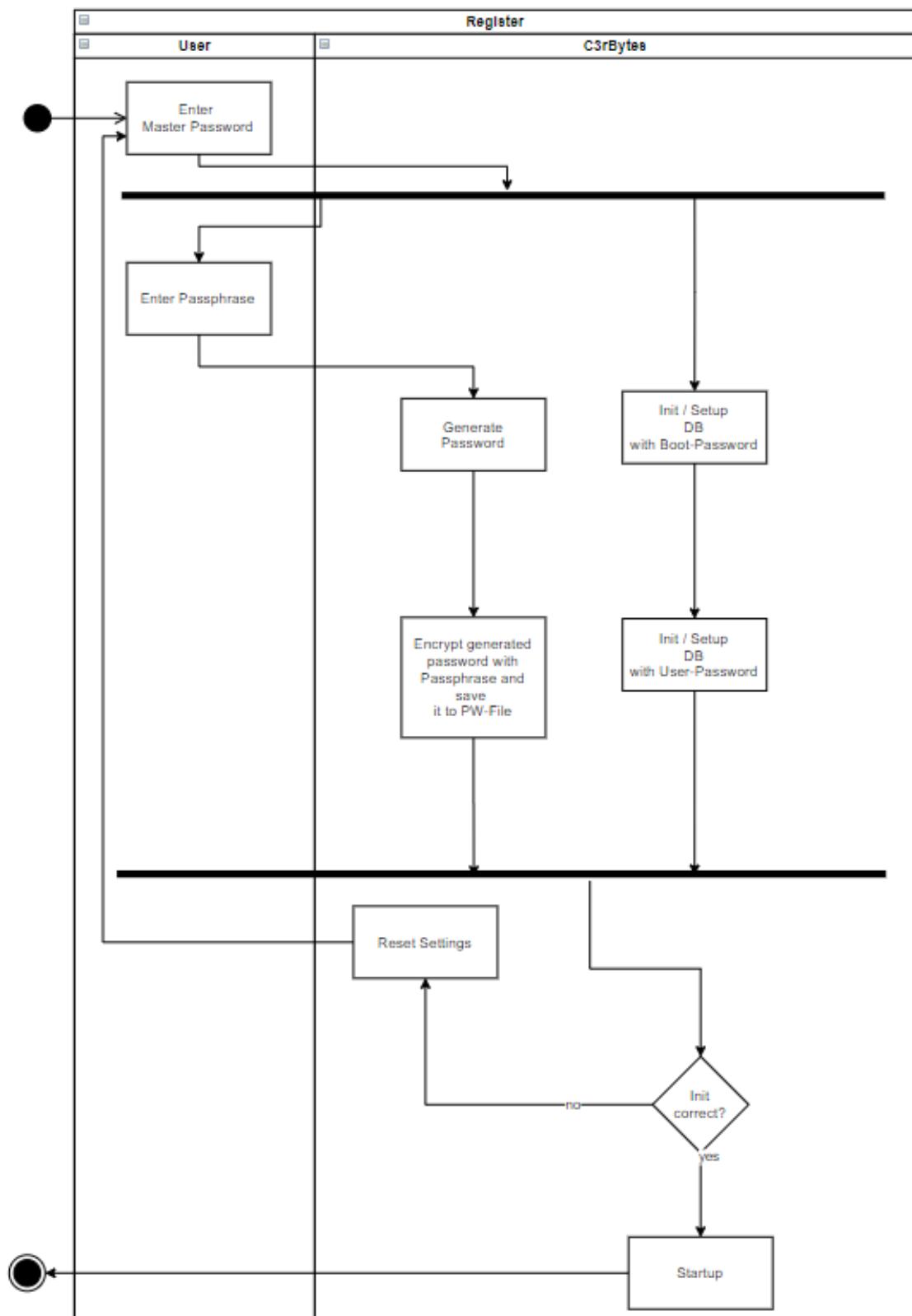


Abbildung 30: Aktivitätsdiagramm Registrierung

2.15.3. Login

Nach erfolgreicher Registrierung erfolgen die regulären Login-Abläufe.

Es handelt sich um die Schritte, welche nötig sind, um sich einzuloggen und um sich beispielsweise das Passwort eines Profils anzeigen zu lassen.

Die Schritte für den Login gestalten sich ähnlich wie im Aktivitätsdiagramm "Register", nur dass man das Master-Passwort und die Master-Passphrase nicht setzt, sondern lediglich eingibt.

Sofern man korrekt eingeloggt ist, d.h. Master-Passwort und Master-Passphrase wurden als korrekt verifiziert, können die Account-Passwörter beliebig oft angezeigt werden. Es sind also keine nochmaligen Eingaben der Master-Passphrase nötig, um ein Profil-Passwort anzeigen zu lassen.

Das nachstehende Diagramm zeigt diese Aktivitäten auf.

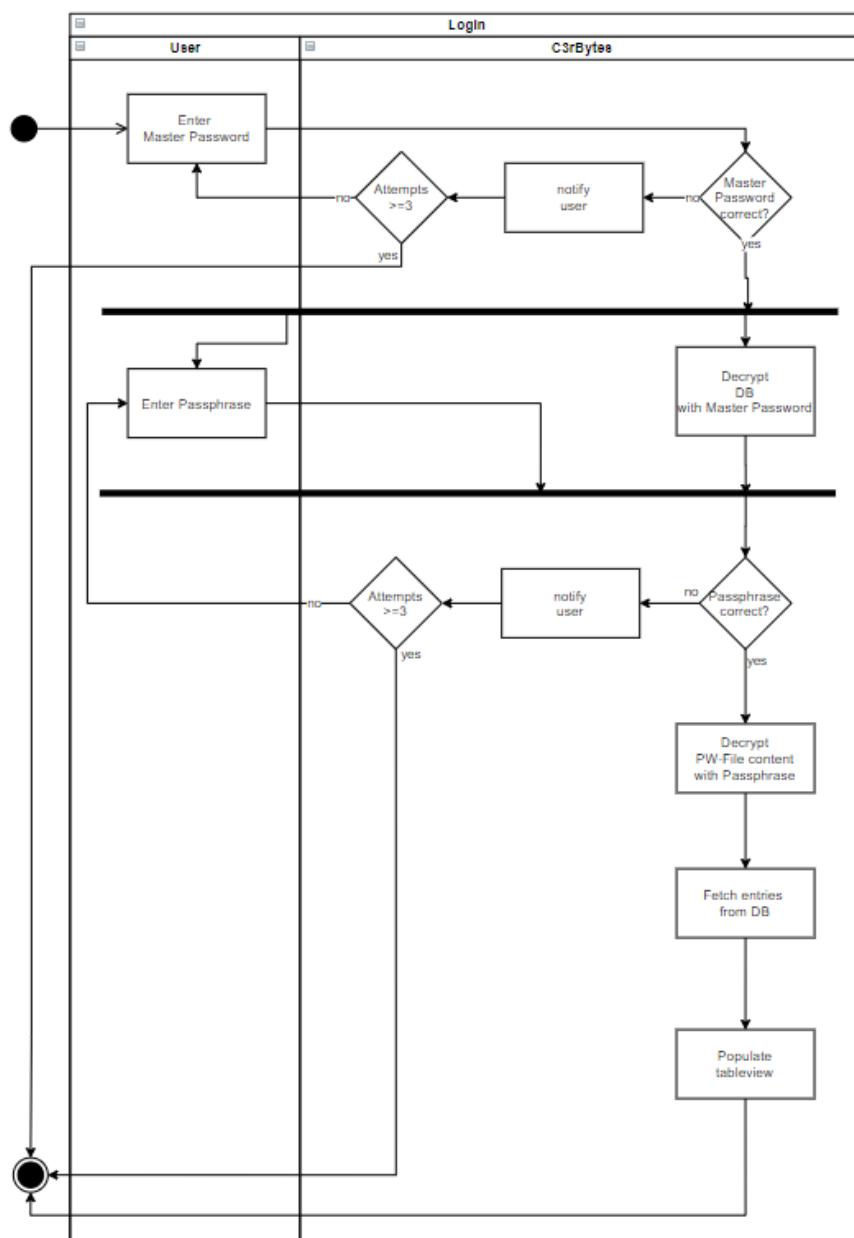


Abbildung 31: Aktivitätsdiagramm Login

2.15.4. Neues Profil anlegen

Das nachfolgende Aktivitätsdiagramm beschreibt die Erfassung eines neuen Profils.

Sobald der Nutzer in der Hauptansicht auf den Button "add profile" gedrückt hat, öffnet sich die Ansicht add_new_item. Dort muss mindestens der User name, ein Passwort und ein Typ erfasst werden, damit ein Eintrag erfolgreich gespeichert werden kann. Dabei kann der Nutzer wahlweise sein Passwort selber eintragen oder sich vom Passwortgenerator ein Passwort nach seinen Wünschen erstellen lassen. Nach dem Speichern kehrt man zurück zur Hauptansicht und die Tabelle wird aktualisiert.

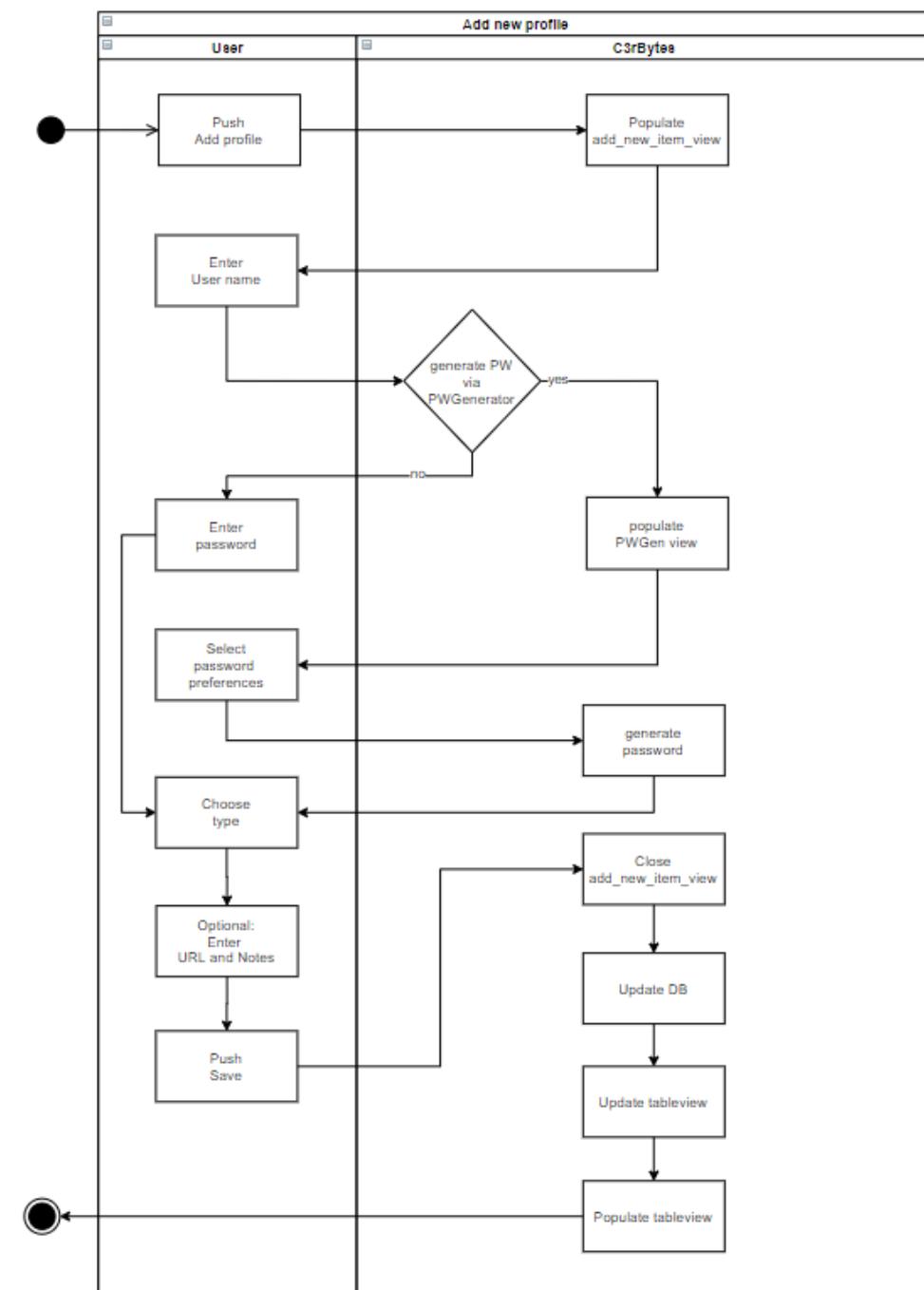


Abbildung 32: Aktivitätsdiagramm Neues Profil hinzufügen

2.15.5. Profil löschen

Will der Nutzer einen Eintrag wieder löschen, geht er gemäss nachfolgendem Diagramm vor.

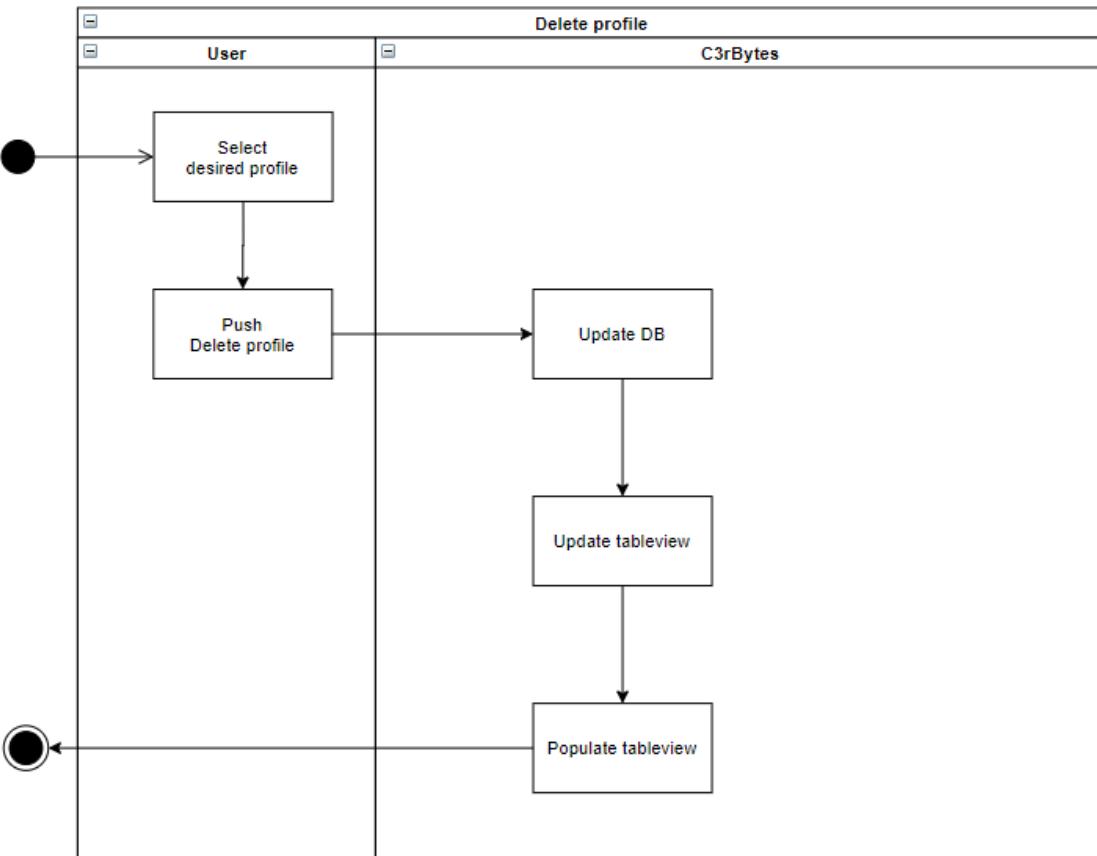


Abbildung 33: Aktivitätsdiagramm Profil löschen

Der Nutzer markiert den zu löschen Datensatz in der Tabelle der Hauptansicht und betätigt den Button “Delete Profile”. C3rBytes löscht daraufhin den betreffenden Eintrag in der Datenbank, und übergibt der View die aktualisierten Daten.

2.16. Sequenzdiagramme

Sequenzdiagramme zeigen die Modellierung von Interaktionen zwischen den beteiligten Objekten auf. Der User ist in diesem Sinne keine Klasse und kein Objekt. Beim User handelt es sich um den Benutzer, welcher mit C3rBytes interagiert.



2.16.1. Registrierung

Wie im Aktivitätsdiagramm Registrierung bereits beschrieben wurde, handelt es sich hierbei um die erstmalige Anmeldung bzw. den initialen Setup-Prozess auf Objekt-Interaktionsebene.

Der Nutzer öffnet C3rBytes. Das Programm überprüft nun, ob die initialen Settings bereits vorgenommen wurden, indem überprüft wird, ob eine gewisse Datei, auch File genannt, existiert. Falls nicht wird der Nutzer aufgefordert zuerst das initiale Setup zu durchlaufen. Dabei muss der Nutzer ein Master-Passwort setzen. Der Controller nimmt dieses entgegen und kontaktiert das DAO (DataAccessObject) damit die Datenbank mit dem hash des Master-Passwortes verschlüsselt werden kann.

Anschliessend kann der Nutzer eine Master-Passphrase setzen. Wie beim Master-Passwort nimmt der Controller die Master-Passphrase entgegen, stösst den Passwort-Generator an, welcher ein Passwort generiert und dem Controller zurückliefert. Der Controller startet nun einen FileEncrypterDecrypter und übergibt diesem den Dateinamen, die Master-Passphrase und das generierte Passwort des Generators. Der FileEncrypterDecrypter verschlüsselt das generierte Passwort mit der Master-Passphrase und speichert diese im File ab.

Wurde die Sequenz erfolgreich durchlaufen, ist das Programm eingerichtet und betriebsbereit.

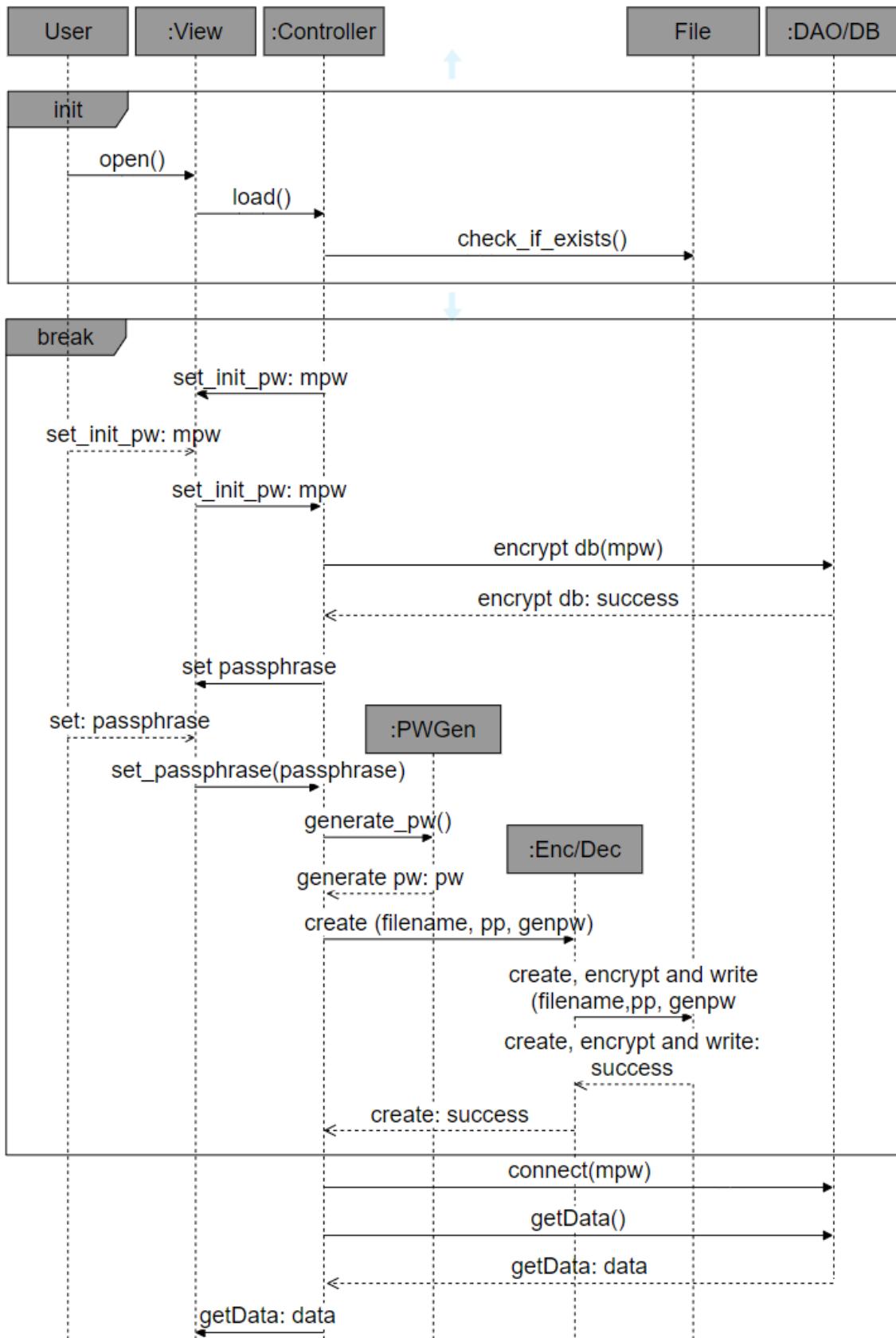


Abbildung 34: Sequenzdiagramm Registrierung



2.16.2. Login

Nach dem erfolgreichen initialen Setup kann sich der Nutzer gemäss untenstehendem Diagramm einloggen. Wie beim Aktivitätsdiagramm bereits geschildert, geschieht dies ähnlich wie beim initialen Login mit der Ausnahme, dass kein Master-Passwort und keine Master-Passphrase gesetzt, sondern lediglich eingegeben und validiert werden. Bei erfolgreicher Anmeldung entschlüsselt und öffnet sich die Datenbank mit den darin enthaltenen Profilen.

Nach jeweils drei erfolglosen Loginversuchen für das Master-Passwort-Login und das Master-Passphrase-Login wird das Programm beendet.

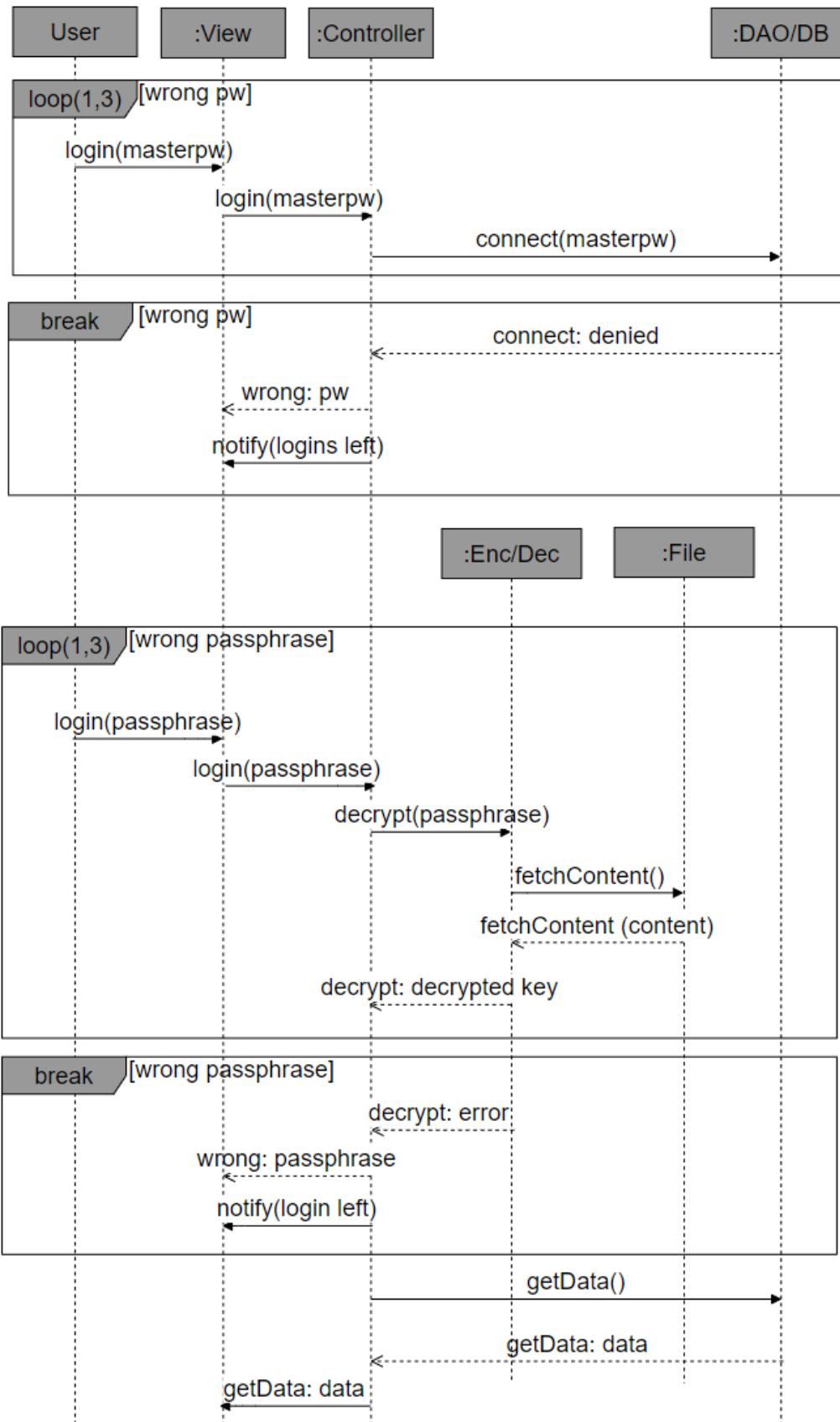


Abbildung 35: Sequenzdiagramm Login

2.16.3. Eintrag erstellen

Möchte der Nutzer einen neuen Eintrag erstellen, durchläuft das Programm folgende Sequenzen.

