



Bachelorarbeit (Informatikingenieurwesen)

Individuell Konfigurierbarer Authentifizierungsservice für Votings und Wettbewerbe

Autor

Christian Bachmann

Betreuung

Jaime Oberle

Auftraggeber

inaffect AG

Datum

23.12.2015

Inhaltsverzeichnis

I	Präambel	3
1	Einführung	4
1.1	Motivation	4
1.2	Aufgabenstellung	5
1.2.1	Ausgangslage	5
1.2.2	Ziel der Arbeit	5
1.2.3	Aufgabenstellung	5
1.2.4	Erwartete Resultate	6
1.3	Rahmenbedingungen Bachelorarbeit	7
2	Projektmanagement	8
2.1	Vorgehensweise	8
2.2	Grobe Projektplanung	9
2.3	Aufwand	10
2.4	Meilensteine	10
2.5	Termine	10
2.6	Infrastruktur	10
2.6.1	Quellcode-Verwaltung	10
2.6.2	Projekt Planung	10
3	Recherche	11
3.1	Fachbegriffe	11
3.2	Erläuterung der Grundlagen	11
3.2.1	Authentifizierung	11
3.2.2	Autorisierung	11
3.2.3	Captcha	12
3.2.4	OAuth	13
3.3	Ähnliche Produkte auf dem Markt	14
3.3.1	OAuth-Provider	14
3.3.2	playbuzz.com	16
3.3.3	WebSMS.com Zwei-Faktor-Authentifizierung	17
3.4	Grundlegende Sicherheitsprinzipien	18
3.4.1	KISS	18
3.4.2	Default-is-deny	18
3.4.3	Open Design	18
3.4.4	Zusammenfassung der Sicherheitsprinzipien	19
4	Anforderungen	20
4.1	Vorgehensweise	20
4.2	Use-Cases	20

5	Architektur	23
5.1	Integration der Schnittstelle	23
5.1.1	Bestehende Systeme für Votings, Wettbewerbe und Quizes	23
5.2	Hauptziel der Studie	24
6	ProofOfConcept	25
7	Fazit	26
A	Glossar	27
B	Verzeichnisse	28
B.1	Abbildungsverzeichnis	28
B.2	Quellenverzeichnis	29
B.3	Tabellenverzeichnis	30

Teil I

Einleitung und Abgrenzung

1 Einführung

1.1 Motivation

Die Digitalisierung fordert die Schweizer Wirtschaft heraus. Ob Banken, Pharma, Detailhandel oder Medienhäuser – es gibt keine Branche, die nicht vor fundamentalen Veränderungen steht. (Millischer 2015) Da verwundert es nicht, dass Wettbewerbe oder Kreuzworträtsel nicht nur auf den letzten Seiten des Klatschhaften oder Zeitungen abgedruckt werden sondern vermehrt online publiziert und durchgeführt werden. Dass bei meinungsbildenden Umfragen oder Abstimmungen weniger auf Telefon zurückgegriffen wird sondern diese immer mehr im Internet durchgeführt werden.

In der Schweiz konnten die grossen Medienhäuser ihre Zugriffszahlen auch 2015 steigern und ihre Toprangierungen beibehalten. ("NET-Metrix-Audit" 2015) Um ihren Werbegewinn und Resonanz zu bewahren oder sogar auszubauen sind Medien angewiesen, dass ihre Stories/Content auf den Social Media Kanälen verlinkt und so viral verbreitet werden. Neben altbekannten plakativen Titeln und interessanten Bildern beleben die Medienhäuser immer mehr ihren Content mit so genannten Playfull Content integriert durch Social Module. Dabei handelt sich um Abstimmungen, Wettbewerbe und Umfragen oder anderen Interaktivitäten im Zusammenhang mit dem verfassten Inhalt. Diese Social-Module werden gerne verlinkt und fördern so die Verbreitung des Contents und dadurch einen Anstieg der Besucherzahlen.

Bei den meisten angebotenen Umfragen, Abstimmungen und Wettbewerbe ist es relativ simpel (gewisses Know-How vorausgesetzt) mehrfach teilzunehmen oder gefälschte Daten zu übermitteln. Dies ist auf zu einfach realisierte Programmierungen zurückzuführen, was der Glaubwürdigkeit solcher Angebote schadet. Social-Module wie Umfragen, Abstimmungen oder Wettbewerbe bedürfen somit einer Authentifizierung, um Betrug oder falschen Stimmabgaben vorzubeugen. Die Eigenentwicklung der gewünschten Authentifizierung für ein Modul übersteigt meist die kleinen Budgets für diese Angebote.

Die Glaubwürdigkeit der Umfragen, Abstimmungen und Wettbewerbe ist durch die aktuelle Situation gefährdet und soll wiederhergestellt werden. Deshalb soll diese Bachelorarbeit die Möglichkeit eines Authentifizierungsservice erörtern. Mit dieser sollen Programmierer über eine visuelle Oberfläche die Bedürfnisse eines Angebots konfigurieren und in ihren jeweiligen Modulen einbinden können.