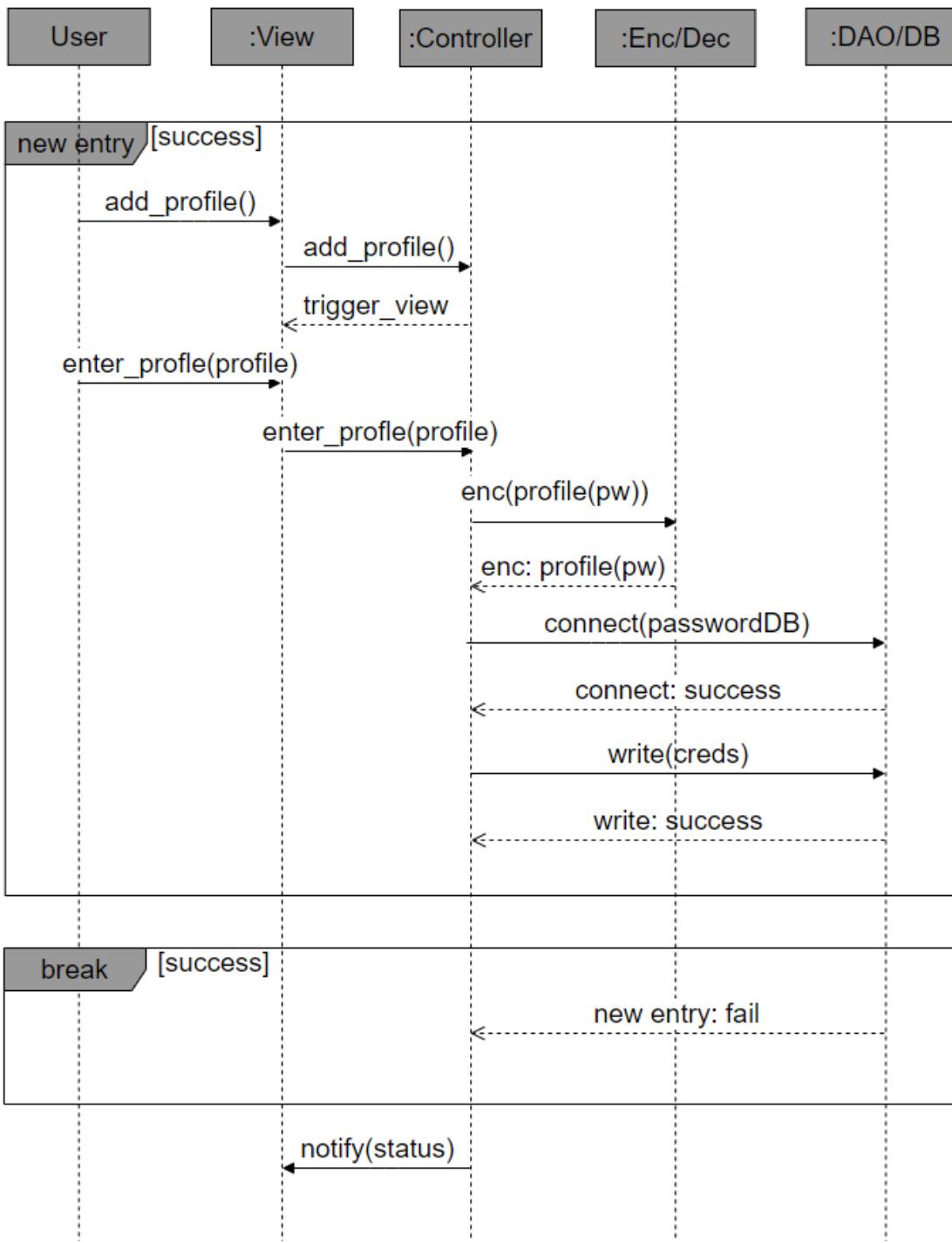


Abbildung 36: Sequenzdiagramm Eintrag erstellen

Der Nutzer kommuniziert dem Controller, dass er einen neuen Eintrag machen möchte. Dieser übernimmt die Eingaben. Das Passwort eines Eintrages wird dem Encrypter/Decrypter übergeben, damit er dieses mit der Master-Passphrase verschlüsseln kann. Der Encrypter/Decrypter übergibt das verschlüsselte Passwort an den Controller zurück. Der Controller seinerseits übergibt die Daten des Eintrags an das DAO weiter, damit dieses in der Datenbank persistiert werden kann.

Sollte die Sequenz scheitern, wird dies dem Nutzer mitgeteilt. Es findet keine Speicherung des Datensatzes statt.

2.16.4. Eintrag löschen

Will der Nutzer hingegen einen Eintrag löschen, ist dies im folgenden Diagramm ersichtlich.

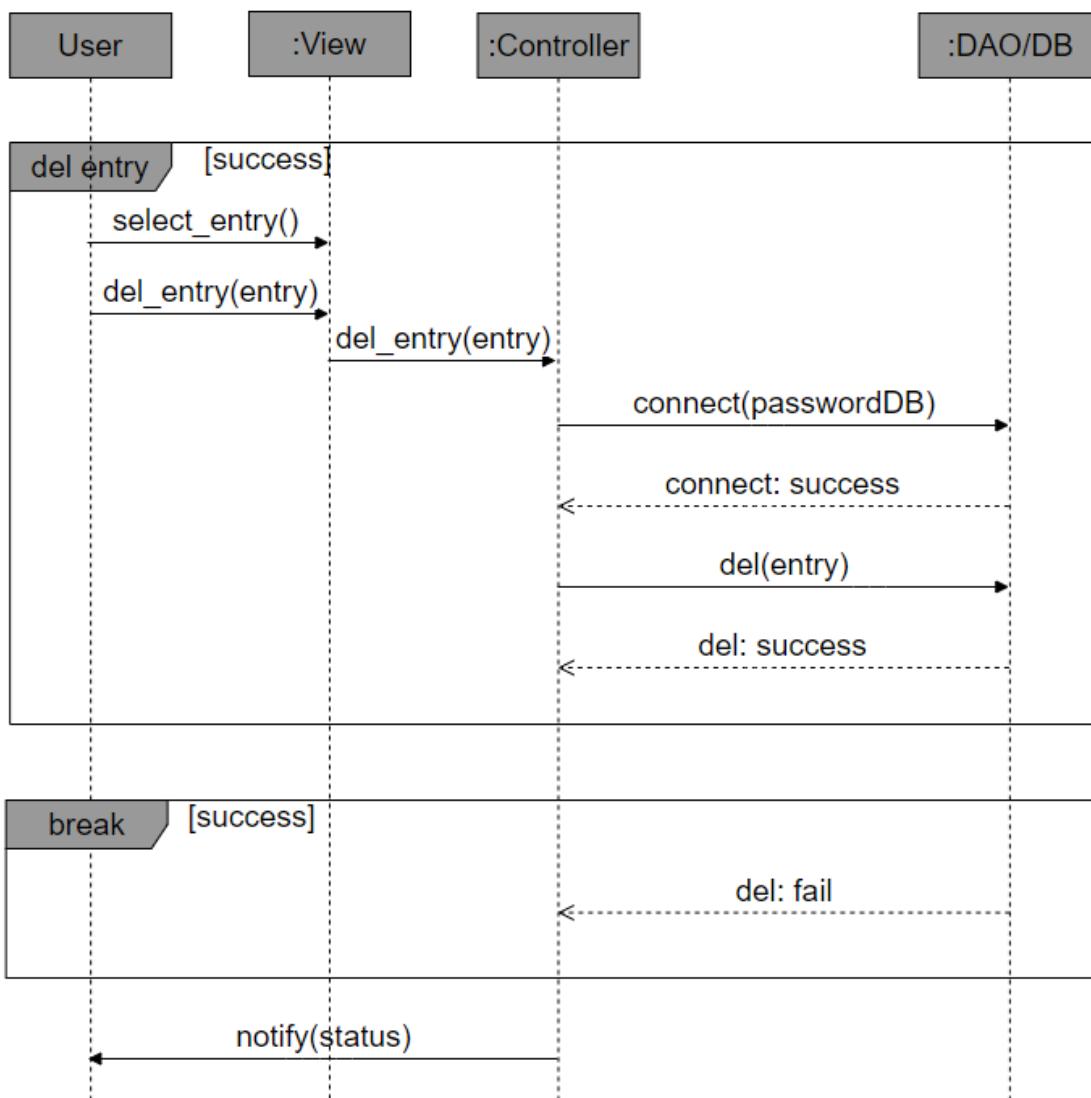


Abbildung 37: Sequenzdiagramm Eintrag löschen

Der Nutzer selektiert die zu lösrende Zeile in der Tabelle. Anschliessend betätigt er den Button zum Löschen des Eintrags. Die View teilt dem Controller den korrekten Eintrag mit. Der Controller übermittelt dem DAO den zu löschenen Eintrag, welcher seinerseits der Datenbank die Anweisung zum Löschen erteilt.

2.16.5. Eintrag ändern

Der Nutzer hat auch die Möglichkeit einen Eintrag zu ändern.

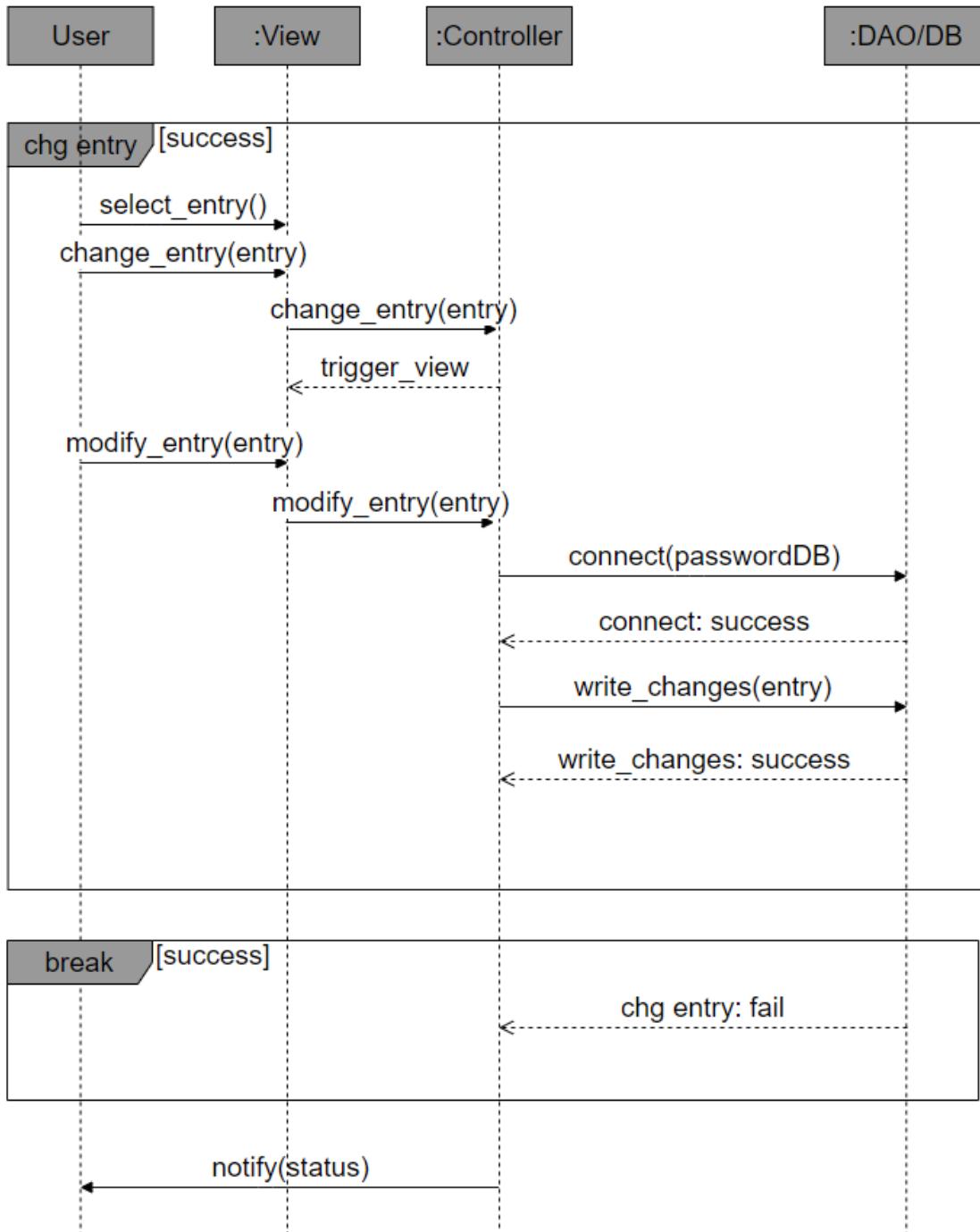


Abbildung 38: Sequenzdiagramm Eintrag ändern

Der Nutzer selektiert in der View den zu ändernden Eintrag und führt die Funktion ändern aus. Die View übergibt dem Controller den ausgewählten Eintrag. Der Controller öffnet daraufhin eine neue View, wo der Nutzer die Anpassungen vornehmen kann. Die View übergibt die Änderungen an den Controller. Dieser wiederum übergibt die Anpassungen an das DAO / DB, wo die Anpassungen persistiert werden.

2.16.6. Master-Passwort ändern

Ein heikleres Unterfangen ist es, das Master-Passwort zu ändern:

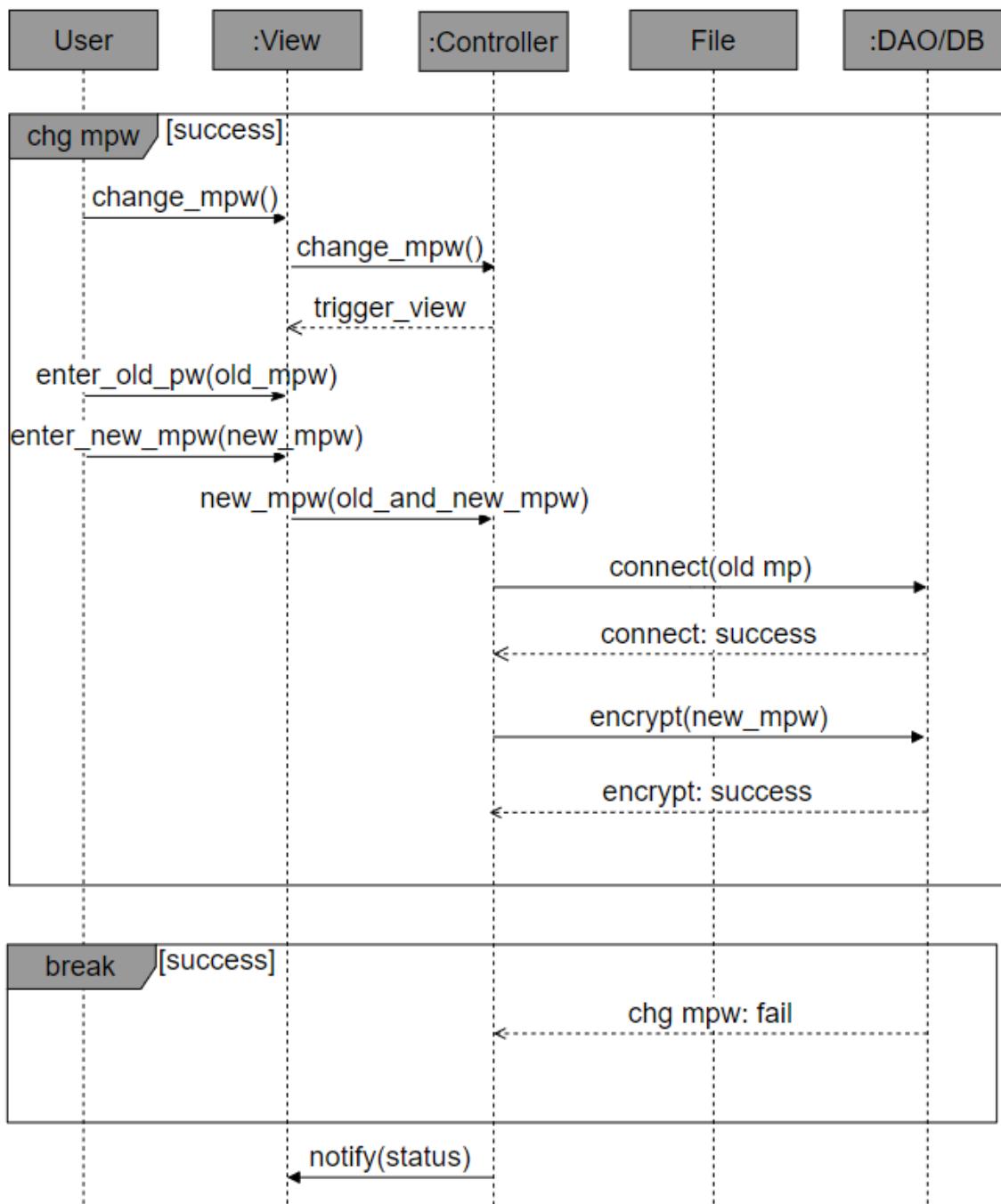


Abbildung 39: Sequenzdiagramm Master-Passwort ändern

Der Nutzer kann über die View das Ändern des Master-Passwortes initiieren. Die View teilt dies seinem Controller mit. Dieser startet eine View, wo der Nutzer sein altes und zwei Mal sein neues Master-Passwort setzen kann. Anschliessend stösst der Controller die DB an, um das Passwort zu ändern.

2.16.7. Master-Passphrase ändern

Ähnlich wie beim Ändern des Masterpassworts kann auch die Passphrase unabhängig vom Master-Passwort geändert werden:

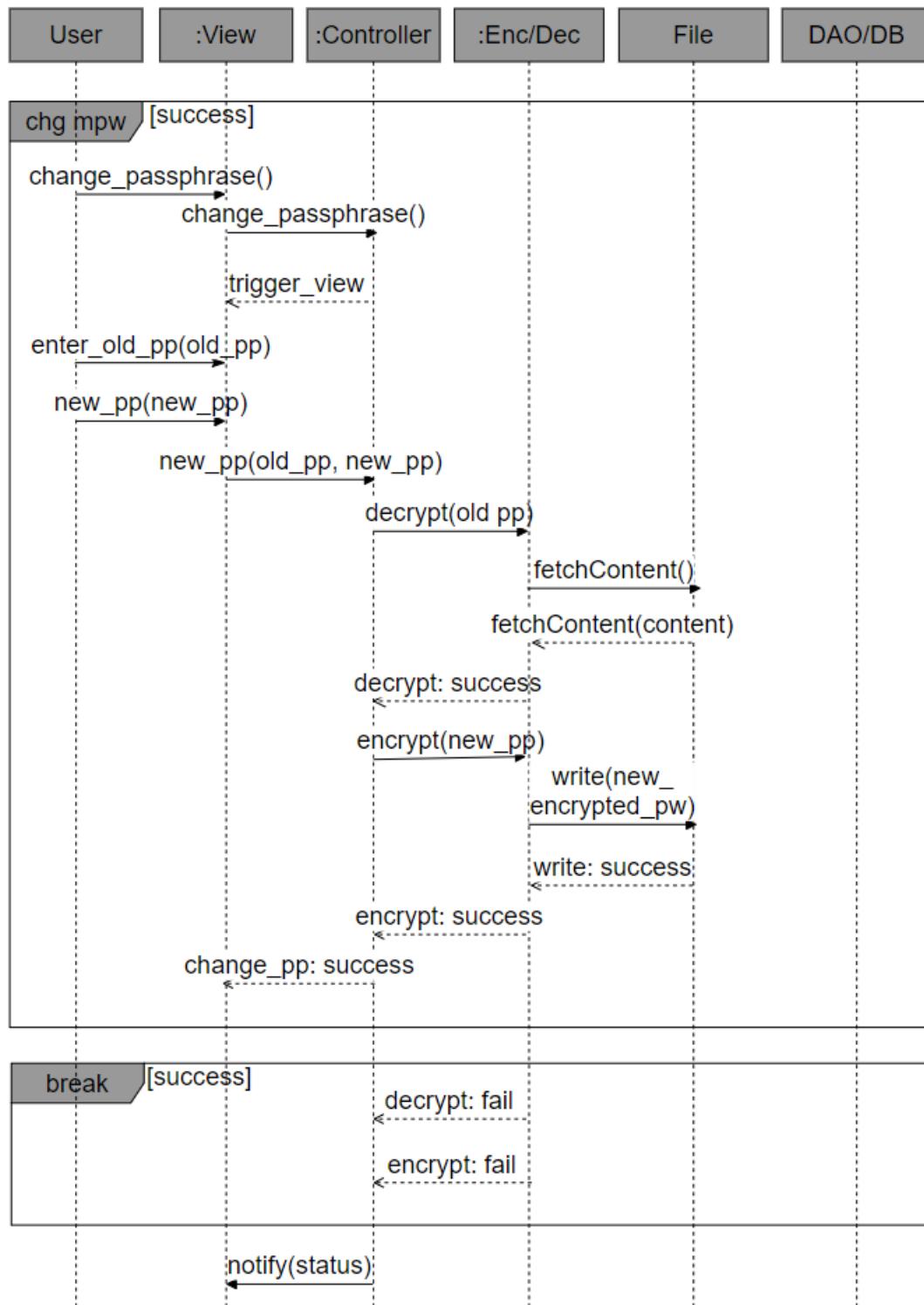


Abbildung 40: Sequenzdiagramm Master-Passphrase ändern

Der Nutzer kommuniziert der View, dass er die Master-Passphrase wechseln will. Diese teilt ihrem Controller mit, dass ein Master-Passphrase Wechsel bevorsteht. Der Controller öffnet eine neue View, wo der Nutzer die korrekte alte sowie zwei Mal die neue Master-Passphrase eingeben muss. Diese View übergibt diese Parameter an ihren Controller. Dieser übergibt die alte Master-Passphrase an den Encrypter/Decrypter, welcher die Eingabe validiert, indem er die Daten aus dem File ausliest und diese versucht zu entschlüsseln. Nach erfolgreichem Entschlüsseln übergibt der Controller dem Encrypter/Decrypter die neue Passphrase zum Verschlüsseln, die im selben File wieder abgelegt wird.

2.17. Verteilungssicht

Die Verteilung von C3rbytes ist in der nachfolgenden Abbildung ersichtlich. Im Grunde wird C3rBytes auf einem USB-Device ausgeliefert, welcher in einem Betriebssystem gemounted bzw. eingehängt wird und auf dem USB-Device ausgeführt wird.

Auf dem USB-Device befindet sich die ausführbare Datei, welche den JAVA-Code von C3rBytes sowie die zugehörige Datenbank enthält.

Das Betriebssystem sollte über einen Standartbrowser neuster Generation verfügen.

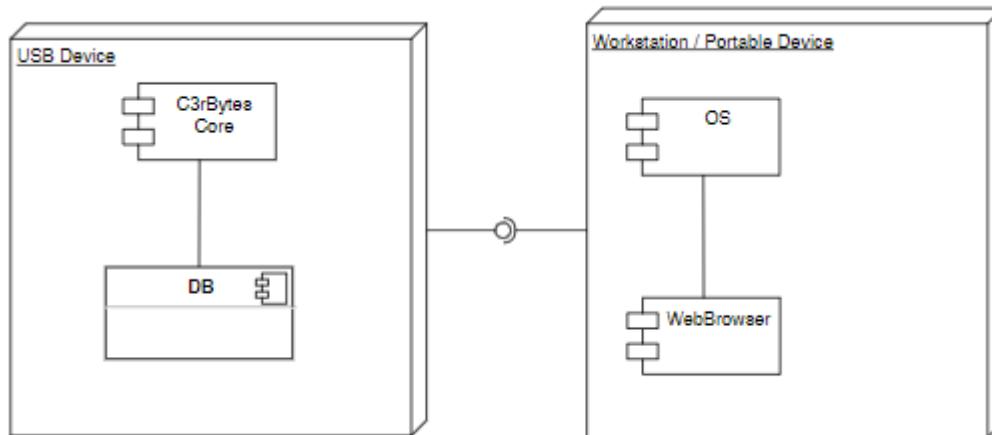


Abbildung 41: Verteilungsdiagramm

2.18. Datenbankarchitektur

2.18.1. Entity–relationship Diagramm

Unsere Datenbank enthält ein Schema (CERBYTES) sowie eine Tabelle (database_entries) mit 8 Feldern bestehend aus einer ID, einem Benutzernamen, einer Beschreibung des Kontotyps, einer URL, einem Passwort, einem Kontoerstellungsdatum, einem Kontoaktualisierungsdatum und einem Feld für Notizen.

Anfangs wollten wir unsere Datenbank so gut wie möglich normalisieren. Wir haben mit dieser Architektur begonnen:

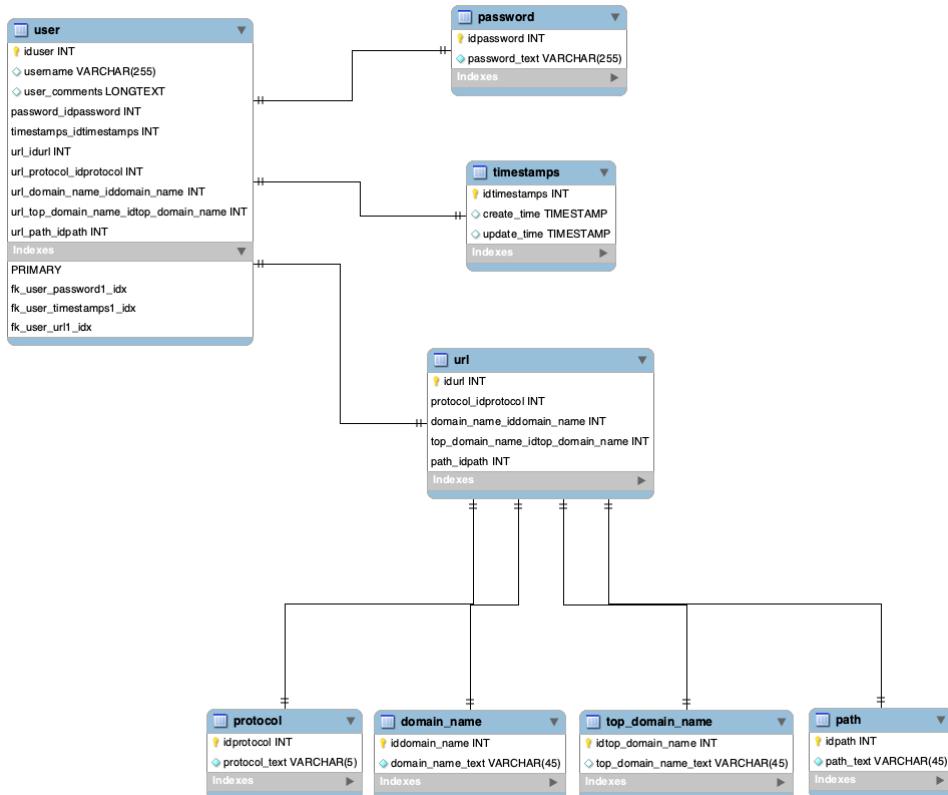


Abbildung 42: Entity-Relationship Diagramm

Daraufhin mussten wir feststellen, dass zur Speicherung eines einzigen Objekts viele Anfragen nötig gewesen wären, und damit kostbare Zeit verschwendet werden würde. Wir verwenden die eingebettete Version von Derby. Daher kann sich die Zeit der Anfragen in diesem Fall schnell erhöhen, wenn man zudem Rücksicht auf die Sicherheit und den Komfort der Benutzung nehmen will.

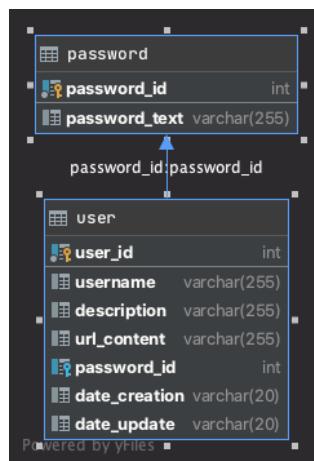


Abbildung 43: Entity-Relationship Diagramm

Zwei Anfragen zur Registrierung eines Artikels sind akzeptabel und überschaubar. Bei den ersten Registrierungstests war die Latenz jedoch noch hoch (manchmal bis zu 2,5 Sekunden), was die Benutzung des Produkts weniger reibungslos, ja sogar störend machte. Daher haben wir uns aus Gründen der Sicherheit und der Flüssigkeit unentschieden, und unsere Datenbank nicht zu normalisieren und sie in der folgenden Form aufzubauen:

database_entries	
user_id	int
username	varchar(80)
description	varchar(25)
url_content	varchar(255)
password_text	varchar(100)
date_creation	varchar(50)
date_update	varchar(50)
note	clob(2048)

Powered by yFiles

Abbildung 44: Datenstruktur database_entries

Die Query sieht wie folgt aus:

```
CREATE TABLE \"CERBYTES\".\"database_entries\" (\n" +
    "        \"user_id\" INTEGER PRIMARY KEY GENERATED ALWAYS AS IDENTITY(Start with 1, Increment
by 1),\n" +
    "        \"username\" VARCHAR(80) DEFAULT NULL,\n" +
    "        \"description\" VARCHAR(25) DEFAULT NULL,\n" +
    "        \"url_content\" VARCHAR(255) DEFAULT NULL,\n" +
    "        \"password_text\" VARCHAR(100) DEFAULT NULL,\n" +
    "        \"date_creation\" VARCHAR(50) DEFAULT NULL,\n" +
    "        \"date_update\" VARCHAR(50) DEFAULT NULL,\n" +
    "        \"note\" CLOB(2K) DEFAULT NULL);\n);
```

Wir haben die technische Entscheidung getroffen, Felder in der Datenbank bereitzustellen, die es dem Benutzer ermöglichen, eine gute Anzahl von Zeichen zu speichern (Grund: Kreativität des Benutzers).

So kann der Benutzer einen kreativen und langen Benutzernamen finden, eine ziemlich lange Url und ein Passwort verwenden. Die anderen Felder (`user_id`, `date_creation`, `date_update`) werden durch Werte bestimmt, die automatisch generiert werden (z. B. `date` dd-mm-yyy hh:mm:ss -> 14-12-2020 23:30:10).

2.18.2. Was ist Derby?

Apache Derby ist eine relationale Open-Source-Datenbank, die vollständig in Java implementiert und unter der Apache-Lizenz, Version 2.0, verfügbar ist.

- Derby hat einen geringen Platzbedarf - etwa 3,5 Megabyte für die Basis-Engine und den eingebetteten JDBC-Treiber.
- Derby basiert auf den Java-, JDBC- und SQL-Standards.
- Derby bietet einen eingebetteten JDBC-Treiber, mit dem Sie Derby in jede Java-basierte Lösung einbetten können.
- Derby unterstützt auch den bekannten Client/Server-Modus mit dem Derby Network Client JDBC-Treiber und Derby Network Server.
- Derby ist einfach zu installieren, bereitzustellen und zu warten. (The Apache DB Project, 2020)

2.18.3. Wie ist Derby in C3rBytes konfiguriert?

In unserem Projekt verwenden wir die eingebettete Version von Derby (eingebetteter Treiber). Dies bedeutet, dass die Datenbank mit der folgenden Anweisung gestartet wird:

```
DriverManager.getConnection(JDBC_URL);
```

Das Derby Url (`JDBC_URL`) besteht aus den folgenden Elementen:

- [driver]:[database product name]:[database name];[create=true/false]
- `jdbc:derby:db/cerbytes;create=true`

Dies sind die minimalen Attribute. Wenn die Datenbank nicht existiert, wird sie angelegt. Anschließend kann `[create=true]` entfernt werden.

Wenn kein Benutzer als Parameter gesetzt ist, dann gehört die Datenbank dem Standardbenutzer APP.

Daher übergeben wir in unserem Fall bei der Erstellung der Datenbank einen Benutzer `[user=cerbytes]` als Parameter. Dies erlaubt uns, `cerbytes` als Datenbankbesitzer zu konfigurieren und ihm wertvolle Rechte zu geben, die für die Kommunikation mit der Datenbank, aber auch für die Änderung des Verschlüsselungsschlüssels oder das Anhalten der verschlüsselten Datenbank notwendig sind.

Im C3rBytes-Projekt ist die Authentifizierung **nativ**. Benutzernamen und verschlüsselte Passwörter werden in einer Datenbank gespeichert. Die Credentials werden in der Systemtabelle `SYSUSERS` der Datenbank gespeichert.

Die gesamte Kommunikation findet mit dem `cerbytes`-Benutzer statt, d.h. für jede Anfrage an die Datenbank wird ein **Benutzername** und ein **Passwort** als Attribut gesetzt. Wenn eine der beiden nicht korrekt ist, wird eine Sql-Ausnahme geworfen.

```
jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=2b457e4d2b862bdeb05b3de0aed167c5
```

Leider werden Passwörter nicht verschlüsselt, wenn sie als Attribut der URL übergeben werden. Für eingebettete und Einzelbenutzer ist dies» kein Problem». Für eine Netzwerkversion wird empfohlen, SSL/TLS zu konfigurieren.

Derby erlaubt es, die Datenbankdaten auf der Festplatte zu verschlüsseln. Dies erhöht die Sicherheit der Daten, da sie gesichert (verschlüsselt) sind, wenn die Datenbank ausgeschaltet ist. Die einzige Möglichkeit, sie zu lesen, ist, sie mit dem **Bootpasswort** zu starten.

Für die Datenbankverschlüsselung muss

1. ein **Verschlüsselungspasswort** als Attribut gesetzt werden. Dies ist das Kennwort, das die verschlüsselten Daten freischaltet, wenn der Benutzer sie verwenden möchte.
2. Ein **Verschlüsselungsalgorismus**: Dies ist ein Transformationsname, wie er in der API-Dokumentation für die Klasse `javax.crypto.Cipher` beschrieben ist. Die Derby-Verschlüsselung stützt sich auf die JCE-Bibliotheken (JCE Libraries), die mit der virtuellen Maschine geliefert werden.

Denken Sie daran, dass die Datenbank mit dem Bootpasswort entschlüsselt wird, und zwar für die gesamte Lebensdauer der JVM oder bis der Datenbankbesitzer sie ausschaltet.

Um unsere Datenbank zu verschlüsseln, übergeben wir die folgenden Attribute in der URL

1. `dataEncryption= true` : ermöglicht Verschlüsselung
2. `bootPassword=myPassword123` : der Verschlüsselungsschlüssel

3. `encryptionAlgorithm=AES/CBC/NoPadding`: der Verschlüsselungsalgorithmus. Wie bei unserer Analyse entschieden, haben wir den AES-Algorithmus gewählt.
4. `encryptionKeyLength=256` (bits); das Bootpasswort muss mindestens 8 Zeichen haben. Wir haschen das Passwort, bevor wir es als Attribut an die Datenbank übergeben. Außerdem erzeugen wir mit dem SHA3-Algorithmus immer eine Zeichenfolge von 64Bytes, unabhängig von der Länge des Passwortes (die Länge des Passwortes vor dem Hash spielt jedoch eine Rolle für die Sicherheit des Passwortes).
5. `user=username;password=password`: wenn die Authentisierung aktiv ist, dann müssen Benutzername und Passwort des Datenbankbesitzers ebenfalls als Parameter übergeben werden (Verschlüsselung ist eine restriktive Operation).
6. Und schließlich muss, um die Datenbank mit einem neuen Schlüssel neu zu verschlüsseln, beim Neustart der Datenbank der Wechsel des Attributes erfolgen
`newBootPassword=newBootpassword`

Unsere URL lautet also wie folgt:

1. **Verschlüsselung:**
`jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=2b457e4d2b862bdeb05b3de0aed167c5;dataEncryption=true;encryptionKeyLength=256;encryptionAlgorithm=AES/CBC/NoPadding;bootPassword=48c8947f69c054a5caa934674ce8881d02bb18fb59d5a63eeaddff735b0e9801e87294783281ae49fc8287a0fd86779b27d7972d3e84f0fa0d826d7cb67dfefc`
2. **Kommunikation mit der Datenbank**
`jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=2b457e4d2b862bdeb05b3de0aed167c5`

Schließlich wollen wir, dass die Groß- und Kleinschreibung in unserer Datenbank nicht beachtet wird (für Suchfunktionen). Außerdem erfassen wir bei der Erstellung die Spracheinstellungen des Systems des Benutzers und übergeben sie als Attribut in der URL. Um eine Datenbank mit ortsbezogener Sortierung zu erstellen, geben Sie die Sprach- und Ländercodes für das Attribut `territory=fr_CH` und den Wert `TERRITORY_BASED` für das Attribut `collation=collation` an:

- `collation=TERRITORY_BASED:PRIMARY`
- `territory=fr_CH`

Bei Stärke `PRIMARY` werden die Zeichen 'A' und 'a' als gleichwertig betrachtet, ebenso wie 'a' ('a' mit schwerem Akzent). (Dieses Verhalten ist bei vielen anderen Datenbanken die übliche Voreinstellung). (The Apache DB Project, o.D.)

So sieht unsere URL aus:

`jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=46bc500c10a84b8d1dae26750de31cbb;territory=fr_CH;collation=TERRITORY_BASED:PRIMARY; [...]`

2.18.4. Ablauf beim Erstellen eines neuen Kontos in C3rBytes :

1. Erstellung einer verschlüsselten Datenbank mit Benutzer (`cerbytes`). Der Verschlüsselungsschlüssel ist das Master-Passwort (64 Bytes). Das Datenbankbenutzer-Passwort wird aus dem Master-Passwort abgeleitet. Das Master-Passwort wird gehasht, dann wird der Wert erneut gehasht und wird zum Passwort des `cerbytes`-Benutzers (32 Bytes). Die Sicherheit ist daher benutzerbasiert. Wenn sein Master-Passwort sicher ist, dann ist auch die Sicherheit von C3rBytes sicher.
2. Anschliessend speichern wir ein Passwort für den Benutzer `cerbytes` :
`"CALL SYSCS_UTIL.SYSCS_CREATE_USER(?,?)";`
 - a. Der Benutzer wurde beim Anlegen der Datenbank angelegt. Wir können das Passwort des Benutzers nicht zurücksetzen, bevor der Benutzer auf diese Weise erstellt wurde. Dieser Benutzer ist der Besitzer der Datenbank.
 - b. Der aufmerksame Leser wird bemerkt haben, dass wir das Passwort als Attribut bereits auf (1) übergegeben haben. Das Passwort ist jedoch noch nicht vom Derby-System registriert und dem Benutzer `cerbytes` zugeordnet. Dies liegt an

der Art und Weise, wie wir die Url generieren. Wir haben zwei Methoden: `createUrl()`, eine sehr vollständige Url, die wir für die Verschlüsselung und den Start der Datenbank verwenden (mit Passwort, sonst startet die Datenbank nicht) und `createUrlSimple()`, eine Url ohne die Verschlüsselungsoptionen. Beide Methoden übergeben das Passwort des Benutzers als Attribut. Eine Alternative wäre eine Dritte Methode zu erstellen aber würde auch zur Codeduplikation führen. (The Apache DB Project, 2020).

3. Dann erstellen wir das Shema CERBYTES und die Tabelle database_entries

```
String sqlCreateSchema = "CREATE SCHEMA \"CERBYTES\"";  
String sqlCreateTable = CREATE TABLE \"CERBYTES\".\"database_entries\" (\n" +  
    "        \"user_id\" INTEGER PRIMARY KEY GENERATED ALWAYS AS IDENTITY(Start with 1, Increment  
by 1),\n" +  
    "        \"username\" VARCHAR(80) DEFAULT NULL,\n" +  
    "        \"description\" VARCHAR(25) DEFAULT NULL,\n" +  
    "        \"url_content\" VARCHAR(255) DEFAULT NULL,\n" +  
    "        \"password_text\" VARCHAR(100) DEFAULT NULL,\n" +  
    "        \"date_creation\" VARCHAR(50) DEFAULT NULL,\n" +  
    "        \"date_update\" VARCHAR(50) DEFAULT NULL,\n" +  
    "        \"note\" CLOB(2K) DEFAULT NULL);
```

Schließlich können wir das Master-Passwort ändern, um die Datenbank mit einem neuen Boot-Passwort erneut zu verschlüsseln. Deshalb müssen wir auch das Datenbank-Passwort (`passwordDB`) ändern, weil beide miteinander verknüpft sind und wir beim nächsten Login Probleme bekommen würden (das Datenbank-Passwort würde nicht mehr übereinstimmen).

Neues BootPassword

```
jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=33c592dc463e20226dbebc7e7c267463;dataEncryption=  
true;encryptionKeyLength=256;encryptionAlgorithm=AES/CBC/NoPadding;bootPassword=4ad2c01fc6007f58720b00fc  
99b978c2a17c577859d31fdbba4b3a749de9383ac4b0738aeaf0b13337db8bfeaf9d8f87faa236fc3c8a68fbf23eb6862fa  
db86e ;newBootPassword=e1e44d20556e97a180b6dd3ed7ae5c465cafd553fa8747dca038fb95635b77a37318f7ddf7a  
ec1f6c3c14bb160ba2497007decf38dd361cab199e3b8c8fe1f5c
```

Neues Passwort der Datenbank.

```
jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=33c592dc463e20226dbebc7e7c267463  
"CALL SYSCS_UTIL.SYSCS_RESET_PASSWORD(?,?)"  
jdbc:derby:db/cerbytes;create=true;user=cerbytes;password=46bc500c10a84b8d1dae26750de31cbb
```

2.19. Benutzeroberfläche

2.19.1. Login View

Die Login View besteht aus zwei Gruppen von zwei aufeinander folgenden, und nahezu identisch aussehenden Fenstern. Die erste Gruppe erscheint nur beim ersten Einloggen, wenn keine Datenbank existiert und fordert den Nutzer auf ein Master-Passwort und eine Master-Passphrase zu setzen. Die zweite Gruppe erscheint nur, wenn bereits eine Datenbank besteht.

Alle Fenster führen drei Buttons auf, eines zum Einloggen, worauf die Datenbank entschlüsselt wird und eines zum Ausloggen, was das jeweilige Fenster schliesst. Ein dritter Knopf existiert, um das versteckte Passwort anzuzeigen.

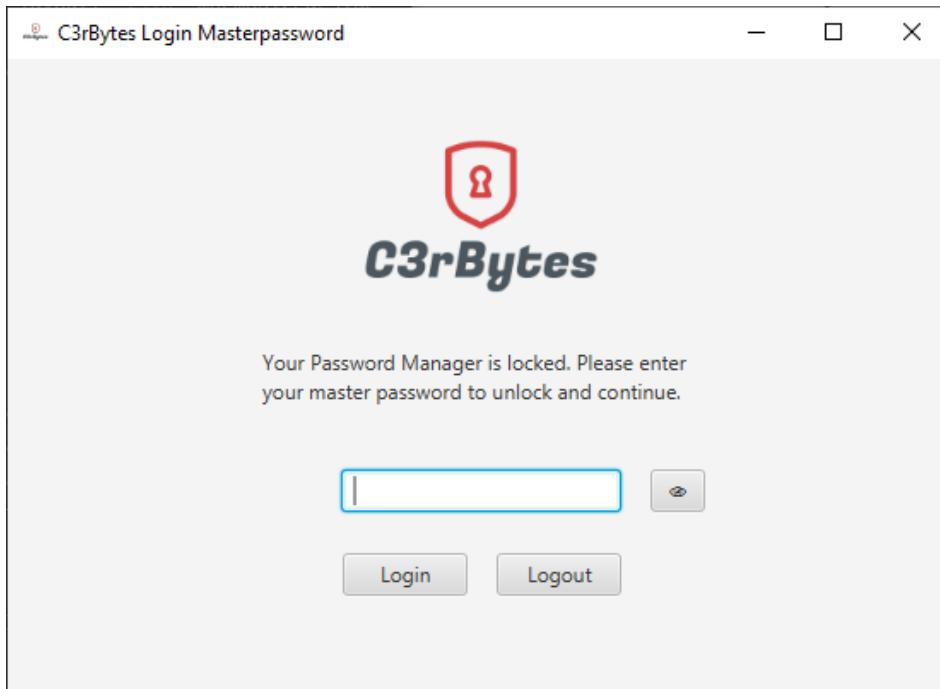


Abbildung 45: Login C3rBytes Master Password

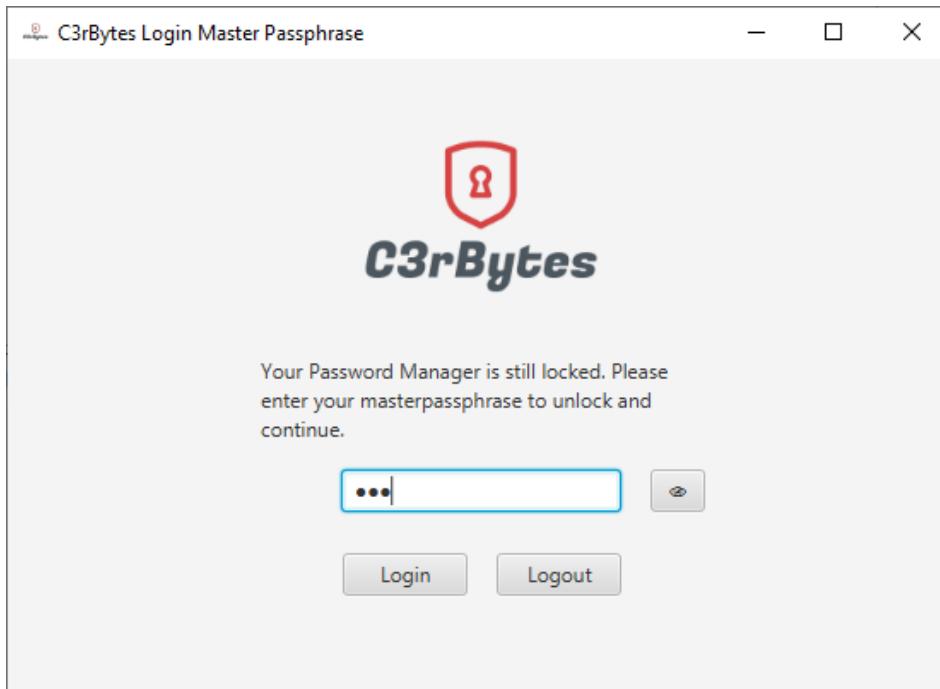


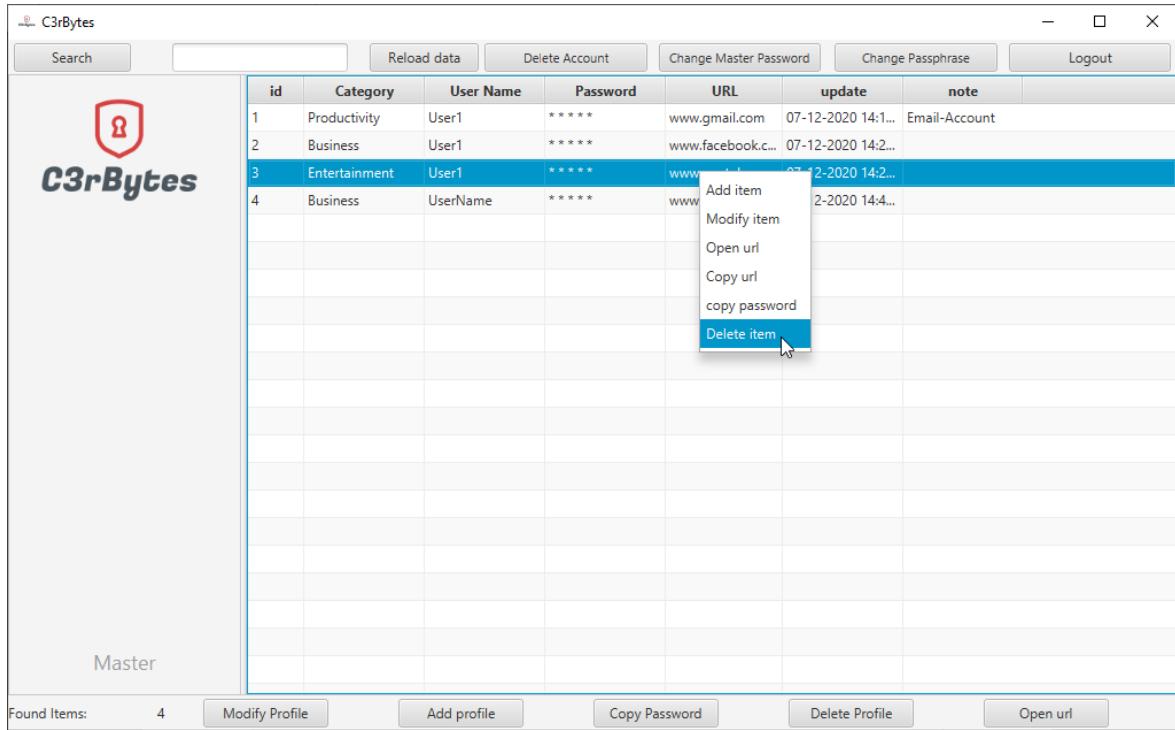
Abbildung 46: Login C3rBytes Master-Passphrase

2.19.2. Main View

Die Main View besteht aus vier Zonen, von denen die oberen und unteren Zonen für Buttons vorgesehen sind, und die zentralen Zonen vertikal unterteilt sind. Die linke Zone wird für die Suche und Informationen verwendet, während die rechte aus einer Tabelle besteht und die erstellten Profile enthält. Diese sind auswählbar und öffnen ein neues Fenster (Siehe 16.4).

Die Buttons der oberen Zone ermöglichen eine Suche, das neu Laden der Tabelle, das Löschen des Accounts, das Ändern des Master-Passworts und -Passphrase und das Ausloggen vom Programm.

Die Buttons unten erlauben ein Profil zu ändern und erstellen, das Passwort des ausgewählten Profils kopieren oder das ausgewählte Profil löschen. Indem man auf einen Eintrag doppelklickt, kann man ein Profil ändern, worauf auch das Update-Feld aktualisiert wird. Wenn man auf ein Profil rechtsklickt kann man ein neues Profil erstellen, ein Profil verändern, eine URL öffnen oder kopieren, das Passwort kopieren oder das Profil löschen.



The screenshot shows the C3rBytes application interface. On the left, there's a sidebar with the C3rBytes logo and a 'Master' section. The main area is a table with columns: id, Category, User Name, Password, URL, update, and note. There are four items listed:

id	Category	User Name	Password	URL	update	note
1	Productivity	User1	*****	www.gmail.com	07-12-2020 14:1...	Email-Account
2	Business	User1	*****	www.facebook.c...	07-12-2020 14:2...	
3	Entertainment	User1	*****	www.netflix.com	07-12-2020 14:2...	
4	Business	UserName	*****	www.linkedin.com	07-12-2020 14:4...	

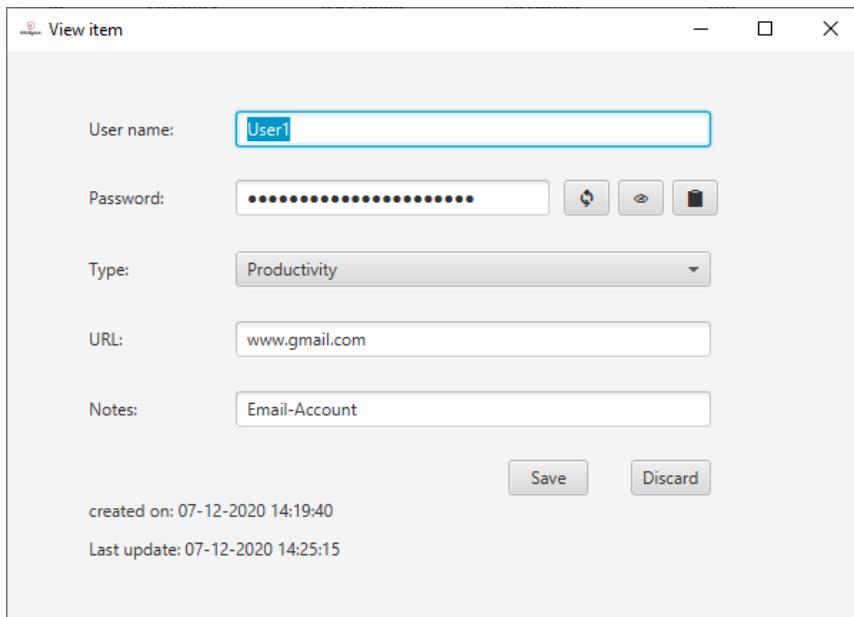
A context menu is open over the third item (ID 3), listing options: Add item, Modify item, Open url, Copy url, copy password, and Delete item. The 'Delete item' option is highlighted with a blue background.

At the bottom of the window, there are buttons for Modify Profile, Add profile, Copy Password, Delete Profile, and Open url. The status bar shows 'Found Items: 4'.

Abbildung 47: C3rBytes Main View

2.19.3. Add New Item View

Wie in [Schnittstellenbeschreibung] beschrieben, besteht ein Profil aus sieben Werten, von denen fünf vom Nutzer ausgefüllt werden können. Diese Ansicht verfügt über vier Textfelder, User Name, Password, URL und Notes, einem Drop-Down Menü für den Profiltyp und fünf Buttons. Zwei Buttons befinden sich auf dem unteren Ende des Fensters und erlauben das Speichern und Verwerfen des Profils, drei Buttons sind neben dem Password-Textfeld. Diese sind "Passwort generieren", "Passwort anzeigen" und "Passwort Kopieren".



The screenshot shows the 'View item' dialog for adding a new item. It contains fields for User name (User1), Password (redacted), Type (Productivity), URL (www.gmail.com), and Notes (Email-Account). Below the form, it shows the creation and last update times: created on: 07-12-2020 14:19:40 and Last update: 07-12-2020 14:25:15. At the bottom are Save and Discard buttons.

Abbildung 48: C3rBytes Add New Item View

2.19.4. Generate Password

Die Generate Password View bietet verschiedene Werte und Checkboxes, die für das Erstellen eines Passwortes verwendet werden. Der Nutzer kann einstellen, welche Elemente das Passwort enthält, wie z.B. Klein- und Grossbuchstaben, Zahlen oder die Länge des Passwortes.

Je ein Button zum Generieren und Kopieren des Passwortes sind neben dem Passwort-Textfeld vorhanden und das Passwort kann mit den Buttons am unteren Rand entweder gespeichert oder verworfen werden.