1.2 Aufgabenstellung

1.2.1 Ausgangslage

Bei populären Medienhäusern und grösseren Unternehmen werden häufig Umfragen, Abstimmungen oder Gewinnspiele im Internet durchgeführt. Bei den meisten angebotenen Programmen ist es relativ simpel (gewisses Know-How vorausgesetzt) mehrfach teilzunehmen oder gefälschte Daten zu übermitteln. Dies ist auf zu einfach realisierte Programmierungen zurückzuführen, was der Glaubwürdigkeit solcher Angebote schadet. Social-Media Module wie Umfragen, Abstimmungen oder Wettbewerbe bedürfen somit einer Authentifizierung, um Betrug oder falschen Stimmabgaben vorzubeugen. Die Eigenentwicklung der gewünschten Authentifizierung für ein Modul übersteigt meist die kleinen Budgets für diese Angebote. Die Firma inaffect AG erstellt Individuallösungen und Webapplikationen im Bereich neuer Medien. Sie ist auf der Suche nach einem Authentifizierungsservice, welche ihre Programmierer mit einer visuellen Oberfläche den Bedürfnissen eines Angebots konfigurieren und in ihr jeweiliges Modul einbinden können.

1.2.2 Ziel der Arbeit

Es soll ein Konzept für eine Authentifizierungsschnittstelle erstellt werden. Dieser Service wird über mehrere Sicherheitsstufen verfügen, die sich in der Menge und Art der zu übermittelnden User-Informationen unterscheiden. Diese Stufen sollen für den Programmierer eines Angebots über eine grafische Oberfläche individuell konfigurierbar sein. Das Konzept soll in Form eines Prototypen umgesetzt werden. Weiter soll mit mehreren Usern eine Studie zur Akzeptanz und Geschwindigkeit der verschiedenen Sicherheitsstufen durchgeführt werden. Die Ergebnisse der Studie werden im Prototyp integriert sein und sollen den Programmierer bei der Auswahl der Sicherheitsstufe unterstützen.

1.2.3 Aufgabenstellung

Im Rahmen der Bachelorarbeit werden vom Studenten folgende Aufgaben durchgeführt:

Recherche

- Research und Marktanalyse bestehender Produkte
- Arten und Methoden der Sicherheits- und Identitätsüberprüfung
- Durchführung einer Anforderungsanalyse für eine Authentifizierungsschnittstelle

Konzept

- Evaluation von geeigneten Authentifizierungsmethoden für verschiedene Stufen
- Spezifikation einer Prototypenapplikation für die Authentifizierungsschnittstelle
- Spezifikation einer Prototypenapplikation für das Verwalten der Authentifizierungsschnittstelle
- Erstellen einer Software-Architektur für die Authentifizierungsschnittstelle und dessen Verwaltung
- Ausarbeiten einer Studie über Akzeptanz und Geschwindigkeit von Authentifizierungsmethoden

Studie

- Durchführen der ausgearbeiteten Studie
- Auswertung der Studie

Proof of Concept

- Entwicklung eines Prototypen der Authentifizierungsschnittstelle und der Verwaltung, basierend auf den erarbeiteten Spezifikationen und Architektur
- Integration der Studienresultate im Prototypen

Fazit

1.2.4 Erwartete Resultate

Im Rahmen dieser Bachelorarbeit werden vom Studenten folgende Resultate erwartet:

Recherche

- Dokumentation des Research und Marktanalyse bestehender Produkte
- Dokumentation der Arten und Methoden der Sicherheits- und Identitätsüberprüfung

Analyse

- Dokumentierte Anforderungsanalyse für eine Authentifizierungsschnittstelle

Konzept

- Dokumentation der Evaluation von geeigneten Authentifizierungsmethoden für verschiedene Stufen
- Dokumentierte Spezifikation einer Prototypenapplikation für die Authentifizierungsschnittstelle
- Dokumentierte Spezifikation einer Prototypenapplikation für das Verwalten der Authentifizierungsschnittstelle
- Dokumentation der Software-Architektur für die Authentifizierungsschnittstelle und dessen Verwaltung
- Dokumentation des Ausarbeitens einer Studie über Akzeptanz und Geschwindigkeit von Authentifizierungsmethoden

Studie

- Dokumentation der Studien-Durchführung
- Dokumentation der Auswertung der Studie

Proof of Concept

- Dokumentierte Entwicklung eines Prototypen der Authentifizierungsschnittstelle und der Verwaltung, basierend auf den erarbeiteten Spezifikationen und Architektur
- Dokumentierte Integration der Studienresultate im Prototypen
- Dokumentiertes Fazit

1.3 Rahmenbedingungen Bachelorarbeit

Die vorliegende Bachelorarbeit umfasst gemäss Regelment unter anderem folgende Punkte:

- Eine Bachelorarbeit besteht aus einer konzeptionellen Arbeit und deren Umsetzung. Der Schwerpunkt liegt auf dem konzeptionellen Teil, in dem die theoretischen und methodischen Grundlagen einer Entwicklung oder eines Konzeptes ausgearbeitet und dargelegt werden. Im Umsetzungsteil erfolgt anschliessend die Beschreibung der Implementierung bzw. der Anwendung. Die Umsetzung besteht zumindest aus einem „Proof of Concept“, um die prinzipielle Realisierbarkeit darzulegen. Die Bachelorarbeit ist als praxisnahes Projekt durchzuführen.
- Der Aufwand für die Fertigstellung einer Bachelorarbeit beträgt insgesamt mindestens 360 Stunden.
- Die Bachelorarbeit hat die Form eines technischen Berichtes.¹

¹(Stern 2012)

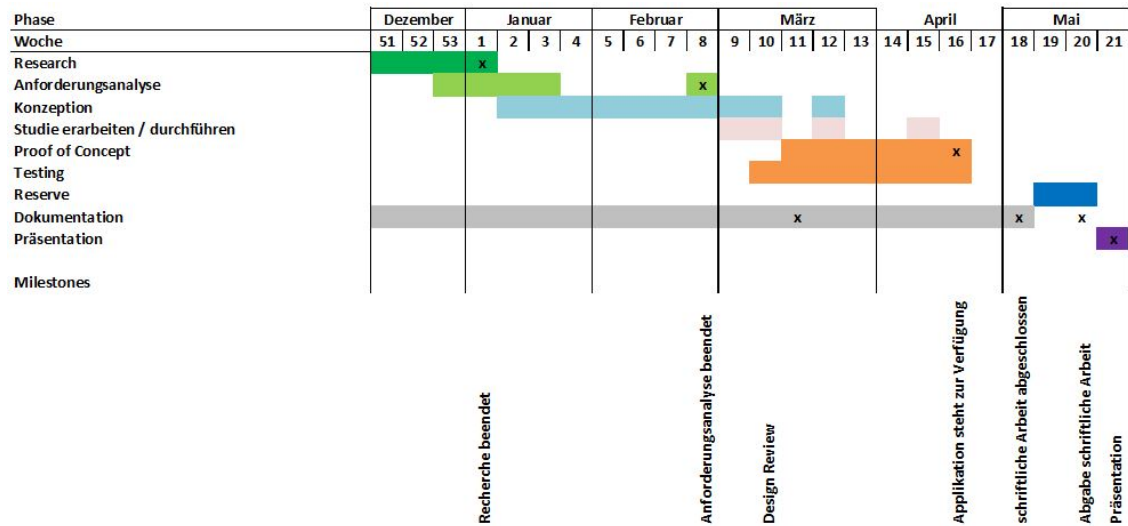
2 Projektmanagement

In diesem Kapitel wird die Planung der Bachelorarbeit ausgeführt. Weiter wird die verwendete Infrastruktur erläutert.

2.1 Vorgehensweise

2.2 Grobe Projektplanung

Der grobe Projektplan illustriert die Strukturierung der Bachelorarbeit über die knapp sechs Monate lange Projektzeit. Der Projektplan liefert einen generellen Überblick über den zeitlichen Ablauf der Bachelorarbeit und legt die Milestones fest.



2.3 Aufwand

Im Unterkapitel Rahmenbedingungen Bachelorarbeit wurde bereits ausgeführt, dass eine Bachelorarbeit laut Regelement mindestens 360 Stunden betragen soll. Diese Rahmenbedingung wurde bei Aufgabenstellung und Aufwandschätzung der Bachelorarbeit berücksichtigt.

Tabelle 2.1: Soll/Ist Analyse

Arbeitsschritt	Soll	Ist
Initialisierung	10	
Recherche	45	
Analyse	20	
Konzeption	80	
Prototyp	60	
Dokumentation	140	
Abgabe	20	
Total	375	xx

2.4 Meilensteine

2.5 Termine

2.6 Infrastruktur

2.6.1 Quellcode-Verwaltung

2.6.2 Projekt Planung

3 Recherche

3.1 Fachbegriffe

Eine ausführliche Erklärung der Fachbegriffe befindet sich im Anhang unter dem Kapitel "Glossar".

3.2 Erläuterung der Grundlagen

In diesem Kapitel werden Funktionsweisen und Grundlage ausgeführt, die als für die Bearbeitung dieser Bachelorthesis herangezogen wurden.