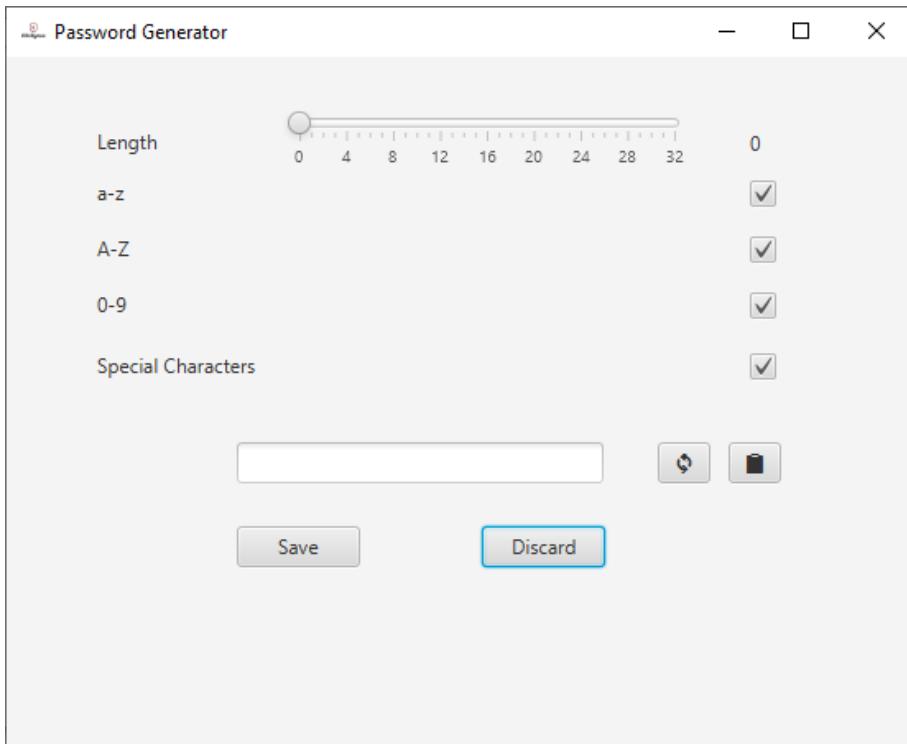


Abbildung 49: C3rBytes Password Generator

2.20. Qualitätssicherung / Tests

2.20.1. Manuelle Testszenarien

Nr.	Testszenario	Beschreibung
T1000	Registration	Test einer vollständigen Registration des Master-Accounts.
T1010	Login	Nach erfolgreicher Registration soll ein Login möglich sein und das Hauptfenster öffnet sich.
T1020	Masteraccount-Daten mutieren	In der App sind das Master-Passwort und die Master-Passphrase zu mutieren und zu prüfen, ob diese Änderungen korrekt übernommen wurden.
T1030	Account anlegen	In der Tabelle soll ein neuer Account angelegt werden und es ist zu prüfen, ob dieser persistiert wird.
T1040	Account mutieren	An den Daten des Accounts sollen Änderungen vorgenommen werden. Dabei ist zu prüfen, ob die Änderungen korrekt und dauerhaft gespeichert werden.
T1050	Accountpassword kopieren	Nachdem die Accountdaten kopiert wurden, soll überprüft werden, ob sich die korrekten Daten im Zwischenspeicher befinden.
T1060	Account löschen	Nach dem Löschen eines Accounts soll dieser auch in der Datenbank unwiderruflich gelöscht sein.
T1070	Master-Account löschen	Wird der Master-Account gelöscht, soll sichergestellt werden, dass der Account komplett samt der zugehörigen DB gelöscht wurde.
T1080	Pen-Testing des Master-Accounts	Es soll versucht werden den Master-Account mit gängigen Ethical Hacking Methoden zu hacken.
T1090	Link mit Default-Browser öffnen	Beim Klick auf den Link der Ressource soll bei einem Weblink der Default-Browser geöffnet und die Ressource aufgerufen werden.
T1100	Passwort generieren	Das Passwort soll nach den Vorgaben des Nutzers erstellt werden.
T1110	Logout	Der Nutzer soll sich korrekt ausloggen können.
T1120	Ungültiger Login	Sofern die Zutrittsdaten nicht korrekt sind, soll kein Login möglich sein.

Tabelle 29: Testszenarien

2.20.2. Testfälle / Testergebnisse

Anbei folgen die Testfälle anhand der Testszenarien, welche zur Überprüfung der Anforderungen und der Stabilität des Programms dienen sowie für die finale Qualitätssicherung ausschlaggebend ist.

Nr. Testszenario	T1000-a
Testszenario	Registration
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Erster Start der Applikation
Ablauf	Programm starten Master-Passwort setzen Master-Passphrase setzen
Erwartetes Ergebnis	Dialog zum Setzen des Master-Passwortes erscheint Passwortfile wird generiert Passwort wird vom Passwortgenerator erstellt und im Passwortfile abgespeichert Passwortfile wird mit der Passphrase verschlüsselt Datenbank wird aufgesetzt und ist leer Hauptansicht wird geöffnet
Ist-Ergebnis	Dialog zum Setzen des Master-Passwortes erfolgreich erschienen Passwortfile wurde generiert Passwort wurde vom Passwortgenerator erstellt und im Passwortfile abgespeichert Inhalt des Passwortfiles wurde mit der Passphrase verschlüsselt Datenbank wurde aufgesetzt und ist leer Hauptansicht wurde geöffnet
Anmerkungen	JUnitTest: <ul style="list-style-type: none"> - setEncryptionTest() wurde bestanden; - setUserDBWithPasswordConnectionTest() wurde bestanden; - dbExecuteUpdateTest() wurde bestanden

Tabelle 30: Testszenario 1000-a Registration

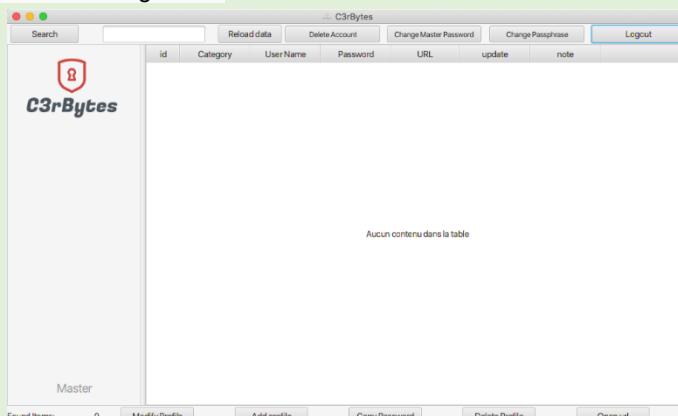


Abbildung 50: Hauptansicht

Nr. Testszenario	T1000-b
Testszenario	Registration
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm geöffnet und erfolgreich eingeloggt
Ablauf	Masteraccount löschen (delete account) Programm neu starten Ablauf gemäss T1000
Erwartetes Ergebnis	Ergebnis gemäss T1000
Ist-Ergebnis	Dialog zum Setzen des Master-Passwortes erfolgreich erschienen Passwortfile wurde generiert Passwort wurde vom Passwortgenerator erstellt und im Passwortfile abgespeichert Inhalt des Passwortfiles wurde mit der Passphrase verschlüsselt Datenbank wurde aufgesetzt und ist leer Hauptansicht wurde geöffnet
Anmerkungen	

Tabelle 31: Testszenario 1000-b Registration

Nr. Testszenario	T1010
Testszenario	Login
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt
Ablauf	Programm starten Einloggen mit Master-Passwort und Master-Passphrase
Erwartetes Ergebnis	Hauptansicht wird geöffnet Datenbankinhalt wird korrekt dargestellt.
Ist-Ergebnis	Hauptansicht wurde erfolgreich geöffnet. Datenbankinhalt wird korrekt dargestellt.
Anmerkungen	

Tabelle 32: Testszenario 1010 Login

Nr. Testszenario	T1020
Testszenario	Masteraccount-Daten mutieren
Tester	Jérémie
Testdatum	01.12.2020
Voraussetzung	Masteraccount gesetzt, Datenbank ist verschlüsselt. Programm gestartet und erfolgreich eingeloggt
Ablauf	<p>1 «Change Master Password» drücken 2 Altes Master-Passwort eingeben 3 Neues Master-Passwort setzen</p> 
	<i>Abbildung 51: Masterpassword ändern</i>
Erwartetes Ergebnis	Datenbank mit neuem Master-Passwort verschlüsselt (die 2 BootPasswords sind unterschiedlich und die Datenbank wurde gestartet.)
Ist-Ergebnis	<p>Das neue Master-Passwort wurde erfolgreich registriert. Die Datenbank wurde mit dem neuen Passwort erfolgreich verschlüsselt.</p>  <p><i>Abbildung 52: Masterpassword wurde erfolgreich geändert (unterschiedliche Werte)</i></p> <p>Die DB startet mit dem alten Master-Passwort nicht mehr.</p>
Anmerkungen	<p>Die neue Verschlüsselung ist eine rechnungsintensive Aufgabe. Mit grossen Datenbanken kann es ein Nachteil sein, aber die Sicherheit ist hier zu bevorzugen.</p> <p>JUnit Test:</p> <ul style="list-style-type: none"> - changeBootPassortTest() wurde bestanden.

Tabelle 33: Testszenario 1020 Masteraccount Daten mutieren

Nr. Testszenario	T1030
Testszenario	Account anlegen (add profile)
Tester	Jérémie Equey
Testdatum	02.12.2020
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt
Ablauf	<p>1. «Add profile» betätigen 2. Accountdaten erfassen (*Username, *Password, *Type, URL, Note) * obligatorisch 3. «Save» drücken</p>
Erwartetes Ergebnis	Add profile Ansicht schliesst sich Der neue Account erscheint korrekt in der Hauptansicht (reload Funktion)
Ist-Ergebnis	Das Fenster zum Speichern eines neuen Eintrags öffnet sich erfolgreich. Pflichtfelder können nicht leer gelassen werden (Klickt auf « save » markiert die fehlenden Pflichtfelder). Wenn alle Pflichtfelder ausgefüllt sind, wird der neue Eintrag fehlerfrei in

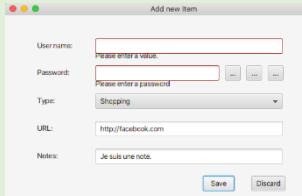
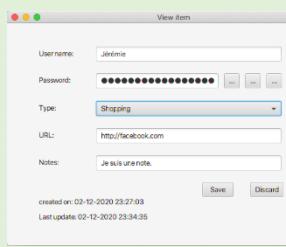
	<p>der Datenbank gespeichert und der Eintrag erscheint in der Hauptansicht. Die "Add profile"-Ansicht hat sich geschlossen.</p>
	 <p>Abbildung 53: "Add profile" Ansicht</p>
	<pre>query INSERT INTO TCEMBYTEST."database_entries" ("username", "description", "url_content", "password_text", "date_update", "note") VALUES('Jérémie', 'Shopping', 'http://facebook.com', 'DWKEIzarnhnpxv0519L19ktrc-iTwQ26P4rqJHr-fncS+HFv299y8S10QLYXyRFeq4Q7tt3+68CigCnN6Gavukpiajs8/tbve5XmLx==', '82-12-2828 23:27:83', '82-12-2828 23:27:83' Connecting to db ... connection successful</pre> <p>Abbildung 54: SQL-Query um die Felder in der Datenbank zu speichern.</p>

Tabelle 34: Testszenario 1030 Add Profile

Nr. Testszenario		T1040
Testszenario		Account mutieren
Tester		Jérémie
Testdatum		02.12.2020
Voraussetzung		Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Account vorhanden
Ablauf		Account selektieren «Modify profile» betätigen Im Fenster Details Accountdaten anpassen (User name, Password, Type, URL, Nodes) «Save» drücken
Erwartetes Ergebnis		Detail Ansicht schliesst sich. Die Änderungen im Account erscheinen korrekt in der Hauptansicht
Ist-Ergebnis		Die Detailansicht wurde beim Speichern geschlossen (save Button). Die Kontodaten wurden erfolgreich geändert und erschienen geändert in der Hauptansicht.
		 <p>Abbildung 56: View Item bevor der Änderung</p>

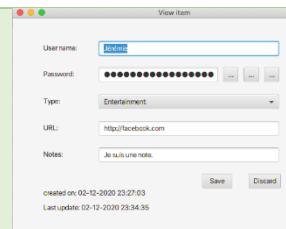


Abbildung 57: View Item nach der Änderung.

id	Category	User Name
1	Shopping	Jérémie

Abbildung 58: MainView mit der Änderung

Anmerkungen	Zwischen dem Zeitpunkt der Änderung des Kontos in der Datenbank und den neu geladenen Dateien gibt es eine kleine Wartezeit.
--------------------	--

Tabelle 35: Testszenario 1040 Account mutieren

Nr.	Testszenario	T1050
Testszenario	Accountpassword kopieren	
Tester	Jérémie Equey	
Testdatum	04.12.2020	
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Account vorhanden	
Ablauf	Account selektieren «Copy Password» betätigen In einen Textbereich wie z.B. Texteditor einfügen.	
Erwartetes Ergebnis	Beim Pasten in einen Texteditor muss das korrekte Passwort im Klartext vorliegen und nach 10 Sekunden aus der Zwischenablage entfernt werden.	
Ist-Ergebnis	Das korrekte Passwort wurde im Klartext vorgelegt.	
Anmerkungen	Nach zehn Sekunden wird das in den Speicher (Clipboard) kopierte Passwort aus Sicherheitsgründen aus dem Speicher gelöscht. Das Passwort steht nun nicht mehr zum Kopieren zur Verfügung.	

Tabelle 36: Testszenario 1050 Accountpassword kopieren

Nr.	Testszenario	T1060
Testszenario	Account löschen (delete profile)	
Tester	Jérémie Equey	
Testdatum	04.12.2020	
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Account vorhanden	
Ablauf	Programm starten Einloggen mit Master-Passwort und Master-Passphrase Account selektieren «Delete Profile» betätigen	

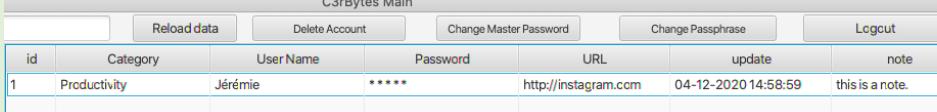
Erwartetes Ergebnis	In der Hauptansicht ist der Account nicht mehr sichtbar In der Datenbank wurde der korrekte Eintrag gelöscht
Ist-Ergebnis	In der Hauptansicht ist der Account nicht mehr sichtbar In der Datenbank wurde der korrekte Eintrag gelöscht und ist nicht mehr vorhanden.  Abbildung 59: Eintrag ist sichtbar  Abbildung 60: Antrag ist nicht mehr sichtbar
Anmerkungen	

Tabelle 37: Testszenario 1060 Account löschen

Nr. Testszenario	T1070
Testszenario	Master-Account löschen
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt
Ablauf	«Master Account löschen» betätigen Dialogfenster mit Ja bestätigen
Erwartetes Ergebnis	Die Datenbank wird zurückgesetzt bzw. ist leer Der Datenbankverzeichnis sowie die Datei (c3r.c3r) werden gelöscht. Der Masteraccount ist nicht mehr gesetzt
Ist-Ergebnis	Die Einträge in der Tabelle sind gelöscht. Das Datenbankverzeichnis ist vom Computer gelöscht. Die Datei mit dem Verschlüsselungsschlüssel für die Datenbankpasswörter (c3r.c3r) ist entfernt worden. Der Masteraccount ist nicht mehr gesetzt und bei einem Neustart muss der User ein Master-Passwort setzen.

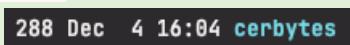

Abbildung 61: cerbytes ist der Datenbankverzeichnis


Abbildung 62: Die Datei, die den Schlüssel enthält, um die Passwörter der Datenbank zu verschlüsseln, bzw. zu entschlüsseln

```
6148 Dec  4 16:04 .DS_Store
512 Dec  4 16:13 .git
487 Dec  4 14:42 .gitignore
512 Dec  4 16:09 .idea
1097 Nov 28 14:24 PWfile.txt
935 Nov 28 14:31 README.md
1034 Dec  2 01:10 build.gradle
 64 Dec  4 16:12 db
```

Abbildung 63: Beide wurden erfolgreich gelöscht.



Abbildung 64: New Login View wird gestartet (beim nächsten Start)

	<pre>6148 Dec 4 16:04 .DS_Store 512 Dec 4 16:13 .git 487 Dec 4 14:42 .gitignore 512 Dec 4 16:09 .idea 1097 Nov 28 14:24 PWfile.txt 935 Nov 28 14:31 README.md 1034 Dec 2 01:10 build.gradle 64 Dec 4 16:12 db</pre>
<p>Abbildung 63: Beide wurden erfolgreich gelöscht.</p>	
Anmerkungen	Unter Windows kann die Protokolldatei nicht gelöscht werden. Es scheint, dass dies an einem Derby-Bug liegen könnte (https://stackoverflow.com) oder dass das System die Ressource blockiert. Trotz unserer Versuche konnten wir dieses Problem für das Windows-System nicht beheben.

Tabelle 38: Testszenario 1070 Master-Account löschen

Nr. Testszenario	T1080
Testszenario	Pen-Testing des Master-Accounts
Tester	Jérémie
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Account vorhanden
Ablauf	Mittels Cracking-Tools das Master-Passwort brute-force Mittels Cracking-Tools die Master-Passphrase brute-force um das Passwordfile zu entschlüsseln Mittels Cracking-Tools die Datenbank versuchen zu dumpen
Erwartetes Ergebnis	Alle Bemühungen sollten fehlschlagen
Ist-Ergebnis	Die Datenbank kann nicht mit einem falschen Passwort gestartet werden. Das Passwort hat 36^{64} mögliche Werte, die log_2 $(36^{64}) = 330$ Bit-Schlüsselstärke entsprechen, die physikalisch nicht zu knacken ist. Die Sicherheit hängt von der Wahl des Passwortes ab (password123 ist nicht empfehlenswert).
Anmerkungen	JunitTest: <ul style="list-style-type: none">- BruteForceBootPasswordTest() bestanden (wenn das BootPassword sich nicht in der Passwordliste befindet). Ein falsches Bootpasswort erlaubt es nicht, die Datenbank zu entschlüsseln oder zu starten. Sobald die Datenbank entschlüsselt ist, stützt sich die Sicherheit auf das Tupel Benutzer+Passwort der Datenbank (das bei jeder Abfrage ausgefüllt werden muss). Die Passwörter innerhalb der Datenbank sind nie im Klartext, außer wenn die Passwörter kopiert oder Passwörter in den entsprechenden Feldern anzeigen werden). Die Passwörter sind ansonsten zu jeder Zeit geschützt.

Tabelle 39: Testszenario 1080 Penetration testing des Master-Accounts

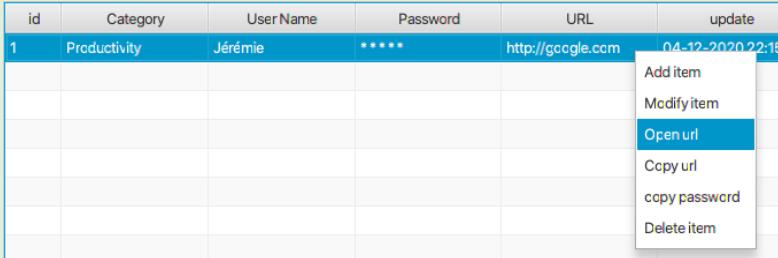
Nr. Testszenario	T1090
Testszenario	Link mit Default-Browser öffnen
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Account vorhanden URL in besagtem Account vorhanden
Ablauf	Account selektieren «Open URL» klicken
Erwartetes Ergebnis	Der Default-Browser soll sich öffnen und die gefragte URL anzeigen, sofern der Dienst verfügbar ist
Ist-Ergebnis	Firefox (Default Browser auf dem Test-Computer) hat sich geöffnet und die Webseite geöffnet (https://google.com). 
Anmerkungen	

Tabelle 40: Testszenario 1090 Link mit Default-Browser öffnen



Abbildung 66: Firefox hat den richtigen Link geöffnet.

Nr. Testszenario	T1100
Testszenario	Passwort generieren
Tester	Jérémie Equey
Testdatum	04.12.20202
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt
Ablauf	Account gemäss T1030 anlegen, aber das Passwort nicht manuell eingeben  Auf « » drücken Im Generatorbildschirm Länge und Zeichensatz wählen «Generieren» drücken «Speichern» drücken

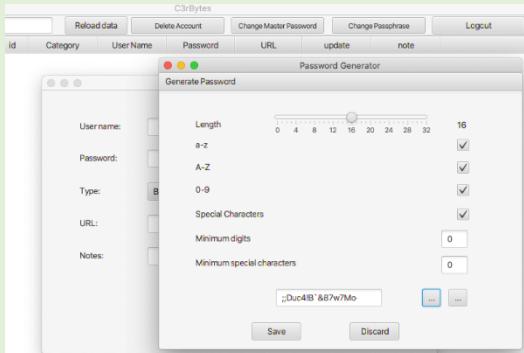
Erwartetes Ergebnis	Generatorbildschirm öffnet sich Länge und Zeichensätze lassen sich wählen Passwort wird korrekt generiert Passwort wird dem Account zugewiesen
Ist-Ergebnis	Generatorbildschirm hat sich geöffnet. Länge und Zeichensätze konnten sich wählen lassen. Passwort wurde korrekt generiert Passwort wurde dem Account zugewiesen
	 <p>Abbildung 67: PasswordGenerator-View wurde geöffnet und das Passwort generiert.</p>

Tabelle 41: Testszenario 1100 Passwort generieren

Nr. Testszenario	T1110
Testszenario	Logout
Tester	Jérémie
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm gestartet und erfolgreich eingeloggt Hauptansicht geöffnet und sichtbar
Ablauf	«Logout» betätigen
Erwartetes Ergebnis	Die Datenbankverbindung wird geschlossen Das Passwortfile ist verschlüsselt Die Fenster schliessen sich Der Prozess ist erfolgreich beendet
Ist-Ergebnis	«Log out»-Button wurde geklickt. Die Datenbankverbindung wurde geschlossen Das Passwortfile wurde verschlüsselt (shutdown=true) Die Fenster haben sich geschlossen Der Prozess ist erfolgreich beendet
Anmerkungen	JUnit: - shutdownDBTest() wurde bestanden.

Tabelle 42: Testszenario 1110 Logout

Nr. Testszenario	T1120-a
Testszenario	Ungültiges Login Masterpasswort
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm geschlossen
Ablauf	Programm öffnen Ungültiges Master-Passworteingeben
Erwartetes Ergebnis	Nach jedem erfolglosen Versuch wird der Nutzer benachrichtigt und nach dem dritten Fehlversuch wird das Programm geschlossen
Ist-Ergebnis	Nach Drei ungültigen Fehlversuchen wurde das Programm geschlossen.
	 <p>The screenshot shows a Mac OS X style window titled "C3rBytes Login Masterpassword". It features a logo with a shield and the text "C3rBytes". Below it says "Your Password Manager is locked. Please enter your master password to unlock and ...". There is a text input field, a "Login" button, and a "Logout" button. At the bottom, it says "Login failed. 2 attempts left".</p>
Anmerkungen	

Tabelle 43: Testszenario 1120-a Ungültiges Login Masterpasswort

Nr. Testszenario	T1120-b
Testszenario	Ungültiger Login Master-Passphrase
Tester	Jérémie Equey
Testdatum	04.12.2020
Voraussetzung	Masteraccount gesetzt Programm geschlossen
Ablauf	Programm öffnen Gültiges Master-Passwort eingeben Ungültige Master-Passphrase eingeben
Erwartetes Ergebnis	Nach jedem erfolglosen Versuch wird der Nutzer benachrichtigt und nach dem dritten Fehlversuch wird das Programm geschlossen
Ist-Ergebnis	Das Programm wurde nach Drei Fehlversuchen terminiert.
	 <p>The screenshot shows a Mac OS X style window titled "C3rBytes Login Master Passphrase". It features a logo with a shield and the text "C3rBytes". Below it says "Your Password Manager is still locked. Please enter your masterpassphrase to unlock and continue.". There is a text input field, a "Login" button, and a "Logout" button. At the bottom, it says "Login failed. 2 attempts left".</p>
Anmerkungen	

Tabelle 44: Testszenario 1120-b Ungültige Login Master-Passphrase

2.20.3. JUnit Tests

Die Sprache Java verfügt über ein Test-Framework. Dies ermöglicht uns, Teile unseres Codes automatisch zu testen, um zu sehen, ob die Logik so funktioniert, wie sie sollte. Dazu verwenden wir das JUnit-Framework (Paket org.junit).

Die wichtigsten Funktionen folgender Pakete wurden vollständig oder teilweise getestet:

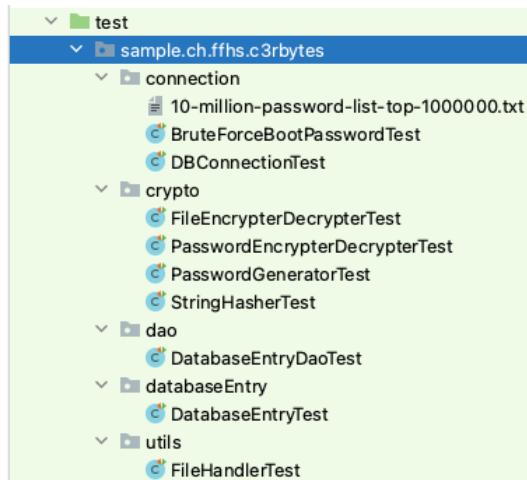


Abbildung 70: Teststruktur

1. databaseEntry



Abbildung 71: DatabaseEntryTest

2. dao

DatabaseEntryDaoTest	
f	helperTest DatabaseEntryDao
f	passworDB String
f	dbName String
m	setup() void
m	createAnEntry() DatabaseEntry
m	setEncryptionTest() void
m	changeBootPasswordTest() void
m	setupUserDBWithPasswordTest() void
m	resetUserDBWithPasswordTest() void
m	connectTest() void
m	disconnectTest() void
m	saveTest() void
m	updateTest() void
m	shutdownDBTest() void
m	deleteTest() void
m	clean() void

Abbildung 72: DatabaseEntryDaoTest

3. connection

DBConnectionTest	
f	newBootPassword String
f	passworDB String
f	dbName String
f	localValue String
m	setup() void
m	setEncryptionTest() void
m	changeBootPasswordTest() void
m	setupUserDBWithPasswordConnectionTest() void
m	resetUserPwdTest() void
m	dbConnectTest() void
m	dbDisconnectTest() void
m	dbExecuteUpdateTest() void
m	dbExecuteQueryTest() void
m	shutdownDBTest() void
m	deleteTest() void
m	clean() void

Abbildung 73: DBConnectionTest

4. crypto

PasswordGeneratorTest		
f	charSet	ArrayList<Integer>
m	laengentest()	void
m	char_lower_test()	void
m	char_upper_test()	void
m	char_digits_test()	void
m	char_specials_test()	void
m	char_lower_upper_test()	void
m	char_lower_upper_digits_test()	void
m	char_lower_upper_digits_special_test()	void
m	clear()	void

Abbildung 74: PasswordGeneratorTest

PasswordEncrypterDecrypterTest		
f	UTF_8	Charset
m	correctText()	void
m	wrongText()	void
m	decryptPassword()	void

Abbildung 75: PasswordEncrypterDecrypterTest

FileEncrypterDecrypterTest		
f	UTF_8	Charset
m	correct_pass()	void
m	wrong_pass()	void
m	decryptFile()	void

Abbildung 76: FileEncrypterDecrypterTest

StringHasherTest		
m	samePWHasherTest()	void
m	differentPWHasherTest()	void

Abbildung 77: StringHasherTest

5. utils

FileHandlerTest		
f	UTF_8	Charset
m	fileHandlerTest()	void

Abbildung 78: FileHandlerTest

2.20.4. Testergebnisse

Alle Tests wurden bestanden.