3.2.1 Authentifizierung

Authentifizierung - beglaubigen, die Echtheit von etwas bezeugen verfolgt das Ziel (*Duden* 2014)

Eine Person oder Objekt eindeutig zu **authentifizieren** bedeute zu ermitteln ob die oder derjenige auch der ist als welcher er sich ausgibt. (Rouse 2015) Dies unterstreicht auch die Ableitung des Wortes vom Englischen Verb *authenticate*, was auf Deutsch sich als *echt erweisen, sich verbürgen, glaubwürdig sein* bedeutet. Das bekannteste Verfahren der Authentifizierung ist die Eingabe von Benutzernamen und Passwort. Weiter ist die PIN-Eingabe bei Bankautomaten oder Mobiletelefon häufig verbreitet. Die Möglichkeiten der Authentifizierung nahe zu grenzenlos. ("Http://authentifizierung.org" 2015)

3.2.2 Autorisierung

Autorisierung - Befugnis, Berechtigung, Erlaubnis, Genehmigung (*Duden* 2014)

Wenn die Authentifizierung erfolgreich war erteilt das System die Autorisierung. Dabei wird der Person oder Objekt erlaubt bestimmte Aktionen/Zugriffe durchzuführen. Meist verfügen unterschiedliche Benutzer eines Systems über verschiedene Zugriffsrechte. Die korrekte Zuweisung der individuellen Rechte ist ebenfalls Bestandteil der Autorisierung.

Der Begriff Authentifizierung wird vielfach mit dem Begriff Autorisierung verwechselt. Die Authentifizierung wird vom Benutzer initiiert. Sie dient dem Nachweis, zur Ausübung bestimmter Rechte befugt zu sein. Die anschließende Autorisierung erfolgt automatisch durch das System selbst. Im Zuge der Autorisierung werden dem Benutzer seine Zugriffsrechte zugewiesen. ("Http://authentifizierung.org" 2015)

3.2.3 Captcha

Captcha - Test, mit dem festgestellt werden kann, ob sich ein Mensch oder ein Computer eines Programms bedient (*Duden* 2014)

Im Jahre 2000 wurde das Captcha an der Carnegie Mellon University erfunden. Captcha steht für **C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part. Luis von Ahn, Professor der Entwickler-Gruppe, erklärte die Dringlichkeit von Captcha damals so: "Anybody can write a program to sign up for millions of accounts, and the idea was to prevent that". (Burling 2012)

Captcha Zahlen

In 2014 wurden 200 Million Captchas pro Tag eingegeben. Dabei braucht ein User durchschnittlich 10 Sekunden das entspricht 500'000 Stunden.¹

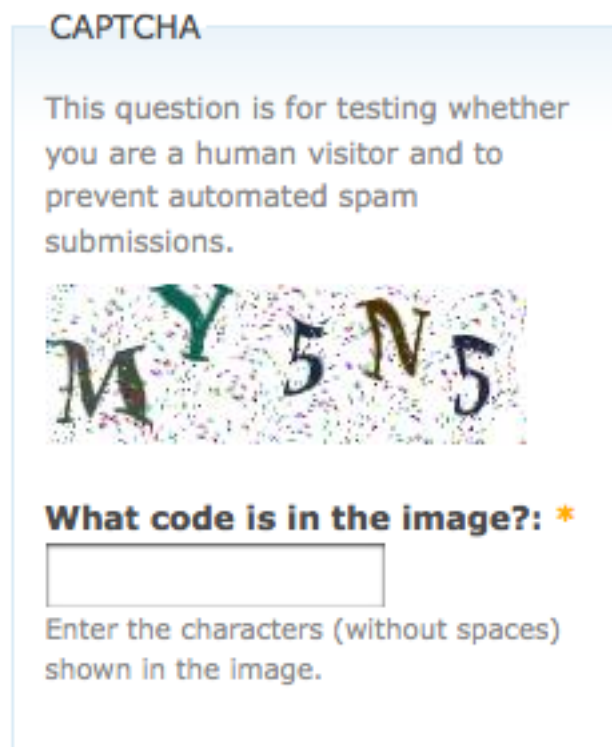


Abbildung 3.1: Beispiele von Captchas *Quelle:drupal.org*

¹Die statistischen Daten wurden von Google 2014 in ihrem Blog publiziert ("ReCAPTCHA Digitization Accuracy" 2014)

3.2.4 OAuth

OAuth ist ein Protokoll. Es erlaubt sichere API-Autorisierungen.

Das Bedürfnis nach OAuth

2006 implementierte Blaine Cook OpenID für Twitter und Magnolia erhielt ein Dashboard welches sich durch OpenID autorisieren lies. Deshalb suchten die Entwickler von Magnolia und Blaine Cook um eine Möglichkeit zu finden OpenID auch für die Verwendung von APIs zu gebrauchen. Sie diskutierten ihre Implementierungen und stellten fest dass es keinen offenen Standard für API-Access Delegation gab. So fingen sie an den Standard zu entwickeln. 2007 entstand daraus eine Google Group. Am 3. Oktober 2007 war dann der OAuth Core 1.0 bereits released worden.

Funktionalität von OAuth

Ein Programm/API (Consumer) stellt über das OAuth-Protokoll einem Endbenutzer(User) Zugriff (Autorisierung) auf seine Daten/Funktionalitäten zur Verfügung. Dieser Zugriff wird von einem anderen Programm (Service) gemanagt. Das Konzept ist nicht generell neu. OAuth ist ähnlich zu Google AuthSub, aol OpenAuth, Yahoo BBAuth, Upcoming api, Flickr api, Amazon Web Services api. OAuth studierte die existierenden Protokolle und standardisiert und kombinierte die existierende industriellen Protokolle. Der wichtigste Unterschied zu den existierenden Protokollen ist, dass OAuth sowohl offen ist und andererseits zu einem Standard geworden ist. Jeden Tag entstehen neue Webseiten welche neue Funktionalitäten und Services offeriert und dabei Funktionalitäten von anderen Webseiten braucht. OAuth stellt dem Programmierer einerseits eine standardisierte Implementierung zur Verfügung. Der Endbenutzer erhält dank dieses Protokolls die Möglichkeit Teile seiner Funktionalität/Daten bei einem anderen Anbieter zur Verfügung zu stellen. So kann der Endbenutzer bei der Facebook OAuth z.B. seine Posts zur Verfügung stellen nicht aber seine Freunde bekannt geben.

Dank der weiten Verbreitung gibt es nun in allen bekannten Programmiersprachen eine Implementierung sowohl von Client wie auch vom Server. Weitere Infos dazu unter oauth.net¹

3.3 Ähnliche Produkte auf dem Markt

Dieses Unterkapitel erläutert existierenden Produkte auf dem Markt.

3.3.1 OAuth-Provider

Die grössten OAuth-Provider wie Google, Facebook und Twitter erzielen eine weiter Verbreitung weltweit:

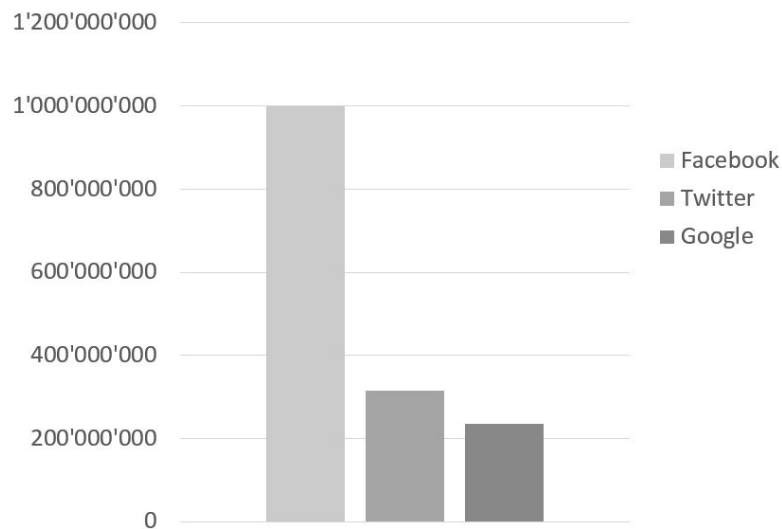


Abbildung 3.2: Aktive Nutzer Weltweit²

Ganze 78% (Interactive 2015) der Schweizer Bevölkerung nutzten SocialMedia und besitzen dadurch einen OAuth-Account:

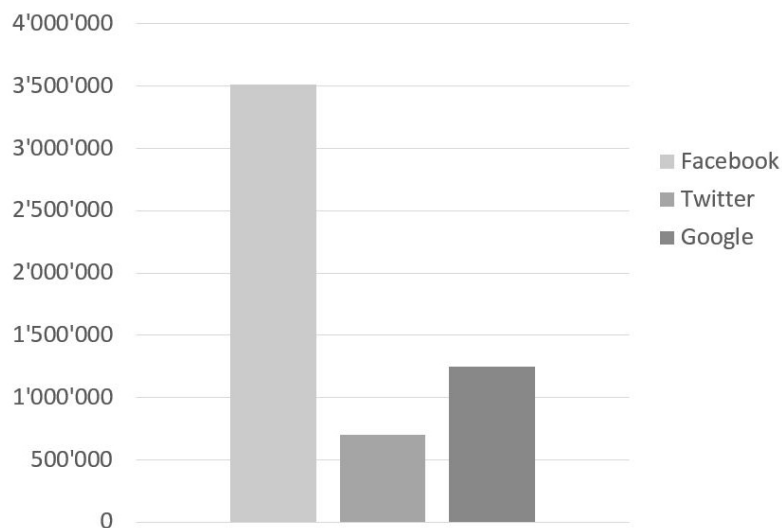


Abbildung 3.3: Anzahl Schweizer Nutzer³

²Das Statistik wurde basierend auf den Daten von socialmedia-institute ("SMI (SocialMedia Institute)" 2015) erstellt. Facebook und Twitter Daten sind am 5. November 2015 und die Google Daten sind im 2014 erhoben worden.

Vorteile

78% der Schweizer Bevölkerung besitzt bereits einen OAuth Account. Das Protokoll ist ein etablierter Standard.