1. databaseEntry		
	✓ DatabaseEntryTest	155 ms
	✓ getPasswordTest()	22 ms
	✓ getDummyIdTest()	
	✓ getUsernameTest()	1ms
	✓ setCreationDateTest()	83 ms
	✓ setUrlTest()	6 ms
	✓ setLastUpdateTest()	2 ms
	✓ setPasswordTest()	3 ms
	✓ setUsernameTest()	1ms
	✓ setNoteTest()	4 ms
	✓ getNoteTest()	3 ms
	✓ getLastUpdateTest()	9 ms
	✓ getCreationDateTest()	1ms
	✓ setIdTest()	15 ms
	✓ getIdTest()	1ms
	✓ setDummyIdTest()	1ms
	✓ getDescriptionTest()	1ms
	✓ getUrlTest()	2 ms

Abbildung 79: Testresultate databaseEntry

2. dao		
	✓ DatabaseEntryDaoTest	8 s 135 ms
	✓ saveTest()	1s 517 ms
	✓ shutdownDBTest()	361 ms
	✓ resetUserDBWithPasswordTest()	301 ms
	✓ disconnectTest()	277 ms
	✓ updateTest()	1s 117 ms
	✓ setupUserDBWithPasswordTest()	313 ms
	✓ connectTest()	2 ms
	✓ deleteTest()	1s 351 ms
	✓ setEncryptionTest()	260 ms
	✓ changeBootPasswordTest()	2 s 636 ms

Abbildung 80: Testresultate dao

3. connection		
	✓ DBConnectionTest	6 s 155 ms
	✓ shutdownDBTest()	1s 276 ms
	✓ setupUserDBWithPasswordConnectionTest()	960 ms
	✓ dbConnectTest()	475 ms
	✓ dbDisconnectTest()	4 ms
	✓ resetUserPwdTest()	40 ms
	✓ dbExecuteQueryTest()	449 ms
	✓ deleteTest()	619 ms
	✓ setEncryptionTest()	538 ms
	✓ changeBootPasswordTest()	1s 701 ms
	✓ dbExecuteUpdateTest()	93 ms
	✓ BruteForceBootPasswordTest	528 ms
	✓ bruteForceBootPasswordTest()	528 ms

Abbildung 81: Testresultate connection

4. crypto		
	✓ StringHasherTest	55 ms
	✓ differentPWHasherTest	37 ms
	✓ samePWHasherTest	18 ms

Abbildung 82: Testresultate crypto

✓ ✓ PasswordEncrypterDecrypterTest	1s 556 ms
✓ decryptPassword()	550 ms
✓ wrongText()	529 ms
✓ correctText()	477 ms
✓ ✓ PasswordGeneratorTest	29 ms
✓ char_digits_test()	7 ms
✓ char_lower_test()	3 ms
✓ char_lower_upper_test()	3 ms
✓ char_specials_test()	4 ms
✓ char_lower_upper_digits_special_test()	3 ms
✓ char_upper_test()	3 ms
✓ char_lower_upper_digits_test()	2 ms
✓ laengentest()	4 ms
✓ ✓ FileEncrypterDecrypterTest	381 ms
✓ correct_pass()	269 ms
✓ wrong_pass()	112ms

Abbildung 83: Testresultate crypto

5. utils



Abbildung 84: Testresultate utils

2.21. Installationsanleitung

Wir gehen davon aus, dass der Benutzer ein Java-SDK auf seinem System installiert und konfiguriert hat (mindestens JDK 11).

2.21.1. Installation via GitLab

Um unser Repository zu installieren und Ihre Entwicklungsumgebung zu konfigurieren, führen Sie bitte die folgenden Schritte aus:

Die Erläuterungen sind allesamt für IntelliJ IDEA (von JetBrains) Version 2020.3 zum aktuellen Zeitpunkt erstellt worden.

1. Klonen Sie das Repository: <https://git.ffhs.ch/jeremie.equey/c3rbytes.git>
 - a. Main Branch: start_over_merge
2. Importieren Sie das Projekt in IntelliJ

Öffnen Sie "Project Structure":

1. In Projekteinstellungen: Projekt: konfigurieren Sie den Projektnamen, das SDK, die Projektsprachstufe (z. B. 8) und den Ausgangsordner für kompilierte Klassen.

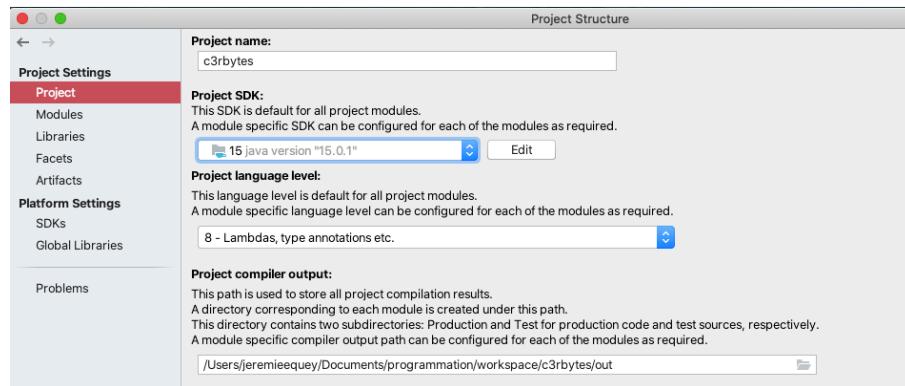


Abbildung 85: Projekt konfigurieren

1. Importieren Sie Bibliotheken für Derby, JavaFX (abhängig von Ihrem OS), hamcrest
 - a. Im Repo finden Sie einen Ordner lib, aus dem Sie die Bibliothek für Derby (Derby/*) und für Hamcrest ¹(hamcrest/*) importieren können.
 - b. JavaFx-Bibliotheken herunterladen: <https://gluonhq.com/products/javafx/> (wählen Sie Ihr SDK entsprechend Ihrem Bone) und importieren Sie es als Bibliothek (javafx-sdk-15.0.1/lib/*)

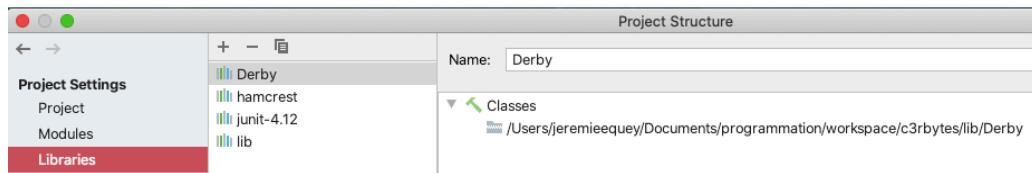


Abbildung 86: Libraries importieren

2. In "Module": Markieren Sie den Ordner src als Quellordner, test als Testordner, gui als Ressourcenordner. Prüfen Sie dann im Tab "Dependencies", ob Ihre Bibliotheken importiert wurden. Importieren Sie sie gegebenenfalls. Klicken Sie auf ok.

¹ Hamcrest ist ein Framework zum Schreiben von Matcher-Objekten, die es erlauben, 'Match'-Regeln deklarativ zu definieren. Wir verwenden hamcrest zum Vergleich einer Zeichenkette in der Klasse PasswordGeneratorTest (Paket crypto). (<http://hamcrest.org/JavaHamcrest/>)

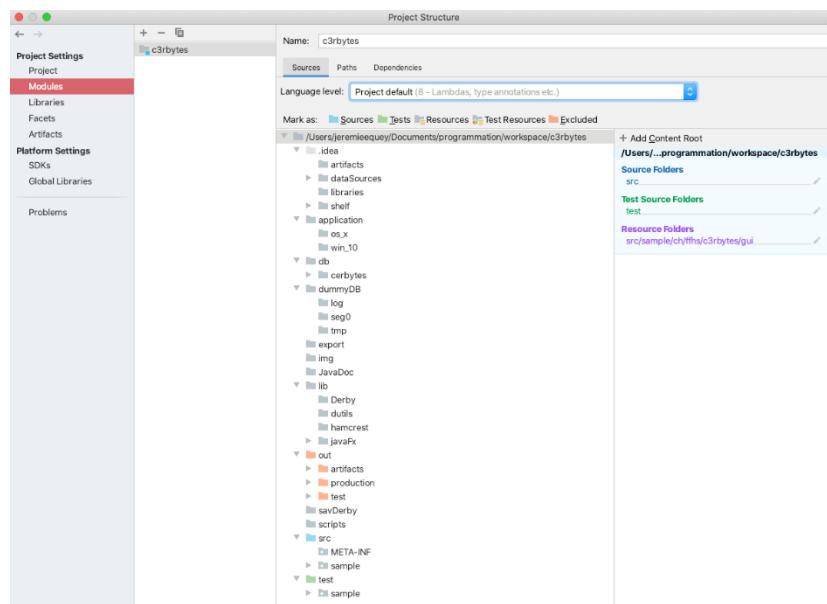


Abbildung 87: Sources

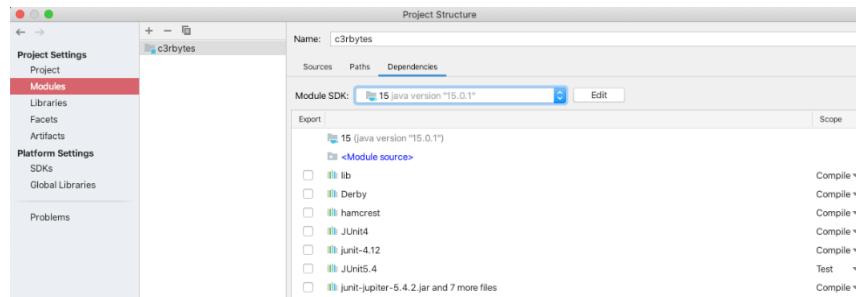


Abbildung 88: Dependencies

3. Klicken Sie dann auf "Build project". Gehen Sie auf den Tab "run", gehen Sie auf "Edit configuration". Fügen Sie, die Parameter der VM ein:

a. --module-path %PATH_TO_FX% --add-modules javafx.controls,javafx.fxml
 --module-path
 /Users/jeremieeqey/Documents/programmation/workspace/c3rbytes/lib/javaFx/javaFx-sdk-11.0.2/lib --add-modules javafx.controls,javafx.fxml

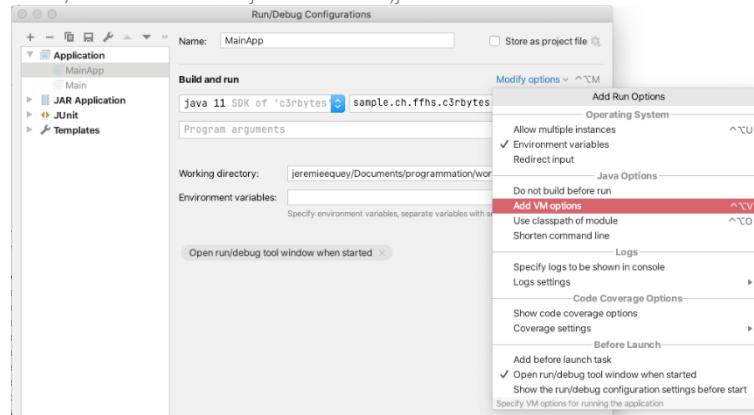


Abbildung 89: add VM options

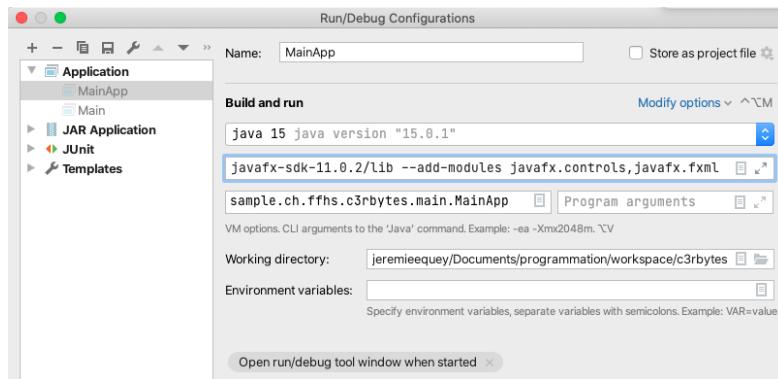


Abbildung 90: VM Optionen

4. Projekt neu erzeugen und Main.java ausführen
5. Einige Libraries müssen möglicherweise beim ersten Start noch importiert werden. Klicken Sie auf die fehlenden Bibliotheken und importieren Sie diese automatisch in Ihr Projekt (z. B. JUnit).



Abbildung 91: C3rBytes erster Login

2.21.2. Erstellung der Jar-Datei

Um die C3rBytes jar-Datei zu erzeugen, gehen Sie wie folgt vor. Die Erläuterungen basieren auf IntelliJ IDEA (von Jetbrains) in der Version 2020.3.

In «Projekt Structure»:

- Artefacts: Klicken Sie «add jar from modules with dependencies» an.

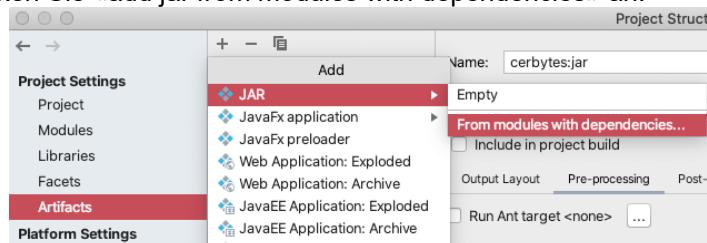


Abbildung 92: jar-Datei erstellen

- Geben Sie hier einen Namen ein und wählen Sie Ihre Main-Methode (in unserem Fall MainApp.java)

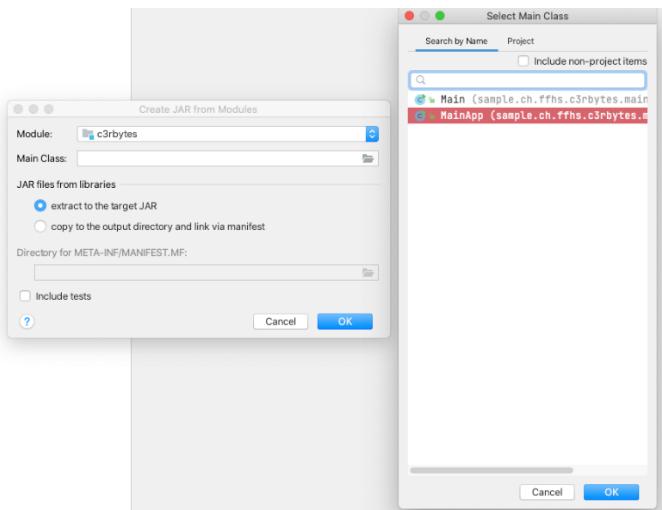


Abbildung 93: Auswahl der korrekten Main-Methode

- Prüfen Sie, ob Ihre Bibliotheken abgerufen wurden. Für javaFX müssen wir noch die *.dylib-Dateien hinzufügen (Mac Os X). Für Windows : *.dll-Dateien hinzufügen.

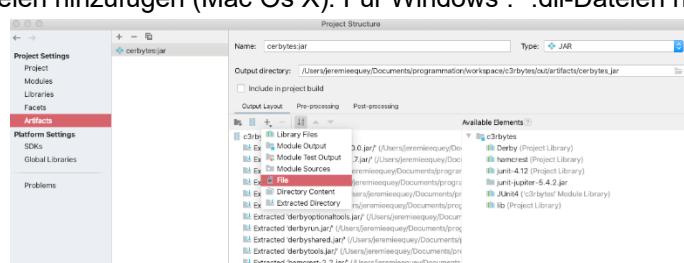


Abbildung 94: Zu importierende dvlib

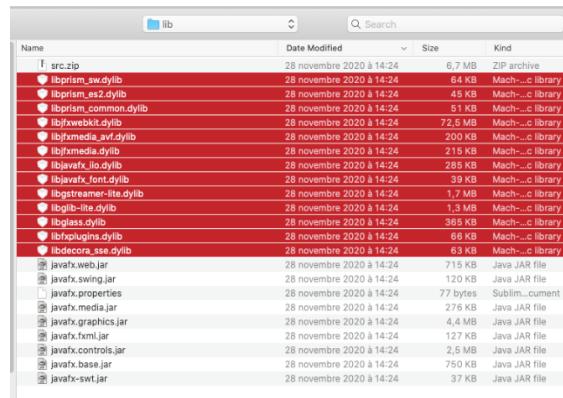


Abbildung 95: die JavaFx *.dylib

- Im Menü "Build": Artefakt erstellen. Die Jar-Datei ist in out/artifacts/cerbytes_jar/jarName.jar zu finden.



Abbildung 96: die .jar-Datei

2.21.3. Wrapping die jar für Windows (.exe)

Unter Windows muss eine .bat-Datei (run.bat) erstellt werden, die die Startanweisungen für die jar-Datei enthält (java -jar c3rbytes.jar). Dieser Vorgang startet die Applikation, öffnet aber auch ein Terminal. Dies ist nicht sehr benutzerfreundlich.

Um die jar-Datei unter Windows leichter verwenden zu können, wandeln wir sie daher in eine ausführbare Datei (.exe) um, mit Hilfe der Software Launch4j.

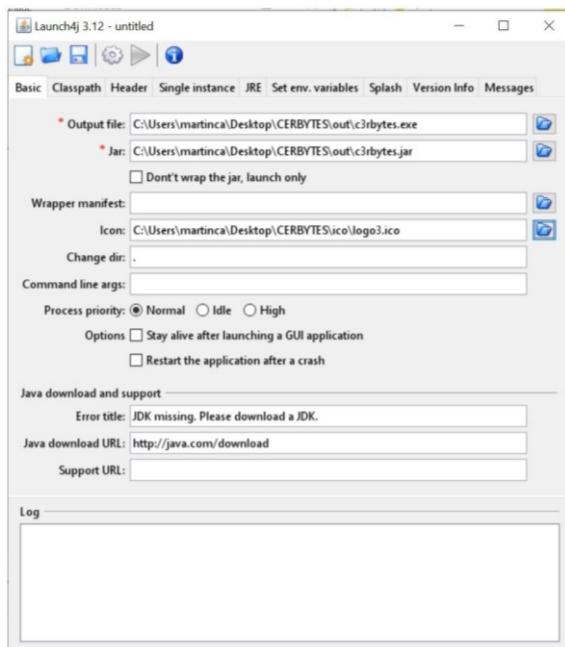


Abbildung 98: Launch4j-Konfiguration Basic

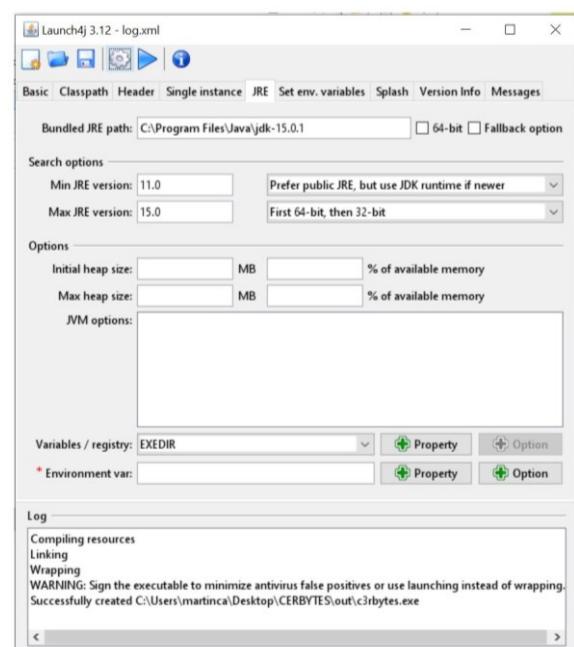


Abbildung 97: Launch4j-Konfiguration JRE

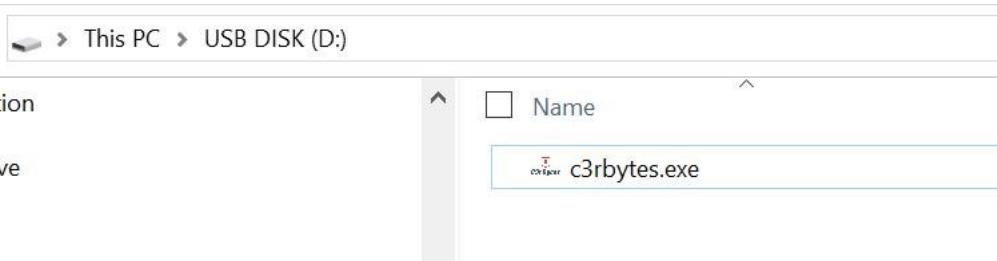


Abbildung 99: die jar-Datei, die in eine ausführbare Datei verpackt ist.

Der Benutzer kann eine Verknüpfung auf seinem Desktop erstellen, die auf das C3rBytes-Programm zeigt.

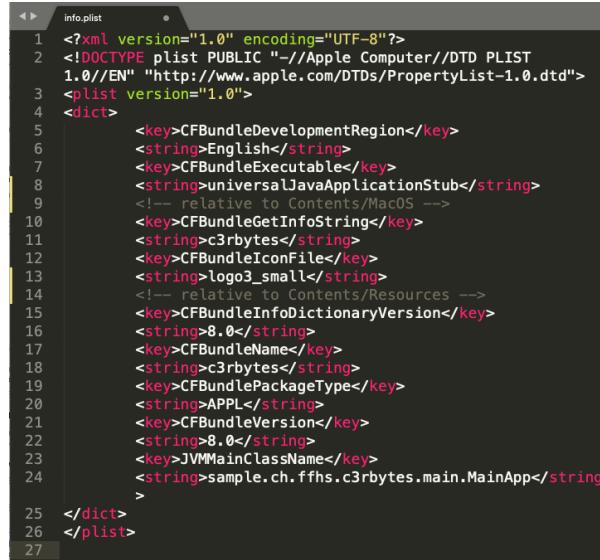
2.21.4. Wrapping die jar für macOS (.app)

Auch wenn man die Applikation durch einen Klick auf die .jar-Datei nutzen kann, können wir die Applikation in eine ausführbare Datei für macOS, das .app-Format, verpacken.

1. Wir erstellen eine neue Datei, die wir als c3rbytes.app benennen. Wir öffnen den Inhalt des Pakets (weil die Datei jetzt als Applikation) erkannt wird.
2. In der Applikation legen wir die folgende Struktur an:

```
c3rbytes.app
|
|---Contents/
|   |
|   |---Info.plist (file)
|   |
|   |---Java/ (for jar)
|   |
|   |---MacOS/ ( bash script )
|   |
|   |---Resources/ (folder for icon)
```

3. Dort platzieren wir eine MacOS-System-spezifische Eigenschaftsdatei, die angibt, wo sich die Ressourcen und unsere MainApp-Klasse befinden. Das Bash-Skript (universalJavaApplicationStub) wird verwendet, um jar-Dateien als macOS-Applikation zu verpacken (<https://github.com/tofi86/universalJavaApplicationStub>).



```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST
3  1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
4  <plist version="1.0">
5      <key>CFBundleDevelopmentRegion</key>
6      <string>English</string>
7      <key>CFBundleExecutable</key>
8      <string>universalJavaApplicationStub</string>
9      <!-- relative to Contents/MacOS -->
10     <key>CFBundleGetInfoString</key>
11     <string>c3rbytes</string>
12     <key>CFBundleIconfile</key>
13     <string>logo3_small</string>
14     <!-- relative to Contents/Resources -->
15     <key>CFBundleInfoDictionaryVersion</key>
16     <string>8.0</string>
17     <key>CFBundleName</key>
18     <string>c3rbytes</string>
19     <key>CFBundlePackageType</key>
20     <string>APPL</string>
21     <key>CFBundleVersion</key>
22     <string>8.0</string>
23     <key>JVMMainClassName</key>
24     <string>sample.ch.ffhs.c3rbytes.main.MainApp</string>
25   </dict>
26 </plist>
27

```

Abbildung 100: Inhalt der info.plist-Datei

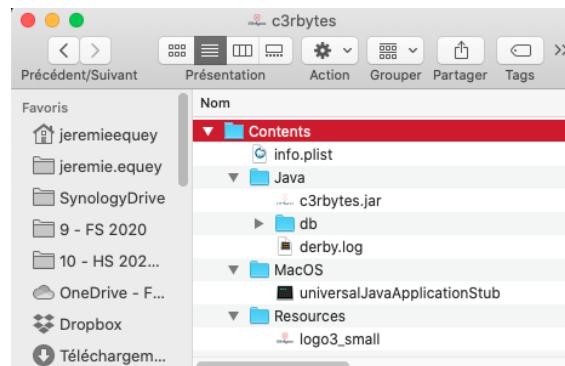


Abbildung 101: c3rbytes-Applikation und Ihre Verzeichniss hierarchie

- Das war's, unsere Applikation ist soweit konfiguriert. Das Icon erscheint in der Applikation, aber auch überall, wenn die Applikation gestartet wird.



Abbildung 102: c3rbytes.app

2.21.5. C3rBytes starten (als .app- oder .exe-Datei)

Da die Datei nicht signiert ist, müssen Sie das Starten der Anwendung validieren und bestätigen. Es ist möglich, dass bestimmte Sicherheitsstrategien des jeweiligen Unternehmens den Start der Anwendung verhindern. In diesem Fall wenden Sie sich bitte an Ihren zuständigen Administrator.

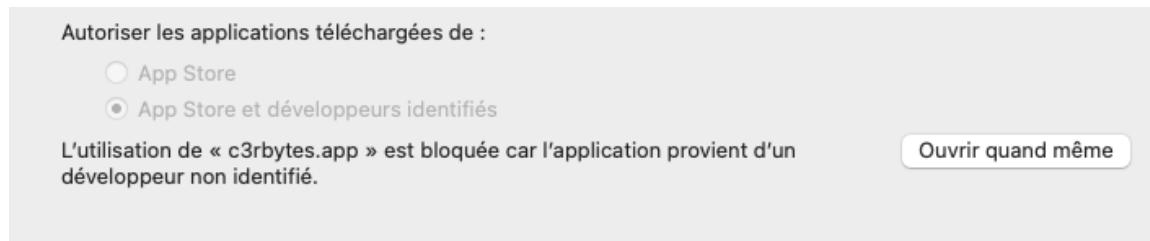


Abbildung 103: Start der Applikation bestätigen

2.21.5.1. Mac OS X (.app)

Die Applikation läuft auf macOS 10.15.7 (Catalina).

1. Stellen Sie sicher, dass Sie ein Java JDK (mindestens 11.0.9, maximal 15.0.x, empfohlen 15.0.1) auf Ihrem Computer haben.
1.1. Sonst JDK herunterladen und installieren:
<https://www.oracle.com/java/technologies/javase-jdk15-downloads.html>
2. Laden Sie die zu Ihrem Betriebssystem passende Datei herunter:
2.1. <https://git.ffhs.ch/jeremie.equey/c3rbytes/-/releases>
3. Wechseln Sie in den Ordner, in den die Datei heruntergeladen wurde, und doppelklicken Sie auf die zip-Datei:
3.1. **c3rbytes.app**
4. Das Programm wird gestartet.

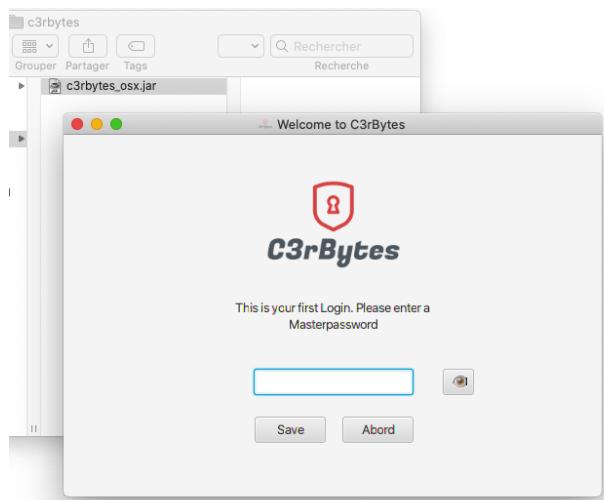


Abbildung 105: C3rBytes auf macOS (.jar)

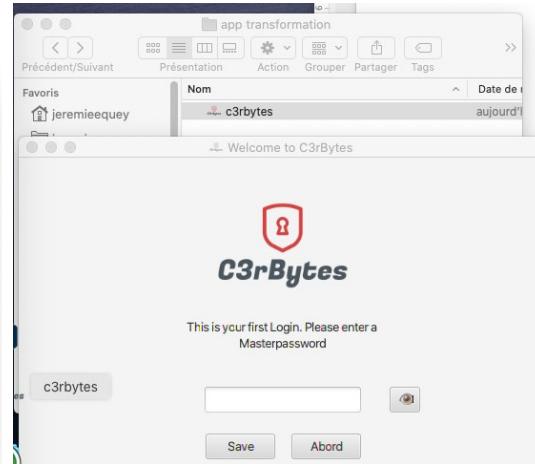


Abbildung 104: C3rBytes auf macOS (.app)

Stellen Sie bei Problemen sicher, dass Sie ein JDK installiert haben.

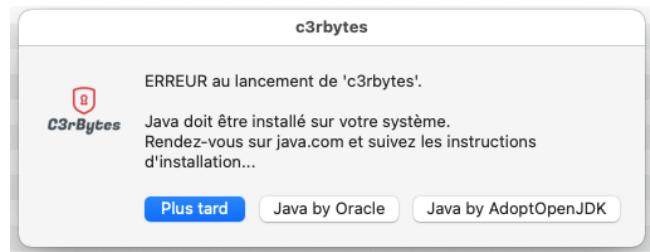


Abbildung 106: Fehlermeldung aufgrund eines fehlenden JDK

2.21.5.2. Windows (exe)

1. Die Applikation läuft unter Windows 10 (64-Bit).
2. Stellen Sie sicher, dass Sie ein Java JDK haben (mindestens 11.0.9, maximal 15.0.x, empfohlen 15.0.1).
 - 2.1. Sonst JDK herunterladen und installieren:
<https://www.oracle.com/java/technologies/javase-jdk15-downloads.html>
3. Laden Sie die zu Ihrem Betriebssystem die exe-Datei herunter:
 - 3.1. <https://git.ffhs.ch/jeremie.equey/c3rbytes/-/releases>
4. Wechseln Sie in den Ordner, in den die Datei heruntergeladen wurde, und doppelklicken Sie auf die exe-Datei:
 - 4.1. **c3rbytes.exe**
5. Das Programm wird gestartet.

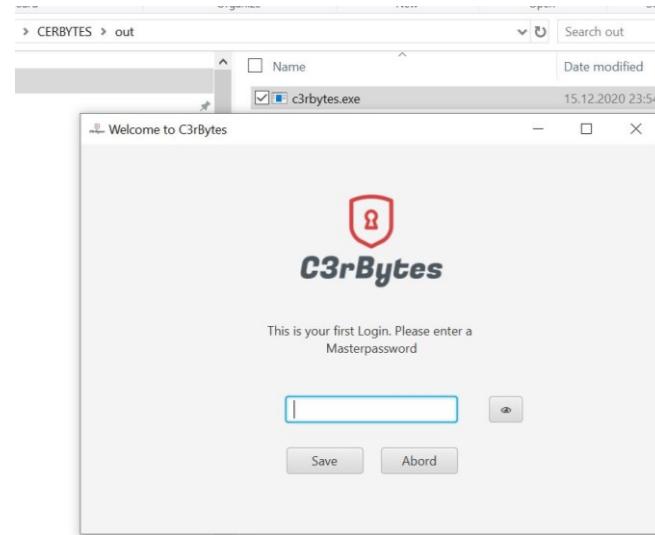


Abbildung 107: C3rBytes auf Windows

Stellen Sie bei Problemen sicher, dass Sie ein JDK installiert haben.

Die detaillierte Verwendung der Software entnehmen Sie bitte dem Handbuch.

2.22. Betriebsanleitung

2.22.1. Einleitung

2.22.1.1. Was ist C3rBytes?

C3rBytes ist ein in Java geschriebener, portabler Passwortmanager, der es erlaubt via einer Apache Derby-Datenbank Profile mit Passwörtern, Nutzernamen und direkten Links zu Webseiten zu erstellen.

2.22.1.2. Ist C3rBytes sicher?

Der Zugriff auf die C3rBytes-Datenbank ist über zwei Schritte mit einem Master-Passwort und einer Master-Passphrase gesichert. Das Master-Passwort besteht aus mind. Acht beliebigen Zeichen. Das Master-Passwort ver- und entschlüsselt die Datenbank. Die Master-Passphrase ist nicht obligatorisch, erhöht aber die Sicherheit. Die Passwörter der Profile werden in der Datenbank nicht im Klartext, sondern verschlüsselt abgelegt. Um diese zu entschlüsseln, wird die Master-Passphrase eingesetzt. Die aus einem Profil kopierten Passwörter werden nach zehn Sekunden aus der Zwischenablage gelöscht.

2.22.1.3. Disclaimer

C3rBytes wurde nach aktuellen Sicherheitsstandards implementiert. Es ist nicht auszuschliessen, dass dennoch Sicherheitsrisiken bestehen bleiben. Jeder Nutzer ist für den Schutz seiner Daten mitverantwortlich. C3rBytes unterstützt Sie lediglich in diesem Unterrfangen.

Es gibt keine 100%-ige Sicherheit.

2.22.2. Kapitel 1: Start

2.22.2.1. Windows:

C3rBytes wird als .exe Datei geliefert. Sollte das Programm zu Anfang von Windows gestoppt werden, muss man eine Ausnahme machen, um das Programm zum ersten Mal laufen zu lassen. Danach kann man es, wie jedes andere Programm mit einem Doppelklick starten.

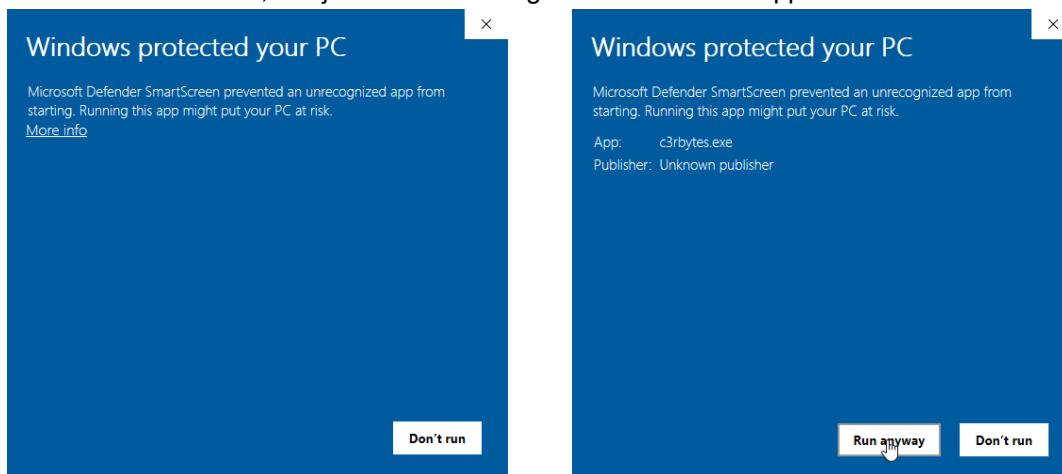


Abbildung 108: Windows Defender SmartScreen

2.22.2.2. macOS:

Um das Programm auf macOS (Catalina) ausführen zu können, klickt man einfach auf die für diese Betriebssysteme ausgelieferte c3rbytes-Applikation (c3rbytes.app) doppelt.

MacOS sollte den Start der Applikation beim ersten Mal blockieren. Gehen Sie zu den Systemeinstellungen und dann zu Sicherheit und Datenschutz. Authentifizieren Sie sich und lassen Sie die Applikation starten. Dieser Schritt muss nur beim ersten Start der Anwendung durchgeführt werden.

2.22.2.3. Erster Login:

Bei einem ersten Login wird der Nutzer aufgefordert ein Passwort für die Datenbank zu erstellen. Das Masterpasswort muss mindestens 8 Zeichen lang sein, während die Masterpassphrase beliebig lang sein kann.

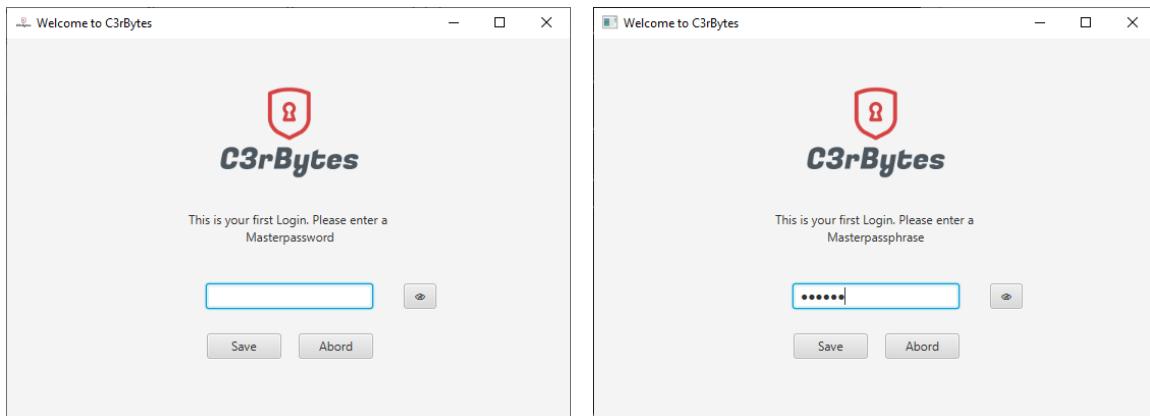


Abbildung 109: C3rBytes erster Login

2.22.2.4. Normaler Login

Nachdem ein Passwort und Passphrase gesetzt wurden, fordert das Programm den Nutzer dazu auf, die gespeicherten Zugangsdaten einzugeben. Falls die Informationen nicht korrekt sind, wird ein Fehlversuch gemeldet und zwei neue Versuche werden dem Nutzer gegeben, bevor das Programm sich selbst beendet.

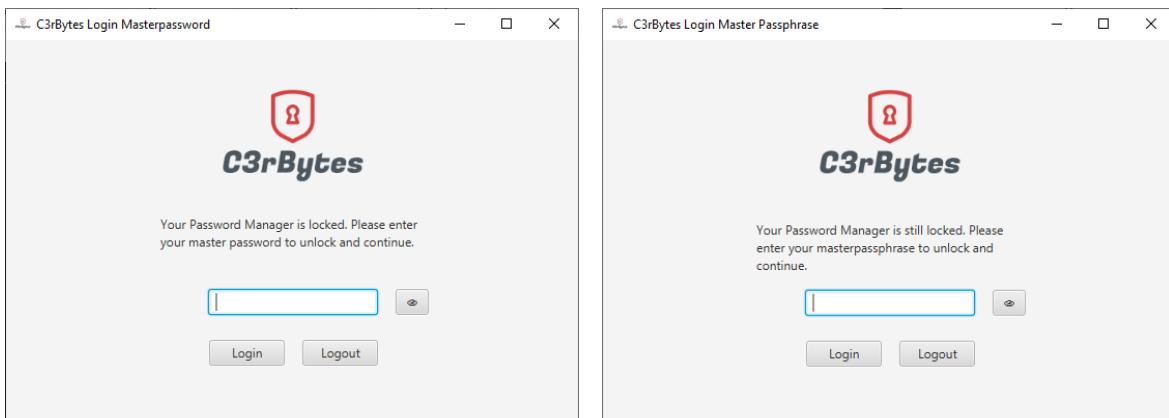
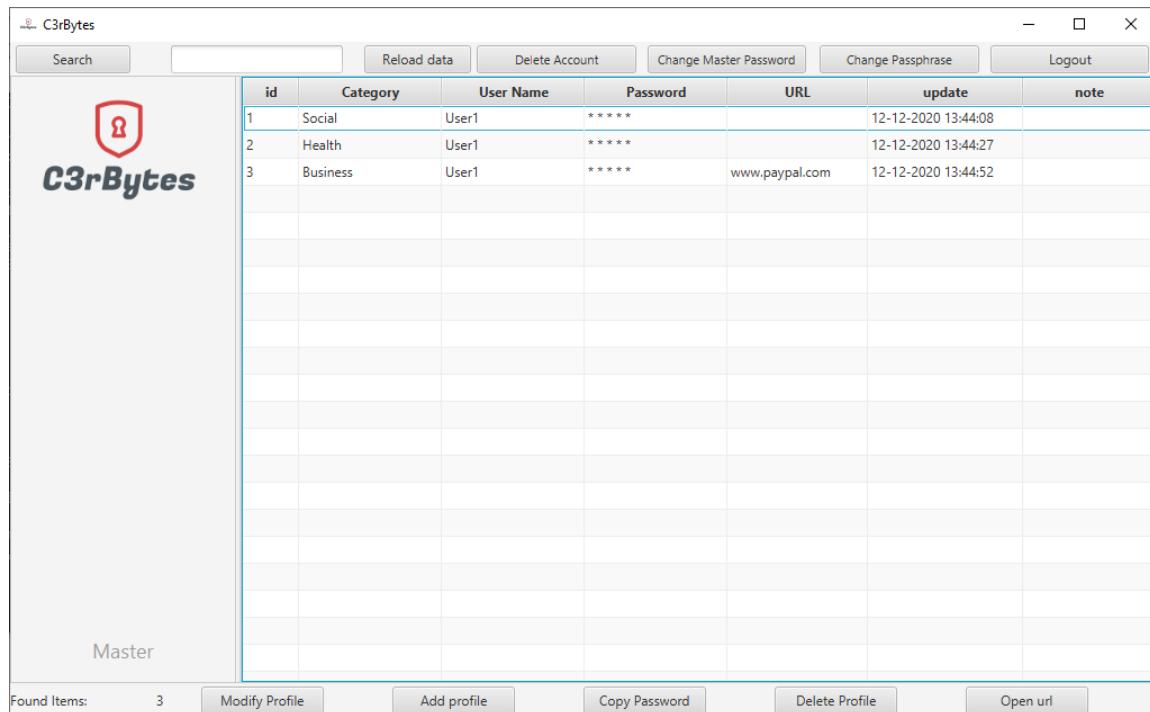


Abbildung 110: C3rBytes Login

2.22.3. Kapitel 2: Hauptmenü

Sobald die Anmeldeinformationen korrekt eingegeben werden, wird man auf das Hauptfenster gebracht. Dieses besteht aus einem oberen Balken, der eine Suchfunktion und Buttons zur Kontoverwaltung besitzt, einer Tabelle, in dem die verschiedenen Profile aufgeführt und verwaltet werden können und einem unteren Balken, der die gleichen Funktionen aufführt, die auch mit einem Rechtsklick der Maustaste verfügbar sind.



The screenshot shows the main interface of the C3rBytes application. On the left, there is a sidebar with the logo 'C3rBytes' and a shield icon. The main area contains a table with the following data:

ID	Category	User Name	Password	URL	update	note
1	Social	User1	*****		12-12-2020 13:44:08	
2	Health	User1	*****		12-12-2020 13:44:27	
3	Business	User1	*****	www.paypal.com	12-12-2020 13:44:52	

At the bottom of the interface, there are several buttons: 'Found Items: 3', 'Modify Profile', 'Add profile', 'Copy Password', 'Delete Profile', and 'Open url'.

Abbildung 111: C3rBytes Hauptview

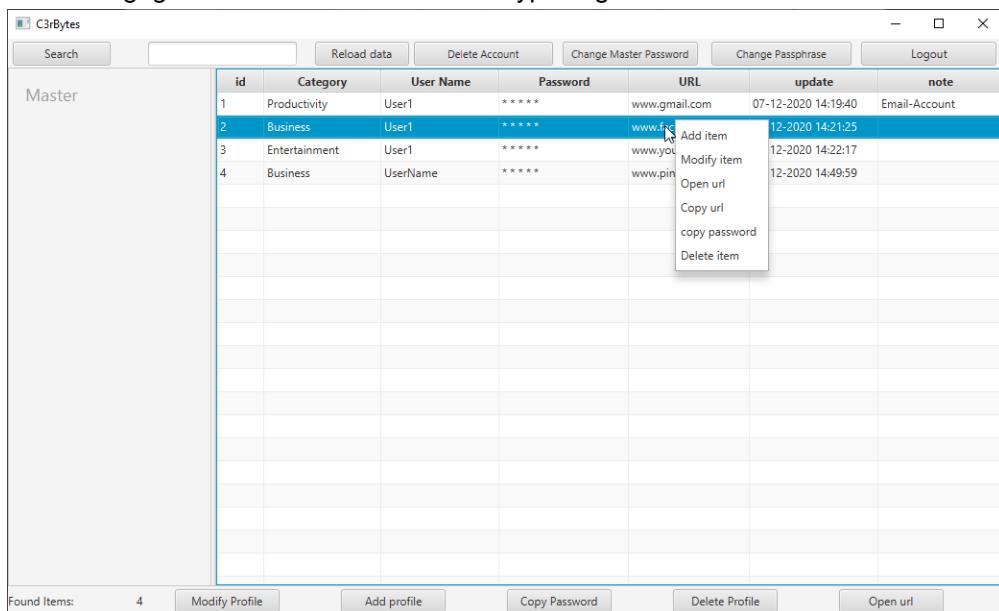
2.22.4. Kapitel 3: Profil

2.22.4.1. Profil erstellen

Um ein Profil zu erstellen gehen Sie wie folgt vor:

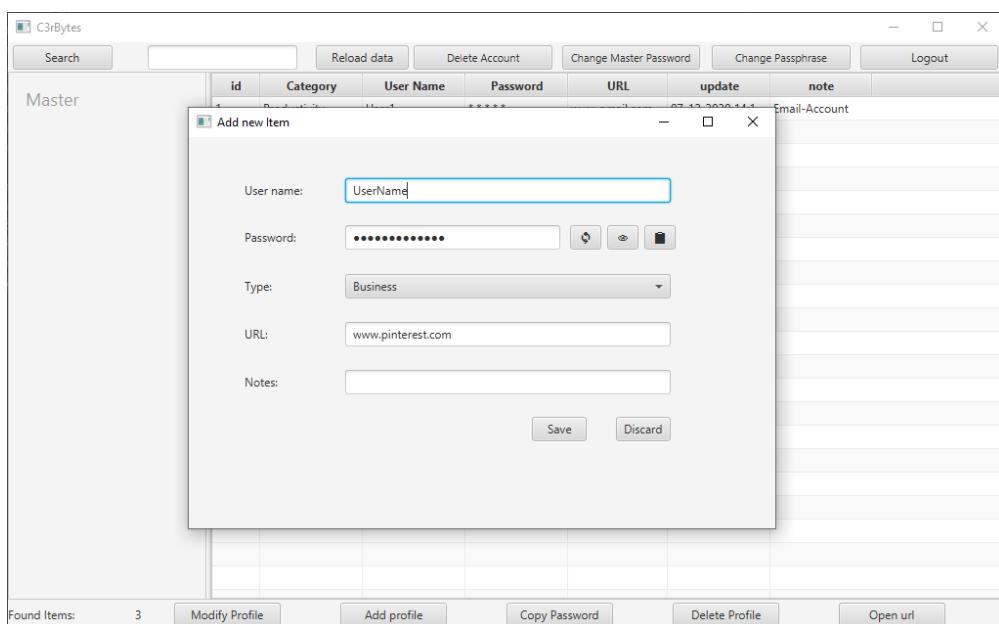
1. Klicken Sie auf «Add profile» oder mit der Rechten Maustaste auf die Tabelle und wählen «add item» aus
2. Geben Sie die Informationen des neuen Profils ein
3. Speichern Sie das Profil indem Sie auf «Save» drücken

Wichtig: Ein Profil kann gespeichert werden, wenn mindestens jeweils ein «User name», ein «Password» eingegeben werden sowie ein Profiltyp ausgewählt wird.



The screenshot shows a window titled 'C3rBytes' with a table titled 'Master'. The table has columns: id, Category, User Name, Password, URL, update, and note. There are four rows of data. A context menu is open over the fourth row (id 4), listing options: Add item, Modify item, Open url, Copy url, copy password, and Delete item. At the bottom of the window, there are buttons: Found Items: 4, Modify Profile, Add profile, Copy Password, Delete Profile, and Open url.

Abbildung 112: C3rBytes Kontextmenü



The screenshot shows the same 'C3rBytes' window as above, but with a modal dialog box titled 'Add new Item' overlaid. The dialog contains fields for User name (UserName), Password (redacted), Type (Business), URL (www.pinterest.com), and Notes. At the bottom are Save and Discard buttons. The main table below shows 'Found Items: 3'.

Abbildung 113: C3rBytes Add Item View

2.22.4.2. Passwörter generieren

1. Öffnen Sie das «Password Generator»-Fenster in der «Item View» indem Sie auf das «» Symbol klicken.
2. Wählen Sie die gewünschten Parameter für das Passwort aus.
3. Klicken Sie auf «» um das Passwort zu generieren
4. Klicken Sie auf «Save» um das Passwort im entsprechenden «Item View»-Feld zu speichern.

Wichtig: Falls Sie das Passwort später verändern, können Änderungen nicht rückgängig gemacht werden.

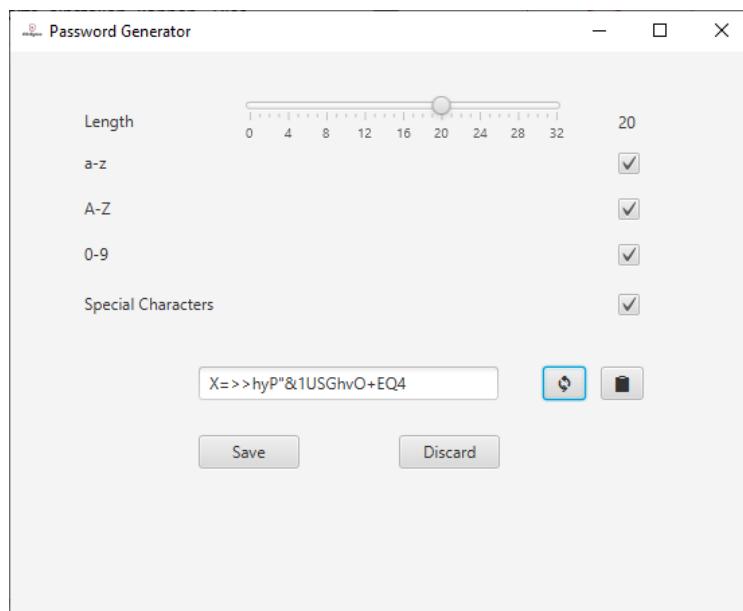


Abbildung 114: C3rBytes Passwortgenerator

2.22.4.3. Webseiten öffnen

1. Klicken Sie mit der rechten Maustaste auf das Profil, welches die zu öffnende Webseite beinhaltet.
2. Klicken Sie in der Drop-down Liste entweder auf «Open URL», um die Webseite direkt zu öffnen, oder auf «Copy URL» um die URL zu speichern und selbst in einem Browser einzufügen.

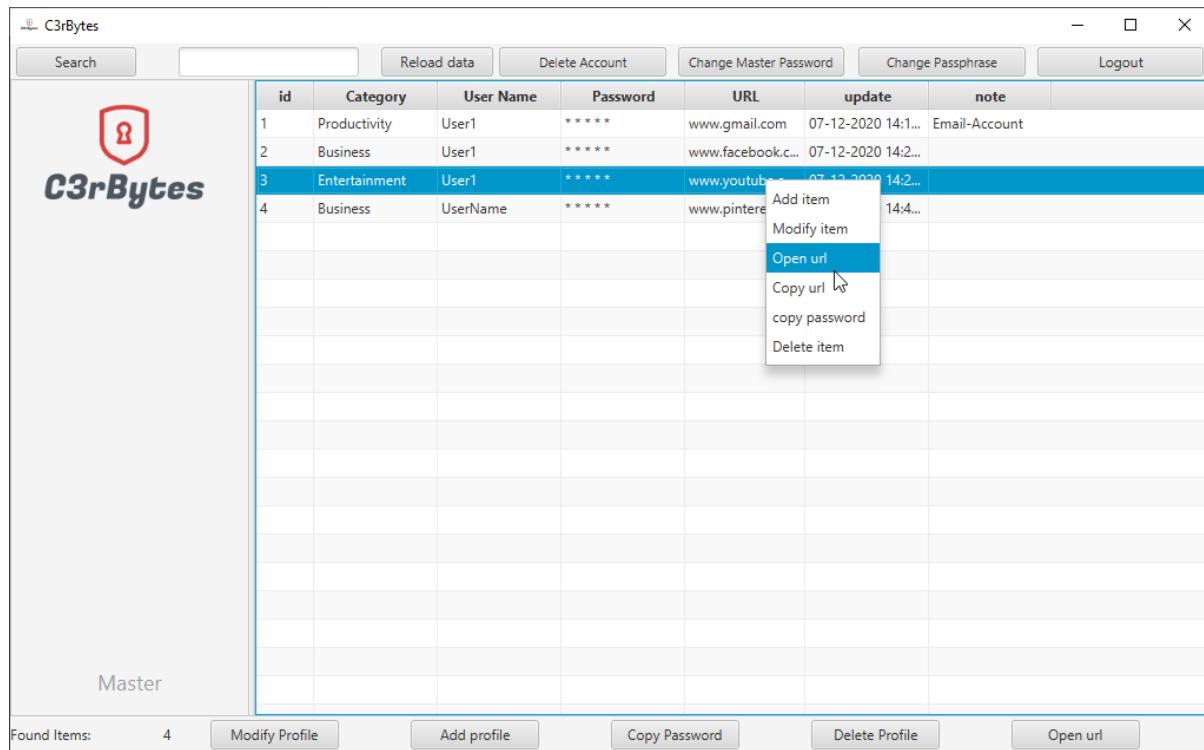


Abbildung 115: C3rBytes Kontextmenü - Open URL

2.22.4.4. Profil verändern

1. Klicken Sie mit der rechten Maustaste auf das zu verändernde Profil.
2. Wählen Sie die Option «Modify item» aus.
3. Verändern Sie die nötigen Informationen im Profil.

Wichtig: Passwortänderungen können nicht rückgängig gemacht werden. Bitte stellen Sie sicher, dass Sie das Passwort nicht mehr brauchen.

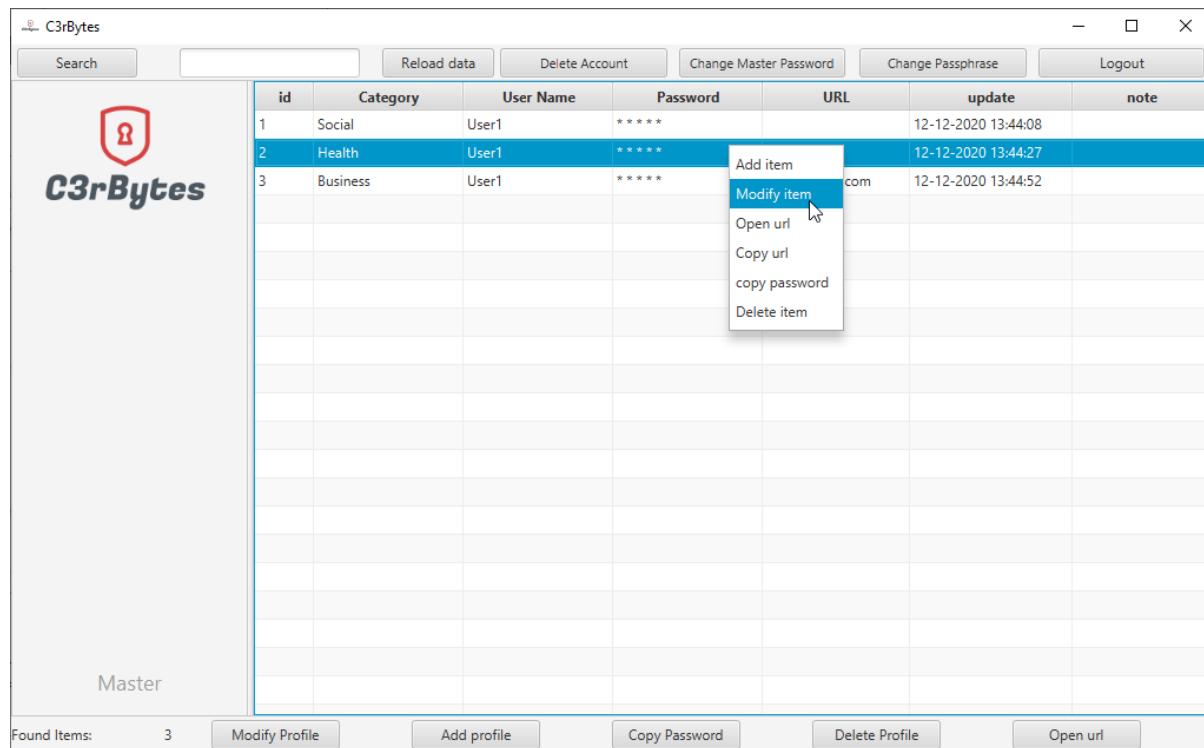


Abbildung 116: C3rBytes Kontextmenü - Modify Item

2.22.4.5. Profil löschen

1. Klicken Sie mit der rechten Maustaste auf das zu löschen Profil.
2. Wählen Sie die Option «Delete item» aus dem Drop-Down Menü aus.
3. Bestätigen Sie die Aktion im Warnungsfenster.