Nachteile

Mehrfachregistrierungen sind möglich. Je nach OAuth-Provider werden verschiedene Daten zur Verfügung gestellt. Pro OAuth Provider kann man sich registrieren, ein Abgleich der verschiedenen OAuth Provider wird vom OAuth-Protokoll nicht zur Verfügung gestellt. 22% der Bevölkerung müsste sich vor Nutzung noch registrieren. Die Implementierung ist trotz vielen Libraries nicht ohne tiefere Programmierkenntnisse möglich.

³Das Statistik wurde basierend auf den Daten von Goldbach Interactive (Interactive 2015) generiert. Die Daten sind im März 2015 erhoben.

3.3.2 playbuzz.com

Youtube von Google ist im Jahr 2015 mit Abstand die meist verbreiteste ("Statistik Plattform" 2015) Videopublishing-Plattform. Medienhäuser nutzen Youtube um einfach Ihren Artikel mit einem Video zu ergänzen. Neben der einfachen Integration profitieren die Medienhäuser von der zusätzlichen Verbreitung über youtube.com und der einfachen viralen Verbreitungsmöglichkeiten von youtube. PlayBuzz möchte das Youtube für Votings, Quiz und ähnlicher Embeded Content zu werden. Neben MTV, Focus, Time oder Bild verwendet seit Herbst 2015 auch ein grosses Medienhaus der Schweiz die Plattform. Tamedia erfasst neu auf 20minuten Votings und Umfragen mit PlayBuzz.

2012 wurde Playbuzz von Shaul Olmert (Sohn des Premie Minster von Israel Ehuad Olmert) und Tom Pachys ins Leben gerufen. Der offizielle Launch war im Dezember 2013. Im Juni 2014 wurde Playbuzz bereits das 1. Mal unter den Top 10 Facebook Shared Publishers aufgelistet. Im Juni 2014 konnte Playbuzz bereits 70 millionen unique views aufweisen. Im September 2014 kamen 7 von den 10 Top Shares auf Facebook laut forbes.com von Playbuzz. Playbuzz setzt nach eigenen Angaben auf Content wie Votes und Quizes welcher gerne Viral geteilt wird und ermöglicht Endnutzer und Redakteuren einfache Verwendung. ("Interview Mit Shaul Olmert" 2015) ("PlayBuzz" 2015)

Vorteile

Playbuzz ist kostenlos und lässt sich einfach integrieren. Durch Verwendung von Playbuzz kann die Verbreitung des eigenen Inhalts stark gesteigert werden. Die Verwaltungsoberfläche und Reports sind übersichtlich und einfach zu bedienen.

Nachteile

Der Verweis auf Playbuzz ist ersichtlich. Auch beim Posten auf den SocialMedia-Kanälen ist die Herkunft von Playbuzz offensichtlich. Die Möglichkeiten in Funktionalität und Design haben Grenzen. Individuelle Erweiterungen sind nicht einfach möglich.

3.3.3 WebSMS.com Zwei-Faktor-Authentifizierung

Seit 1. Juli 2004 müssen auch bei Prepaid-Karten in der Schweiz Personalien hinterlegt werden.⁴ Dadurch ist eine eindeutige Authentifizierung über Mobilenummer gewährleistet. Die Mobilefunkanbieter schenken die Anzahl Telefonnummern pro Person ein:

Attribut	Anzahl SIM Cards
Swisscom	5 Sim Cards pro Person
Beschreibung	Fehler/Problembeschreibung des Tickets.
Anmerkungen	Alle Antworten von Technikern und Kunden. Eine Antwort des Technikers kann als FAQ-Eintrag markiert werden.
Arbeitszeit	Alle erfassten und aufgewendeten Stunden für den Support-Fall.
tArbeitszeit	Das Total der erfassten Arbeitszeit.
pNotes	Persönliche Bemerkungen zum Support-Fall. Nur der Autor einer Notiz, kann diese bearbeiten oder lesen.

2012 wurde Playbuzz von Shaul Olmert (Sohn des Premie Minister von Israel Ehud Olmert) und Tom Pachys ins Leben gerufen. Der offizielle Launch war im Dezember 2013. Im Juni 2014 wurde Playbuzz bereits das 1. Mal unter den Top 10 Facebook Shared Publishers aufgelistet. Im Juni 2014 konnte Playbuzz bereits 70 millionen unique views aufweisen. Im September 2014 kamen 7 von den 10 Top Shares auf Facebook laut forbes.com von Playbuzz. Playbuzz setzt nach eigenen Angaben auf Content wie Votes und Quizes welcher gerne Viral geteilt wird und ermöglicht Endnutzer und Redakteuren einfache Verwendung. ("Interview Mit Shaul Olmert" 2015) ("PlayBuzz" 2015)

Vorteile

Playbuzz ist kostenlos und lässt sich einfach integrieren. Durch Verwendung von Playbuzz kann die Verbreitung des eigenen Inhalts stark gesteigert werden. Die Verwaltungsoberfläche und Reports sind übersichtlich und einfach zu bedienen.