Wichtig: Gelöschte Profile sind können nicht wiederhergestellt werden und müssen manuell neu erfasst werden.

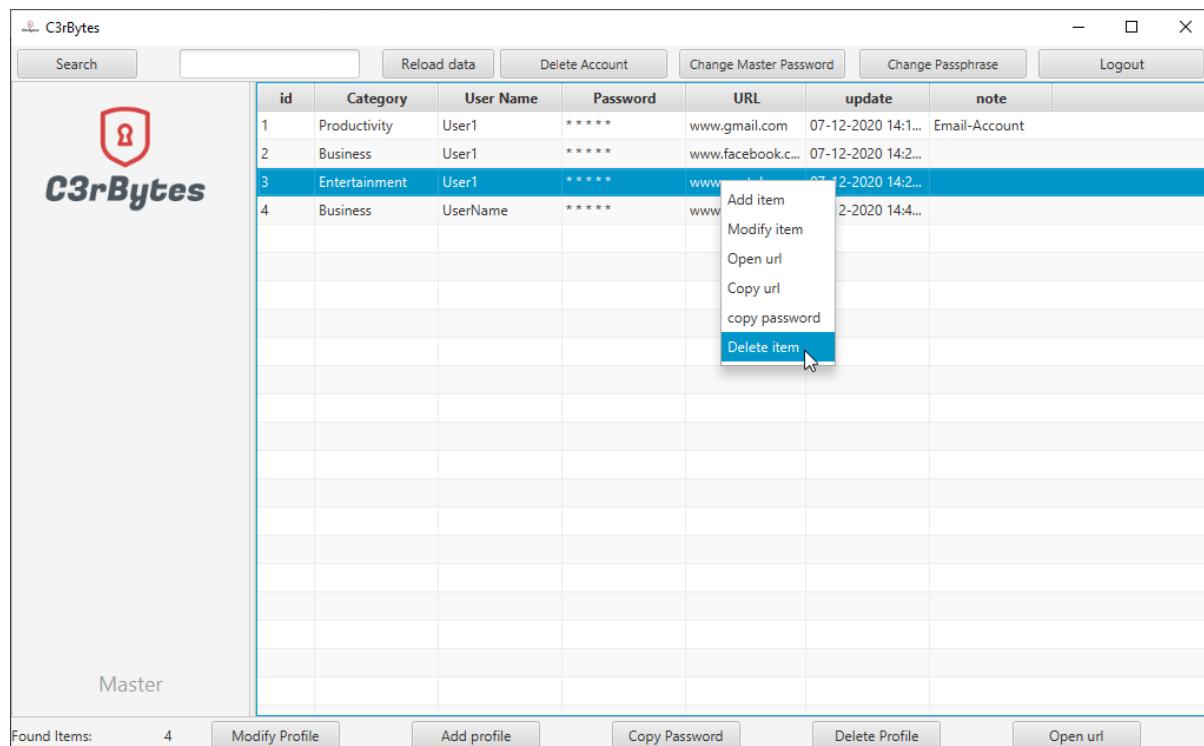


Abbildung 117: C3rBytes Kontextmenü - Delete item

2.22.5. Kapitel 4: Kontoeinstellungen ändern

2.22.5.1. Kontopasswort und -passphrase ändern

Falls man das Kontopasswort oder dessen Passphrase ändern will geht man wie folgt vor:

1. Klicken Sie auf «Change Master Password» oder «Change Passphrase».
2. Geben Sie das alte Passwort oder die alte Passphrase ein.
 - a. Das Masterpasswort muss mindestens acht Zeichen lang sein.
3. Um die neuen Kontoinformationen zu speichern, müssen Sie diese im Bestätigungsfeld ein zweites Mal eingeben.

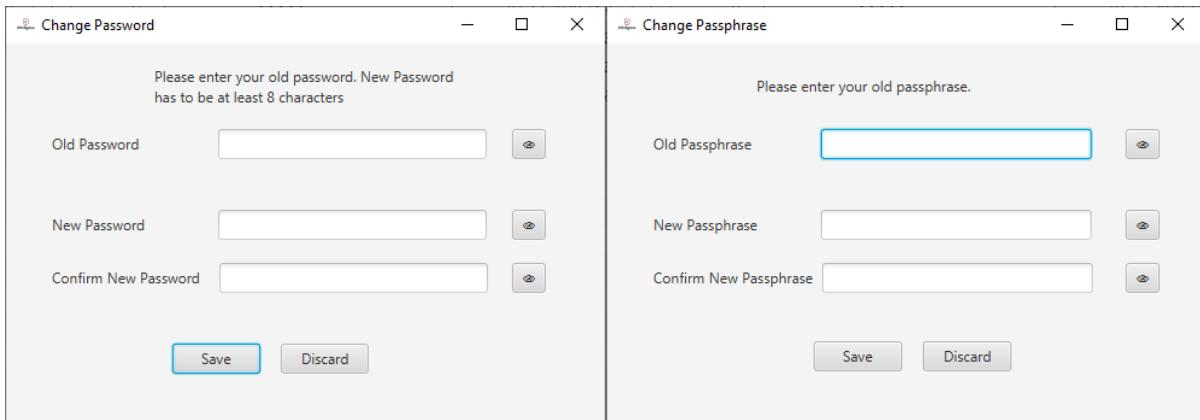


Abbildung 118: C3rBtes Kontoinformationen mutieren

2.22.5.2. Konto löschen

Wenn man das gesamte Konto und die Datenbank löschen möchte, geht man wie folgt vor:

1. Klicken Sie auf «Delete Account».
2. Ein Bestätigungsfenster wird geöffnet das den Nutzer fragt, ob das Konto gelöscht werden soll.
3. Bestätigen Sie das Löschen des Kontos.

Wenn das Konto gelöscht wurde, wird man beim nächsten Start von C3rBytes wieder aufgefordert ein neues Masterpasswort und -Passphrase einzugeben und eine neue Datenbank wird erstellt.

Wichtig: Ein gelöschtes Konto und dessen Informationen können nicht wiederhergestellt werden.

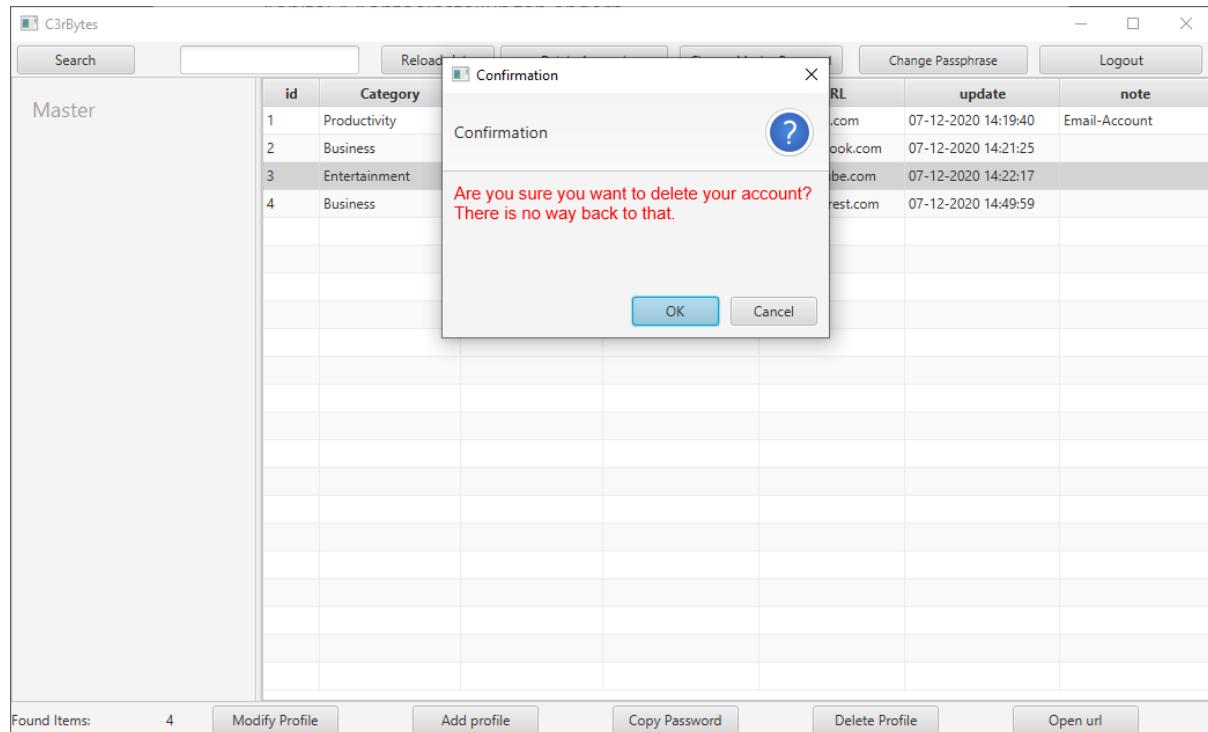


Abbildung 119: C3rBytes Konto löschen

2.23. Evaluierungsbericht

Bereits beim Kick-off zur Projektarbeit am 18. August 2020 wurde die Projektidee eines Passwort-Managers festgelegt und kurz darauf auch der Name vom Urheber der Idee, Jérémie Equey, in Absprache mit dem Team bestimmt. Die Basis für das Projekt C3rBytes war gelegt.

Das Team war begeistert und motiviert. Viele Ideen für den funktionalen Umfang wurden in die Runde geworfen. Man begann schon bald mit der strukturellen Planung, klärte die Ausgangslage ab, formulierte erste Grobanforderungen und folglich die Ziele zur Umsetzung.

Um dem dringend nötigen Austausch Rechnung zu tragen wurden wöchentliche Sitzungen mit dem Tool Teams vereinbart, was gut funktioniert hat. Weiterhin wurde OneDrive sowie weitere Tools aus der Office365-Suite für das parallele Bearbeiten von Dokumenten eingesetzt.

Rückblickend wollen wir nun einige Punkte beleuchten.

Was ist uns gelungen?

- In Anbetracht, dass zwei Drittel des Teams nicht in der IT-Branche tätig sind und auch sonst beruflich nicht mit Projekten in Kontakt geraten, können wir dennoch behaupten, dass wir unser Ziel erreicht haben, auch wenn dies nicht immer auf konventionelle bzw. lehrbuchmässige Weise stattgefunden hat.
- Der Passwort-Manager läuft robust und schützt die Daten vor unberechtigtem Zugriff. Aktuell sind uns keine Verwundbarkeiten bekannt.
- C3rBytes kann zudem auf einem USB-Stick betrieben werden und erfüllt somit ein Kann-Ziel.

Was hätte besser laufen können?

- Der Aufwand für die Kommunikation wurde unterschätzt.
- Ebenso für das Erstellen, Nachführen und Pflegen der Dokumentation. Bis zuletzt mussten Anpassungen vorgenommen werden.
- Aufgrund von keiner bis geringer praktischer Erfahrung im Projektmanagement, war es schwierig, die Aufwände und Ressourcen einzuschätzen und dementsprechend einzuplanen.
- Die Umsetzung der Massnahmen beim Projektmanagement und deren Nachverfolgung haben wir zu sehr vernachlässigt.
- Dasselbe gilt für das Konzeptionieren und Entwickeln von Software. Beim Implementieren der Software traten immer neue und ungeplante Aspekte zu Tage, welche zur Umsetzung des Ziels umgesetzt werden mussten. Dies erforderte einen beträchtlichen Mehraufwand.

Was haben wir gelernt?

- Erfahrung kann durch nichts ersetzt werden.
- Eine sorgfältige und durchdachte Planung kann viel Aufwand ersparen. (Dokumentation ist teuer, aber keine ist noch teurer).
- Regelmässiger Austausch mit den Teammitgliedern hilft Doppelspurigkeiten zu vermeiden.
- Wir haben JavaFX als neues Framework für die Erstellung von grafischen Oberflächen kennen gelernt.
- Unsere Kenntnisse im Projektmanagement, in der Softwareentwicklung, in Java, in der Kryptologie sowie im Datenbankbereich haben wir wiederauffrischen, praktisch anwenden und erweitern dürfen.
- Ein Projekt ist nie perfekt, auch wenn man alles x-fach überprüft. Man kann durch eine Rekapitulation wertvolle Erkenntnisse gewinnen, um für folgende Projekte gewappnet zu sein.

2.24. Verwendete Literatur

[Buch] M. Seidl, M. Brandsteidl, C. Huemer, G. Kappel «*UML @ Classroom*». Heidelberg: dpunkt.verlag GmbH, 2012

[Buch] H. Licher, J. Ludewig «*Software Engineering: Grundlagen, Menschen, Prozesse, Techniken*». Heidelberg: dpunkt.verlag GmbH, 2013



3. Einverständniserklärung / Selbstständigkeitserklärung

Mit seiner Unterschrift bestätigt der jeweilige Autor, seinen Teil der Dokumentation selbstständig erarbeitet oder andere Quellen gekennzeichnet zu haben sowie die Arbeit vollumfänglich gelesen und akzeptiert zu haben. Des Weiteren wird mit der Unterschrift bestätigt, dass ein Projekt mit gleichem oder ähnlichem Inhalt, weder an der FFHS noch an einer anderen Hochschule bereits eingereicht wurde oder für andere Lehrveranstaltungen als dieser eingegeben wird.

Jérémie Equey

Mersid Hazbiu

Olaf Schmidt

Datum:

Bern, 22.12.2020

Datum:

Fribourg, 22.12.2020

Datum:

Niedergesteln, 22.12.2020

Unterschrift:

Unterschrift:

Unterschrift:

4. Abbildungsverzeichnis

Abbildung 1: Rollenorganisation.....	11
Abbildung 2: Team C3rBytes	12
Abbildung 3: Terminplanung	23
Abbildung 4: Ressourcenplanung	24
Abbildung 5: Ressourcenplanung	25
Abbildung 6: Alle Use Cases.....	50
Abbildung 7: Login and Passwords management login.....	52
Abbildung 8: Actions related to an item.	54
Abbildung 9: Search and related actions	59
Abbildung 10: Nutzwertanalyse Verschlüsselungsalgorithmen	72
Abbildung 11: Nutzwertanalyse Datenbanksystem	74
Abbildung 12: Nutzwertanalyse Two-Step Authentication	78
Abbildung 13: Nutzwertanalyse UI-Framework.....	80
Abbildung 14: Architektur-Pattern MVC.....	82
Abbildung 15: Kontextdiagramm	83
Abbildung 16: Domänenmodell	85
Abbildung 17: Komponentendiagramm	86
Abbildung 18: Klassendiagramm Gesamtstruktur.....	88
Abbildung 19: Package-Struktur.....	89
Abbildung 20: Klassenmodell Main-Package.....	89
Abbildung 21: Klassendiagramm Controller Package	90
Abbildung 22 Klassenmodell Controller	90
Abbildung 23: Klassenmodell Crypto-Package	91
Abbildung 24: Klassenmodell Utils-Package	92
Abbildung 25: Klassenmodell DAO	93
Abbildung 26: Klassenmodell DatabaseEntryDao	94
Abbildung 27: Klassenmodell DatabaseEntry	95
Abbildung 28: Klassenmodell DBConnection	96
Abbildung 29: Zwei-Schritt-Zugriffssicherheit	97
Abbildung 30: Aktivitätsdiagramm Registrierung	99
Abbildung 31: Aktivitätsdiagramm Login.....	100
Abbildung 32: Aktivitätsdiagramm Neues Profil hinzufügen	101
Abbildung 33: Aktivitätsdiagramm Profil löschen	102
Abbildung 34: Sequenzdiagramm Registrierung.....	104
Abbildung 35: Sequenzdiagramm Login.....	106
Abbildung 36: Sequenzdiagramm Eintrag erstellen	107
Abbildung 37: Sequenzdiagramm Eintrag löschen	108
Abbildung 38: Sequenzdiagramm Eintrag ändern	109
Abbildung 39: Sequenzdiagramm Master-Passwort ändern	110
Abbildung 40: Sequenzdiagramm Master-Passphrase ändern	111
Abbildung 41: Verteilungsdiagramm.....	112
Abbildung 42: Entity-Relationship Diagramm	113
Abbildung 43: Entity-Relationship Diagramm	113
Abbildung 44: Datenstruktur database_entries	114
Abbildung 45: Login C3rBytes Master Password.....	118
Abbildung 46: Login C3rBytes Master-Passphrase.....	119
Abbildung 47: C3rBytes Main View	120
Abbildung 48: C3rBytes Add New Item View	120
Abbildung 49: C3rBytes Password Generator	121
Abbildung 50: Hauptansicht	123
Abbildung 51: Masterpassword ändern	125
Abbildung 52: Masterpassword wurde erfolgreich geändert (unterschiedliche Werte)	125

Abbildung 53: "Add profile" Ansicht	126
Abbildung 54: SQL-Query um die Felder in der Datenbank zu speichern.....	126
Abbildung 55: Mainview	126
Abbildung 56: View Item bevor der Änderung	126
Abbildung 57: View Item nach der Änderung.....	127
Abbildung 58: MainView mit der Änderung.....	127
Abbildung 59: Eintrag ist sichtbar.....	128
Abbildung 60: Antrag ist nicht mehr sichtbar	128
Abbildung 61: cerbytes ist der Datenbankverzeichnis.....	128
Abbildung 62: Die Datei, die den Schlüssel enthält, um die Passwörter der Datenbank zu verschlüsseln, bzw. zu entschlüsseln.....	128
Abbildung 63: Beide wurden erfolgreich gelöscht.	129
Abbildung 64: New Login View wird gestartet (beim nächsten Start).....	129
Abbildung 65: die MainView erlaubt einen Link anzuklicken.	130
Abbildung 66: Firefox hat den richtigen Link geöffnet	130
Abbildung 67: PasswordGenerator-View wurde geöffnet und das Passwort generiert.....	131
Abbildung 68: ein Fehlerversuch (MasterPassword).....	132
Abbildung 69: ein Fehlerversuch (Passphrase)	132
Abbildung 70: Teststruktur	133
Abbildung 71: DatabaseEntryTest.....	133
Abbildung 72: DatabaseEntryDaoTest	134
Abbildung 73: DBConnectionTest	134
Abbildung 74: PasswordGeneratorTest.....	135
Abbildung 75: PasswordEncrypterDecrypterTest	135
Abbildung 76: FileEncrypterDecrypterTest	135
Abbildung 77: StringHasherTest	135
Abbildung 78: FileHandlerTest.....	135
Abbildung 79: Testresultate databaseEntry	136
Abbildung 80: Testresultate dao.....	136
Abbildung 81: Testresultate connection.....	136
Abbildung 82: Testresultate crypto.....	136
Abbildung 83: Testresultate crypto.....	137
Abbildung 84: Testresultate utils	137
Abbildung 85: Projekt konfigurieren.....	138
Abbildung 86: Libraries importieren.....	138
Abbildung 87: Sources.....	139
Abbildung 88: Dependencies	139
Abbildung 89: add VM options	139
Abbildung 90: VM Optionen	140
Abbildung 91: C3rBytes erster Login.....	140
Abbildung 92: .jar-Datei erstellen	141
Abbildung 93: Auswahl der korrekten Main-Methode.....	141
Abbildung 94: Zu importierende dvlib	141
Abbildung 95: die JavaFx *.dylib	142
Abbildung 96: die .jar-Datei.....	142
Abbildung 97: Launch4j-Konfiguration JRE	142
Abbildung 98: Launch4j-Konfiguration Basic	142
Abbildung 99: die jar-Datei, die in eine ausführbare Datei verpackt ist.....	143
Abbildung 100: Inhalt der info.plist-Datei	144
Abbildung 101: c3rbytes-Applikation und Ihre Verzeichniss hierarchie.....	144
Abbildung 102: c3rbytes.app.....	144
Abbildung 103: Start der Applikation bestätigen	145
Abbildung 104: C3rBytes auf macOS (.app).....	145
Abbildung 105: C3rBytes auf macOS (.jar).....	145



Abbildung 106: Fehlermeldung aufgrund eines fehlenden JDK.....	146
Abbildung 107: C3rBytes auf Windows	146
Abbildung 108: Windows Defender SmartScreen.....	147
Abbildung 109: C3rBytes erster Login.....	148
Abbildung 110: C3rBytes Login.....	148
Abbildung 111: C3rBytes Hauptview	149
Abbildung 112: C3rBytes Kontextmenü.....	150
Abbildung 113: C3rBytes Add Item View.....	150
Abbildung 114: C3rBytes Passwortgenerator	151
Abbildung 115: C3rBytes Kontextmenü - Open URL	152
Abbildung 116: C3rBytes Kontextmenü - Modify Item.....	153
Abbildung 117: C3rBytes Kontextmenü - Delete item	154
Abbildung 118: C3rBtes Kontoinformationen mutieren	155
Abbildung 119: C3rBytes Konto löschen	156

5. Tabellenverzeichnis

Tabelle 1: Phasen.....	10
Tabelle 2: Phasenbeschreibungen	11
Tabelle 3: Meilensteine	11
Tabelle 4: Rollenverteilung	12
Tabelle 5: SWOT-Analyse	16
Tabelle 6: Risikenbeschreibung	17
Tabelle 7: Risikomatrix	18
Tabelle 8: Problemlösung inkl. Wertung	19
Tabelle 9: Probleme und Massnahmen	20
Tabelle 10: Projektstrukturplan	22
Tabelle 11: Zielgruppen	46
Tabelle 12: Stakeholders	47
Tabelle 13: UC01 Set master password	51
Tabelle 14: UC02 Login	53
Tabelle 15: UC03 enter new items	55
Tabelle 16: UC04 Modify item	56
Tabelle 17: UC05 Change master password	57
Tabelle 18: UC06 Delete user account	58
Tabelle 19: UC07 Search item	60
Tabelle 20: UC08 Open URL	61
Tabelle 21: UC09 Generate password	62
Tabelle 22: UC10 Copy password	63
Tabelle 23: UC11 Delete item	64
Tabelle 24: UC12 Login with 2SA	65
Tabelle 25: UC13 Logout	66
Tabelle 26: UC14 Set master passphrase	67
Tabelle 27: UC15 Change master password	68
Tabelle 28: Morphologischer Kasten	80
Tabelle 29: Testszenarien	122
Tabelle 30: Testszenario 1000-a Registration	123
Tabelle 31: Testszenario 1000-b Registration	124
Tabelle 32: Testszenario 1010 Login	124
Tabelle 33: Testszenario 1020 Masteraccount Daten mutieren	125
Tabelle 34: Testszenario 1030 Add Profile	126
Tabelle 35: Testszenario 1040 Account mutieren	127
Tabelle 36: Testszenario 1050 Accountpassword kopieren	127
Tabelle 37: Testszenario 1060 Account löschen	128
Tabelle 38: Testszenario 1070 Master-Account löschen	129
Tabelle 39: Testszenario 1080 Penetration testing des Master-Accounts	129
Tabelle 40: Testszenario 1090 Link mit Default-Browser öffnen	130
Tabelle 41: Testszenario 1100 Passwort generieren	131
Tabelle 42: Testszenario 1110 Logout	131
Tabelle 43: Testszenario 1120-a Ungültiges Login Masterpassword	132
Tabelle 44: Testszenario 1120-b Ungültige Login Master-Phrase	132



6. Anhang

6.1. Statusberichte

6.1.1. Statusbericht 1

PA_5, Projektarbeit, INF-P-AT005, BE1

Statusbericht 1 Gruppe 4:



Jérémie Equey, Olaf Schmidt, Mersid Hazbiu

19.09.2020

Ausgangslage

Täglich erfahren wir von spezialisierten Websites, dass ein Datenleck mit persönlichen Informationen ins Netz gestellt wurde. Häufig enthalten diese Daten Benutzernamen, E-Mail-Adressen und Passwörter. Diese im Web verloren gegangenen Daten stellen dann eine Gefahr für die Besitzer dieser Passwörter dar, da ihre Informationen nicht mehr sicher sind und somit kompromittiert werden könnten.

Diese gesammelten Informationen können dann in einem Wörterbuch (Wordlist) landen und werden unter anderem zur Durchführung von Brute-Force-Angriffen verwendet.

Die Konsequenzen eines solchen Datenverlusts können für die betroffenen Benutzer katastrophal sein. Ein Hacker könnte z.B. das Passwort eines E-Mail-Kontos ändern, so dass der ursprüngliche Benutzer keinen Zugriff mehr auf seine Daten hat.

Auch wenn Unternehmen vor einigen Jahren begonnen haben, Strategien zur Sicherung von Konten (2- oder Multi-Faktor-Authentifizierung) zu implementieren, ist es immer noch wichtig, eine gute Hygiene mit Ihren Passwörtern zu haben. Ein Passwort ist ein Passwort zur einmaligen Verwendung. Es sollte komplex sein. Komplexität bedeutet jedoch für einen Menschen oder einen Computer leider nicht dasselbe. *Bz54!_@* ist vielleicht für uns kompliziert, jedoch nicht für einen Computer. Außerdem ist seine Entropie äusserst schlecht. Auf der anderen Seite ist *@/_I-am-A-PINK-Licorne-3012_#* ein Passwort mit einer besseren Entropie und für den Benutzer leicht zu merken.

Daher ist die Benutzung eines Passwort-Managers heutzutage eine Pflicht, wenn man sein digitales Leben schützen und nicht riskieren will, alle seine Daten durch das Recycling alter Passwörter zu verlieren.

Was versteht man unter einem Password Manager?

Ein Passwort-Manager ist eine Art Software, der es einem Benutzer ermöglicht, seine Passwörter zu verwalten, entweder durch Zentralisierung aller seiner Identifikatoren und Passwörter in einer Datenbank (Portfolio) oder durch deren Berechnung auf Anfrage. Der Passwort-Manager ist durch ein eindeutiges Passwort geschützt, so dass man sich nur ein Passwort merken muss.

Der Benutzer, der so von dem Zwang befreit wird, sich verschiedene Passwörter zu merken, kann auch kompliziertere (und daher robustere) wählen und für jedes Konto oder jedes Dokument ein anderes Passwort haben (so dass, wenn eines der Passwörter abgefangen wird, die anderen Konten oder Dokumente nicht angreifbar gemacht werden). (Wikipedia)

Gesamtstatus

Der aktuelle Stand entspricht der Planung. Gegenwärtig sehen wir keine Probleme. Das Team hat sich gegenseitig kennen gelernt und fühlt sich durch die Herausforderung sehr motiviert. Das Team hat mit einem anhaltenden Rhythmus begonnen: Es möchte eine Teamsitzung pro Woche leiten, um Fragen, Richtung, Probleme, usw. zu diskutieren (bisher 3 Sitzungen).

Die Initialisierungsphase wurde abgeschlossen. Das Team befindet sich im Moment am Ende der Vortudiephase. Der Projektauftrag wurde vom Team validiert. Das Dokument muss noch vom Team geprüft (bis am 26. September 2020). Die Anforderungen wurden ebenfalls definiert und von allen Teilnehmern akzeptiert. Die Gruppe führt derzeit erste Überlegungen darüber durch, wie die Applikation Informationen speichern wird. Dies ist ein wichtiger Schritt, denn es ist das Kerngeschäft unserer Anwendung, nämlich die nahtlose Sicherheit der gespeicherten Daten.

Dieses Projekt verwendet das Phasen-Vorgehensmodell des «Handbuch Projektmanagement» vom Springer Verlag 2011. Dabei werden die Phasen sinnvoll auf das Projekt angepasst.

Für die Softwareentwicklung hat sich der Projektausschuss für das Wasserfallmodell entschieden. Es ist ein lineares (nicht iteratives) Vorgehensmodell, das sich für dieses Projekt mit vorgegebenen Anforderungen und Leistungen präzise beschreiben lässt.



Nr.	Phase	Beschreibung
1	Initialisierung	Definition der Aufgaben, Ziele und der Zielgruppe. Anforderungen aufnehmen, Ergebnisse formulieren und Meilensteine planen.
2	Vorstudie	Erstellung einer Grobplanung für die Umsetzung der Projektidee. Definition der Ablaufplanung, Projekt-Organisation und des Projektauftrags.
3	Konzept	Gesamtkonzept fertigstellen, Lösungsansätze (Varianten) aufzeigen und bewerten. Detaillierte Projektlösung planen und erarbeiten.
4	Realisierung	Umsetzung des Detailplans. Einführung planen, Controlling durchführen und Abweichungen kommunizieren
5	Abschluss	Übergabe organisieren, Abschlussbericht und Schlussrechnung erstellen. Projektdokumentation ergänzen, Projektbeurteilung und «Lessons Learned» verarbeiten. Projektdokumentation archivieren.

id	Name	Status	%	Massnahmen
1	Initialisierung	Planmäßig	100	
2	Vorstudie	Planmäßig	80	Alle Dokumente
3	Konzeption	Planmäßig	10	Erste Überlegungen zum Lösungsansatz (zB. Daten vs File)
4	Realisierung	Nicht gestartet	0	
5	Projektabschluss	Nicht gestartet	0	

Status Termine inkl. Bewertung / allfällige Massnahmen

Das Team hat wöchentliche (online) Teamsitzung durchgeführt. Folgende Termine wurden durchgeführt:

id	Name	Datum	Ort	Bemerkung
1	Kick-Off Meeting	21.08.2020	Online	Kein Apero L
2	Teamsitzung	03.09.2020	MS-Team	
3	Teamsitzung	10.09.2020	MS-Team	
4	Teamsitzung	18.09.2020	MS-Team	Vorbereitung Status-Meeting 1
5	Status-Meeting 1	19.09.2020	MS-Team	
6	Teamsitzung	19.09.2020	MS-Team	Nachbearbeitung Status-

Status Lieferobjekte inkl. Bewertung / allfällige Massnahmen

MS	Name	Lieferobjekte	Start	Deadline	Status	Bemerkung
1	Initialisierung		01.09.2020	10.09.2020		
		Projektziele		10.09.2020	planmäßig	
		Anforderungen (Grob)		10.09.2020	planmäßig	
		Meilensteinen planen		10.09.2020	planmäßig	
		1. Rollenverteilung		10.09.2020	planmäßig	
2	Vorstudie		11.09.2020	18.09.2020		<i>Bis am 26. September verlängert</i>
		Projektauftrag		18.09.2020	planmäßig	
		Planung (initiale)		18.09.2020	planmäßig	
		Projektstrukturplan		18.09.2020	i.	
		Riskikoanalyse		18.09.2020	Bearbeitung	
		Status-Bericht 1		18.09.2020	planmäßig	reviewed
		Präsentation		18.09.2020	planmäßig	reviewed
		2. Rollenverteilung		19.09.2020	planmäßig	
3	Konzeption		20.09.2020	18.10.2020		Neues Datum 27.09.2020
		Fachanforderungen		18.10.2020		
		Nichtfunktionale Anforderungen		18.10.2020		
		Funktionale Konzeption		18.10.2020		
		Technische Konzeption		18.10.2020		
		Testkonzept		18.10.2020		
4	Realisierung		27.10.2020	13.12.2020		
		Quellcode		13.12.2020		
		Unit-Test		13.12.2020		
		Testprotokolle		13.12.2020		
		Java Dokumentation		13.12.2020		
		Installationsanleitung		13.12.2020		

	(Booklet) Produkt Vorbereiten zur Abgabe (VM)	13.12.2020
5	Projektabchluss	14.12.2020 20.12.2020
	Benutzerhandbuch Projektdokumentation	20.12.2020 16.12.2020

Status Qualität inkl. Bewertung / allfällige Massnahmen

Name	Bewertung (Qualität)	Bemerkung	Massnahmen
Projektziele	5	Angenommen	
Anforderungen (Grob)	5	Angenommen	
Planung (initiale)	4	Angenommen	
Statusbericht 1	5	Angenommen	
Präsentation SB 1	5	Angenommen	
Projektauftrag	(5)	In Bearbeitung	Bis am 26. September
Projektstrukturplan		In Bearbeitung	Bis am 26. September
Riskikoanalyse		In Bearbeitung	Bis am 26. September
Pflichtenheft		In Bearbeitung	Bis am 26. September

Skala: 6: hervorragend / 5: gut / 4: genügend / 3: ungenügend / 2: sehr ungenügend / 1: abgelehnt

Nächste Schritte / Änderungsanträge

Der Beginn der Konzeptionsphase war für den 19. September (nach Status-Meeting 1) vorgesehen. Wir geben uns jedoch eine weitere Woche Zeit, um alle Lieferobjekte der Vorstudienphase zu überprüfen und zu akzeptieren. Daher wird die nächste Phase am 26. September beginnen, was uns zwei Monate für die Konzeption und Realisierung gibt. Die letzten zwei Wochen sind für den Abschluss des Projekts und die Lieferung der Dokumentation reserviert. Wir planen außerdem, 10 Tage zu reservieren. Unsere Erfahrung hat uns gelehrt, dass es bei der Software-Entwicklung ein Problem gibt, wenn alles nach Plan läuft.

In dieser Phase gibt es keine Änderungsanträge (ausser Verlängerung der Voranalysephase)

Aktualisierte Rollenorganisation

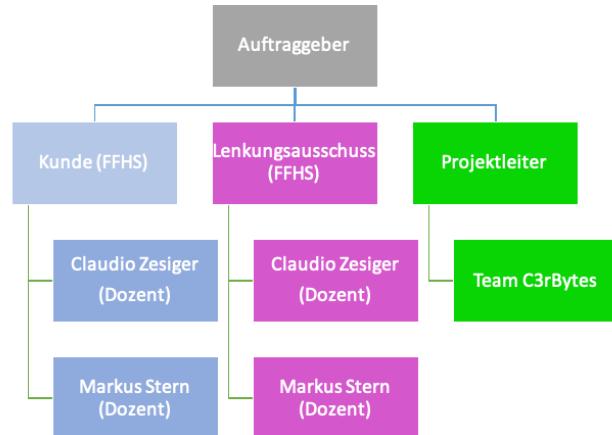


Abbildung 1: Rollenorganisation

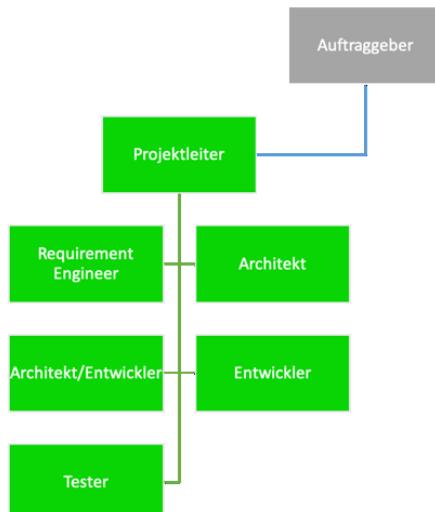


Abbildung 2: Team C3rBytes

Aktuelle Rollenverteilung:

Vorname	Name	aktuelle Rolle	bis
Jérémie	Equey	P	19.09.2020
Olaf	Schmidt	R	19.09.2020
Mersid	Hazbiu	R	19.09.2020

Ab 20.09.2020

Vorname	Name	aktuelle Rolle	bis
Jérémie	Equey	A	20.10.2020
Olaf	Schmidt	P	20.10.2020
Mersid	Hazbiu	A	20.10.2020

**Abk.** Beschreibung

PL	Projektleiter
E	Entwickler
A	Architekt
A/E	Architekt/Entwickler
RE	Requirement Engineer
TE	Tester

Diskussion

Fragen:

6.1.2. Statusbericht 1 – Reflexion**PA_5, Projektarbeit, INF-P-AT005, BE1****Reflexion zum Statusbericht 1 Gruppe 4:****Jérémie Equey, Olaf Schmidt, Mersid Hazbiu**

19.09.2020

Anwesende:

- Dozenten: Claudio Zesiger, Markus Stern
- Studenten: Olaf Schmidt, Mersid Hazbiu, Jérémie Equey

An unserem 1. Statusmeeting am 19.10.2020 (09:55-10:40) werden die folgenden Massnahmen zur Anwendung gelangen.

Projektmanagement

Nr	Inputs von Dozenten	Massnahmen

1	Fehlende Nutzwertanalysen wurden angemerkt (3 Varianten)	Für wichtige Funktionen des Projektes werden Nutzwertanalysen erstellt, z.B. Datenbank-/Persistenzsystem, GUI-Framework, 2FA, Sicherheitsalgorithmen <i>Bemerkung:</i> der Lösungszyklus ist unabhängig von den Zielen.
2	Ausser den Funktionszielen fehlten andere Zielarten. Soziale Ziele, Ökonomische Ziele, Terminziele, Ergebnisziele etc.	Es werden zusätzliche Ziele erstellt und diese entsprechend definiert. Ein neuer Änderungsantrag wird, wenn nötig eingereicht.
3	Die Arbeitspakete für den Strukturplan waren zu grob definiert.	Die Arbeitspakete werden im Projektplan genauer aufgenommen, um auch deren Fortschritt präziser verfolgen zu können.
4	Ausgangslage präzisieren	Auf die Ausgangslage wurde nicht eingegangen. Diese ist im Anhang zu finden.
5	Begründeter Lösungsansatz	Gewählter Lösungsansatz (gewählte Variante) ist „kurz und bündig“ schriftlich zu begründen. Der Lösungsansatz muss aus der Problemsituation bzw. Auftragssituation im Sinne des Problemlösungszyklus (Problem identifizieren, Lösungssuche, Auswahl) approximativ nachvollziehbar sein. Es muss erkennbar sein, dass die Lösungsvariante(n) aus der Situationsanalyse (IST-Situation) und der Zielformulierung (Projektziele) abgeleitet ist/sind. Zur Darstellungsform werden
6	Rollenverteilung	Namen der Dozenten anpassen.

Software Engineering

Nr.	Inputs von Dozenten	Massnahmen
1	Libraries können auf anderen Libraries aufbauen	Einen Dependency-Check via OWASP. Wir können Online-Tools verwenden, um dies zu überprüfen
2	Lösungen müssen effektiv erklärt bzw. rechtfertigt werden, damit in der Bewertung keine Probleme auftauchen	Wenn sich das Team für eine Non-standard Lösung beschliesst, muss dieser Entscheid gut dokumentiert und argumentiert sein, z.B. Eine verschleierte Datei zur Persistierung anstatt einer

Ausgangslage

Täglich erfahren wir von spezialisierten Websites, dass via Datenleck persönliche Informationen ins Netz gestellt wurden. Häufig enthalten diese Daten Benutzernamen, E-Mail-Adressen und Passwörter. Diese im Web verloren gegangenen Daten stellen dann eine Gefahr für die Besitzer dieser Passwörter dar, da ihre Informationen nicht mehr sicher sind und somit kompromittiert werden könnten.

Diese gesammelten Informationen können dann in einem Wörterbuch (Wordlist) landen und werden unter anderem zur Durchführung von Brute-Force-Angriffen verwendet.

Die Konsequenzen eines solchen Datenverlusts können für die betroffenen Benutzer katastrophal sein. Ein Hacker könnte z.B. das Passwort eines E-Mail-Kontos ändern, so dass der ursprüngliche Benutzer keinen Zugriff mehr auf seine Daten hat.

Auch wenn Unternehmen vor einigen Jahren begonnen haben, Strategien zur Sicherung von Konten (2- oder Multi-Faktor-Authentifizierung) zu implementieren, ist es immer noch wichtig, eine gute Hygiene mit Ihren Passwörtern zu haben. Ein Passwort ist ein Passwort zur einmaligen Verwendung. Es sollte komplex sein. Komplexität bedeutet jedoch für einen Menschen oder einen Computer leider nicht dasselbe. *Bz54!_@* ist vielleicht für uns kompliziert, jedoch nicht für einen Computer. Außerdem ist seine Entropie äusserst schlecht. Auf der anderen Seite ist *@/_I-am-A-PINK-Licorne-3012_#* ein Passwort mit einer besseren Entropie aber für den Benutzer schlechter zu merken.

Daher ist die Benutzung eines Passwort-Managers heutzutage eine Pflicht, wenn man sein digitales Leben schützen und nicht riskieren will, alle seine Daten durch das Recycling alter Passwörter zu verlieren.

Lösungsansatz (Entwurf)

Wie in der Ausgangslage erklärt, möchten wir als Zielsetzung einen Passwordmanager entwickeln, damit die Benutzer (unsere Zielgruppe) ihre Passworthygiene verbessern können, sowie bessere und stärke Passwörter verwenden. Der Passwort-Manager soll auch intuitiv zu benutzen sein.

Wir haben uns aus den folgenden Gründen für eine Standalone-Applikation entschieden:

1. Eine einfache verschlüsselte Excel-Datei, oder anderer Dateityp, ist nicht intuitiv für die Nutzer und kann schnell verloren gehen oder im schlimmsten Fall entschlüsselt werden.
2. Eine Webapp erfordert einen zu grossen zeitlichen und ressourcetechnischen Aufwand, der nicht mit dem Umfang eines Projektes innerhalb eines Semesters an der FFHS vereinbar ist. Des Weiteren ist eine Webapp weitaus anfälliger auf Sicherheitsattacken.
3. Eine Standalone-Applikation ist ein in sich geschlossenes System und kann für Nutzer so designt werden, dass sie sich nur um die minimalen Funktionen, d.h. Merken des Masterpasswortes, kümmern müssen.
4. Java muss als Entwicklungssprache zu mindestens für die Hälfte des Projektes verwendet werden. Webapps verwenden primär andere Entwicklungssprachen (z.B. JavaScript für das Frontend), und wie in Punkt 1 beschrieben, verwendet Java per se nicht.

Als Konsequenz haben wir uns für eine Standalone-Applikation entschieden, da diese die Anforderungen am besten erfüllt.

6.1.3. Statusbericht 2**PA_5, Projektarbeit, INF-P-AT005, BE1****Statusbericht 2 Gruppe 4:**

Jérémie Equey, Olaf Schmidt, Mersid Hazbiu

14.11.2020

Gesamtstatus

Der aktuelle Stand entspricht mehr oder weniger die Planung. Wir mussten die Konzeptionsphase um eine Woche verschieben in der Überzeugung, dass wir die in der Entwurfsphase verlorene Woche wieder aufholen konnten. Das war nicht möglich, weil der Arbeitsaufwand in dieser Phase sehr gross war und wir mit dem *Try-and-Error-Prinzip* etwas Zeit verloren haben. Jetzt haben wir wieder das gleiche Problem, nämlich dass wir mit einer Woche Verspätung in die Realisierungsphase eintreten. Aufgrund unserer mangelnden Erfahrung bei der Durchführung von IT-Projekten dieser Größenordnung (über einen Zeitraum von sechs Monaten) haben wir bestimmte Elemente unterschätzt.

Die Zusammenarbeit innerhalb des Teams ist gut, auch wenn wir feststellen, dass dieses Modul sehr intensiv, sogar zu intensiv ist und dass dies Auswirkungen auf die Arbeit hat, die wir für die anderen Module dieses Semesters zu leisten haben.



Nr.	Phase	Beschreibung
1	Initialisierung	Definition der Aufgaben, Ziele und der Zielgruppe. Anforderungen aufnehmen, Ergebnisse formulieren und Meilensteine planen.
2	Vorstudie	Erstellung einer Grobplanung für die Umsetzung der Projektidee. Definition der Ablaufplanung, Projekt-Organisation und des Projektauftrags.
3	Konzept	Gesamtkonzept fertigstellen, Lösungsansätze (Varianten) aufzeigen und bewerten. Detaillierte Projektlösung planen und erarbeiten.
4	Realisierung	Umsetzung des Detailplans. Einführung planen, Controlling durchführen und Abweichungen kommunizieren
5	Abschluss	Übergabe organisieren, Abschlussbericht und Schlussrechnung erstellen. Projektdokumentation ergänzen, Projektbeurteilung und «Lessons Learned» verarbeiten. Projektdokumentation archivieren.

id	Name	Status	%	Massnahmen
1	Initialisierung	Planmäßig	100	
2	Vorstudie	Planmäßig	100	
3	Konzeption	Nicht planmäßig	90	
4	Realisierung	gestartet	10	Es besteht ein Risiko, dass die Kann-Kriterien nicht 100% implementiert werden
5	Projektabschluss	Nicht gestartet	0	

Status Termine inkl. Bewertung / allfällige Massnahmen

Das Team hat wöchentliche (online) Teamsitzung durchgeführt. Folgende Termine wurden durchgeführt:

id	Name	Datum	Ort	Bemerkung
1	Kick-Off Meeting	21.08.2020	Online	Kein Apero 😊
2	Teamsitzung	03.09.2020	MS-Team	
3	Teamsitzung	10.09.2020	MS-Team	
4	Teamsitzung	18.09.2020	MS-Team	Vorbereitung Status-Meeting 1
5	Status-Meeting 1	19.09.2020	Online	
6	Teamsitzung	19.09.2020	MS-Team	Nachbearbeitung Status-Meeting 1
7	Teamsitzung	01.10.2020	MS-Team	
8	Teamsitzung	09.10.2020	MS-Team	
9	Teamsitzung	16.10.2020	MS-Team	
10	Teamsitzung	23.10.2020	MS-Team	
11	Teamsitzung	06.11.2020	MS-Team	
12	Teamsitzung	12.11.2020	MS-Team	Vorbereitung Status Meeting 2
13	Status-Meeting 2	14.11.2020	Online	
14	Teamsitzung	14.11.2020	MS-Team	Nachbearbeitung Status-Meeting 2

Status Lieferobjekte inkl. Bewertung / allfällige Massnahmen

MS	Name	Lieferobjekte	Start	Deadline	Status	Bemerkung
1	Initialisierung		01.09.2020	10.09.2020		
		Projektziele		10.09.2020	planmäßig	
		Anforderungen (Grob)		10.09.2020	planmäßig	
		Meilensteinen planen		10.09.2020	planmäßig	
		1. Rollenverteilung		10.09.2020	planmäßig	
2	Vorstudie		11.09.2020	18.09.2020		<i>Bis am 26. September verlängert</i>
		Projektauftrag		18.09.2020	planmäßig	
		Planung (initiale)		18.09.2020	planmäßig	
		Projektstrukturplan		18.09.2020	planmäßig	
		Risikoanalyse		18.09.2020	planmäßig	

		Status-Bericht 1		18.09.2020	planmäßig	reviewed
		Präsentation		18.09.2020	planmäßig	reviewed
		2. Rollenverteilung		19.09.2020	planmäßig	
3	Konzeption		20.09.2020	18.10.2020		Neues Datum 27.09.2020
		Fachanforderungen			i. Bearbeitung/nicht planmäßig	Wird in der Konzeption noch weiterentwickelt
		Nichtfunktionale Anforderungen		06.11.2020	planmäßig	
		Funktionale Konzeption		06.11.2020	planmäßig	
		Technische Konzeption		06.11.2020	planmäßig	
		Testkonzept		06.11.2020	planmäßig	
4	Realisierung		27.10.2020	13.12.2020		Neues Datum: 06.11.2020
		Quellcode		13.12.2020	i. Bearbeitung	Es besteht ein Risiko, dass die Kannkriterien nicht 100% implementiert werden.
		Unit-Test		13.12.2020		
		Testprotokolle		13.12.2020		
		Java Dokumentation		13.12.2020		
		Installationsanleitungen (Booklet)		13.12.2020		
		Produkt Vorbereiten zur Abgabe (VM)		13.12.2020		
5	Projektabchluss		14.12.2020	20.12.2020		
		Benutzerhandbuch		20.12.2020		
		Projektdokumentation		16.12.2020		

Status Qualität inkl. Bewertung / allfällige Massnahmen

Name	Bewertung (Qualität)	Bemerkung	Massnahmen
Projektstrukturplan	5	Angenommen	
Risikoanalyse	5	Angenommen	
Pflichtenheft	5	Angenommen	
Fachanforderung		i. Bearbeitung	
Nichtfunktionale Anforderungen		Angenommen	
Funktionale Konzeption		Angenommen	
Technische Konzeption		Angenommen	
Testkonzept		gemacht	Wird angenommen
Statusbericht 2		angenommen	
Präsentation SB 2		angenommen	

Skala: 6: hervorragend / 5: gut / 4: genügend / 3: ungenügend / 2: sehr ungenügend / 1: abgelehnt

Nächste Schritte / Änderungsanträge

Wir treten jetzt in die Realisierungsphase ein. Die Zeit drängt, da wir in der Entwurfsphase aufgrund von Änderungen an einigen unserer Designs (wiederum aus Sicherheitsgründen) etwas Zeit verloren haben. Darüber hinaus hat uns die Realisierung von PoC (PoC Gui, PoC Password generator/verschlüsselung, PoC Database) sehr in Anspruch genommen, aber wir sind zuversichtlich für die Zukunft, denn wir werden einige Teile des entwickelten Codes wiederverwenden können, was uns etwas Zeit sparen sollte. Nichtsdestotrotz ist der Weg mit Fallen gepflastert, und wir gehen davon aus, dass wir für diese Phase viele Stunden einplanen werden, da keiner von uns im wirklichen Leben ein Entwickler ist.

Aktualisierte Rollenorganisation

Abbildung 1: Rollenorganisation

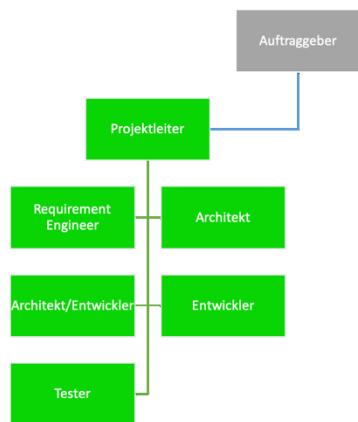


Abbildung 2: Team C3rBytes

Aktuelle Rollenverteilung:

Vorname	Name	aktuelle Rolle	bis
Jérémie	Equey	PL	19.09.2020
Olaf	Schmidt	RE	19.09.2020
Mersid	Hazbiu	RE	19.09.2020

Ab 20.09.2020

Vorname	Name	aktuelle Rolle	bis
Jérémie	Equey	A/E	06.11.2020
Olaf	Schmidt	PL	06.11.2020
Mersid	Hazbiu	A/E	06.11.2020

Ab 06.11.2020

Vorname	Name	aktuelle Rolle	bis
Jérémie	Equey	A/E	14.12.2020
Olaf	Schmidt	A/E	14.12.2020
Mersid	Hazbiu	PL	14.12.2020

Abk. Beschreibung

PL Projektleiter

- E** Entwickler
A Architekt
A/E Architekt/Entwickler
RE Requirement Engineer
TE Tester

Diskussion

Fragen:

- DB in 3NF?
-

**6.1.4. Statusbericht 2 – Reflexion****PA_5, Projektarbeit, INF-P-AT005, BE1****Reflexion zum Statusbericht 2 Gruppe 4:****Jérémie Equey, Olaf Schmidt, Mersid Hazbiu**

14.11.2020

Anwesende:

- Dozenten: Claudio Zesiger, Markus Stern
- Studenten: Olaf Schmidt, Mersid Hazbiu, Jérémie Equey

An unserem 2. Statusmeeting am 14.11.2020 (09:55-10:40) werden die folgenden Massnahmen zur Anwendung gelangen.

Projektmanagement

Nr.	Inputs von Dozenten	Massnahmen
1	Lösungsvarianten sind nicht aufgeführt, es fehlt eine Nutzwertanalyse wie die Projektziele umgesetzt werden können.	<p>Mit dem Lösungsproblemzyklus Wahl einer Variante (auf 3 Varianten) um eine Password Manager-Software zu entwickeln (Nutzwertsanalyse: Verteilte Software vs Standalone Software vs Verteilte-Standalone Software).</p> <p>Es muss eine sauberer Lösungsansatz implementiert werden. (Dozent Stern)</p>
2	Terminplanung fehlt	Terminplanung mit den Arbeitspaketen sowie den PSP in der Dokumentation sauber darstellen.
3	Projektabchlussphase: 1 Woche um diese Phase zu implementieren ist eng.	Wir versuchen die Dokumentation laufend zu ergänzen soweit möglich.
4	Lieferobjekte in der Arbeitsmappe: sind diese in der Dokumentation der Gruppe implementiert/beschrieben/vorhanden?	<p>Es wird geprüft, ob alle Lieferobjekte gemäss Arbeitsmappe erstellt wurden. Falls nein, wir werden die fehlenden Elemente noch bis Abgabe des Projekts implementieren/beschreiben.</p> <p>z.B. Arbeitsmappe 3.2.6: Beschreibung der Benutzung des GUI (hier wird das MVC Pattern unserer Applikation erläutert)</p>

Software Engineering

Nr.	Inputs von Dozenten	Massnahmen
1	Das Passwort in der Zwischenablage zu haben, kann ein Sicherheitsrisiko sein, da Programme dies lesen können	Passwort aus Zwischenablage entfernen nach einiger Zeit
2	Datenbank in die 3NF zu bringen?	<ul style="list-style-type: none"> • Datenbank nicht normalisieren, um Transaktionszeiten zu reduzieren • Kurze Transaktionszeiten erhöhen die Sicherheit, da die Zeit zwischen Entschlüsselung der Datenbank und Verschlüsselung der Datenbank kürzer wird.

3	Use Case: prüfen ob die Akteure Datenbank und System wirklich nötig sind	Sie sind nicht nötig. Die Use Cases werden entsprechend angepasst.
4	Singleton Pattern für die Verbindung mit der Datenbank	Dozent Zesiger: wichtiger ist, dass die Verbindung jedes Mal sauber geschlossen wird. Singleton ist nicht unbedingt nötig in diesem Projekt.
	Implementation von Multithreading	Es würde eine Komplexität einfügen, die unser Projekt nur unnötig aufblasen würde. Die Komplexität würde stark steigen und der Performanzgewinn würde, wenn überhaupt, nur gering sein. Im "worst case" würde die Performanz sogar darunter leiden.

Wir danken den Dozenten für Ihre Zeit und ihre Aufmerksamkeit sowie für die nützlichen Inputs.