Nachteile

Der Verweis auf Playbuzz ist ersichtlich. Auch beim Posten auf den SocialMedia-Kanälen ist die Herkunft von Playbuzz offensichtlich. Die Möglichkeiten in Funktionalität und Design haben Grenzen. Individuelle Erweiterungen sind nicht einfach möglich.

⁴Meldung des UVEKS über Gesetzesänderung: ("NET-Metrix-Audit" 2004)

3.4 Grundlegende Sicherheitsprinzipien

In diesem Unterkapitel werden die Grundlagen der Sicherheitsprinzipien vermittelt auf denen danach eine Authentifizierungssoftware aufgebaut werden kann.

3.4.1 KISS

Keep It Stupid and Simple

Ein verbreitetes Problem unter Software Engineers und Programmier heute ist, dass sie dazu tendieren Probleme zu kompliziert und verschachtelt zu lösen. 8-9 von 10 Entwicklern machen den Fehler, Probleme zu wenig auseinander zu brechen und alles in einem grossen Programm zu lösen. Anstatt es in kleinen Paketen verständlich zu programmieren. (Hanik 2015)

Software Entwickler profitieren von Kiss:

- You will be able to solve more problems, faster.
- You will be able to produce code to solve complex problems in fewer lines of code
- You will be able to produce higher quality code
- You will be able to build larger systems, easier to maintain
- You're code base will be more flexible, easier to extend, modify or refactor when new requirements arrive
- You will be able to achieve more than you ever imagined
- You will be able to work in large development groups and large projects since all the code is stupid simple

KISS fördert die Sicherheit

Die Begründung warum KISS die Sicherheit fördert liefert Saltzer und Schroeder: Ungewollte Zugriffspfade können nur durch zeilenweise Codeinspektion entdeckt werden und die wiederum setzt voraus, dass Designs einfach und klein sein sind. Designs müssen so beschaffen sein, dass sie abgeschlossene Bereiche enthalten, über die konkrete und sichere Aussagen über Zugriffsmöglichkeiten und Effekte getroffen werden können. (Kriha and Schmitz 2009, 93)

3.4.2 Default-is-deny

Ob eine Person oder Programm Zugriff auf Daten/Funktionen haben, sollte nicht durch Verbote sondern durch explizite Erlaubnis geregelt werden. Dies bedeutet solange keine explizite Erlaubnis gesetzt ist, kann das Programm oder die Person nicht auf die Daten oder Funktionen zugreifen. You *deny* it. So simpel und logisch diese Idee klingt, umso verwunderlich ist wie viele Organisationen und Entwicklungsfirma nicht dieses vorgehen verwenden. z.B. Filesysteme setzen auf Verbote anstatt auf explizite Erlaubnisse. (Rothman 2015) , (Kriha and Schmitz 2009, 94)

3.4.3 Open Design

Abgeleitet von der Kryptografie: Nicht das Design der Software sollte die Sicherheit sein, sondern der verwendete Schlüssel. Dieses Konzept gilt es in der Softwareentwicklung und

Systemtechnik nur bedingt einzuhalten. Die Software soll nach dem Ansatz entworfen werden. Mindestens intern soll das Software-Design durch einen Design-Review Prozess analysiert werden. In manchen Fällen macht es jedoch das Softwaredesign geheimzuhalten um einem Angreifer nicht zu viele Informationen zur Verfügung zu stellen. (Kriha and Schmitz 2009, 95)

3.4.4 Zusammenfassung der Sicherheitsprinzipien

Die wichtigsten Sicherheitsprinzipien zusammengefasst:

- Software muss aus kleinen, isolierten Einheiten aufgebaut werden, deren externe Beziehungen am Interface deutlich werden. Damit wird sowohl praktische Schadensreduzierung durch Isolation als auch eine schnelle und einfache Sicherheitsanalyse möglich.
- Zugriffsentscheidungen dürfen nur auf der Basis expliziter, minimaler und keinesfalls durch immer und global verfügbare Permissions fallen.
- Das Softwaredesign von Applikationen sollte wenn möglich öffentlich sein. Zumindest sollte ein interner Review-Prozess stattfinden, in dessen Verlauf eine Sicherheitsanalyse durch nicht an der Entwicklung Beteiligte erstellt wird.

4 Anforderungen

Dieses Kapitel beschreibt das Durchführen einer Anforderungsanalyse festgehalten. Anhand der Anforderungsanalyse sollen die Anforderungen für die entwickelnden Software ermittelt werden. Die Anforderungen bilden die Basis für die Architektur, das Softwaredesign, die Implementierung und die Testfälle. Ihnen ist dem entsprechend ein sehr grosser Stellenwert zuzuschreiben.

4.1 Vorgehensweise

Die Schlüsselwörter „muss“, „muss nicht“, „erforderlich“, „empfohlen“, „sollte“, „sollte nicht“, „kann“ und „optional“ in allen folgenden Abschnitten sind gemäss RFC 2119 zu interpretieren. (Bradner 1997)

4.2 Use-Cases

Im Nachfolgenden werden alle UseCases aufgelistet die im Rahmen dieser Thesis gefunden wurden.

UC-01 Registration

UseCase	
Ziel	Ein Developer ist am Authentifizierungsschnittstellen-Service registrieren
Beschreibung	Ein Developer muss sich am Authentifizierungsschnittstellen-Service registrieren können
Akteure	Developer
Vorbedingung	Keine
Ergebnis	Registrierter Developer
Hauptszenario	Der Developer füllt ein Registrierungsformular aus und bestätigt seine E-Mail Adresse
Alternativszenario	-

UC-01 Login

UseCase	
Ziel	Ein Developer kann sich beim Authentifizierungsschnittstellen-Service

UseCase	
Beschreibung	Ein Developer muss sich am Authentifizierungsschnittstellen-Service authentifizieren können
Akteure	Developer
Vorbedingung	Der Developer ist registriert.
Ergebnis	Authentifizierter und eigeloggter Developer
Hauptszenario	Der Developer loggt sich mit E-Mail und Passwort am Authentifizierungsschnittstellen-Service ein.
Alternativszenario	Der Developer sendet sich das verpasste Passwort per E-Mail zu. Erstellt über den im erhaltenden E-Mail enthaltenen Link ein neues Passwort und loggt sich mit E-mail und dem neuen Passwort am Authentifizierungsschnittstellen-Service ein.

UC-01 Konfigurieren einer neuen Social-Media Modul Authentifizierungsvorgang

UseCase	
Ziel	Es ist eine neuer Authentifizierungsvorgang für ein neues Social Media-Modul konfiguriert
Beschreibung	Der Developer kann eine neue Aktivität eröffnen
Akteure	Developer
Vorbedingung	Der Developer hat sich am System angemeldet
Ergebnis	Neuer Authentifizierungsvorgang
Hauptszenario	Der Developer eröffnet einen neuen Authentifizierungsvorgang. Er benennt ihn sinnvoll. Die zu verwendende(n) Authentifizierungskomponenten werden ausgewählt. Bei der Konfiguration unterstützen die Resultate die Studie den Developer für die optimale Konfiguration. Am Ende der Konfiguration werden die Akzeptanzkriterien für eine erfolgreiche Authentifizierung festgelegt.
Alternativszenario	Ein bestehender Authentifizierungsvorgang wird dupliziert

UC-01 Einbinden in vorhandenes System

UseCase	
Ziel	Die Authentifizierungsschnittstelle kann in ein (bestehendes) System eingebunden werden
Beschreibung	Der Developer kann die Authentifizierungsschnittstelle in seinem System integrieren
Akteure	Developer
Vorbedingung	Der Developer hat sich am System angemeldet. Der Developer hat einen neuen Authentifizierungsvorgang konfiguriert

UseCase	
Ergebnis	Der Developer hat eine Möglichkeit die Authentifizierungsschnittstelle mit seinem konfigurierten Authentifizierungsvorgangs in seiner Software einzubinden
Hauptszenario	Der Developer öffnet die Einbindeseite. Es werden ihm alle Schritte zur Erfolgreichen Einbindung aufgelistet. Der Code liegt individualisiert vor. Der Developer kopiert den Code in sein Programm
Alternativszenario	-

UC-01 Report eines Authentifizierungsvorgangs

UseCase	
Ziel	Die Verwendung des Authentifizierungsvorgangs ist übersichtlich dargestellt
Beschreibung	Um den Verwendung des Authentifizierungsvorgangs auszuwerten soll ein Report erstellt werden
Akteure	Developer
Vorbedingung	Der Developer hat sich am System angemeldet. Der Developer hat ein neues Authentifizierungsvorgang konfiguriert. (Der Authentifizierungsvorgang ist eingebunden und verwendet worden)
Ergebnis	Report eines Authentifizierungsvorgangs
Hauptszenario	Nach Beenden eines Quizes, Votings, Wettbewerbs logt sich der Developer im System ein und generiert einen automatisierten Report um die Verwendung des Authentifizierungsvorgangs auszuwerten.
Alternativszenario	Um den Zwischenstand deines Quizes, Votings, Wettbewerbs auszuwerten logt sich der Developer im System ein und generiert einen automatisierten Report um die Verwendung des Authentifizierungsvorgangs auszuwerten.

5 Architektur

5.1 Integration der Schnittstelle

Wie in der Anforderungsanalyse und Aufgabenstellung geschrieben, soll die Schnittstelle möglichst einfach in Bestehende Systeme integriert werden können. Bevor wir untersuchen wie wir die Integration umsetzen können, bedarf es die wichtigsten bestehenden Systeme zu kennen.

5.1.1 Bestehende Systeme für Votings, Wettbewerbe und Quizes

Das bestehende Social-Media Modul wird als Teil einer Webseite in einer Webapplikation geführt. Die Webapplikation zum Verwalten von Inhalten kurz CMS werden von Statista.com mehrmals im Jahr ausgewertet.¹ Folgend ist die letzte Erhebung abgebildet:

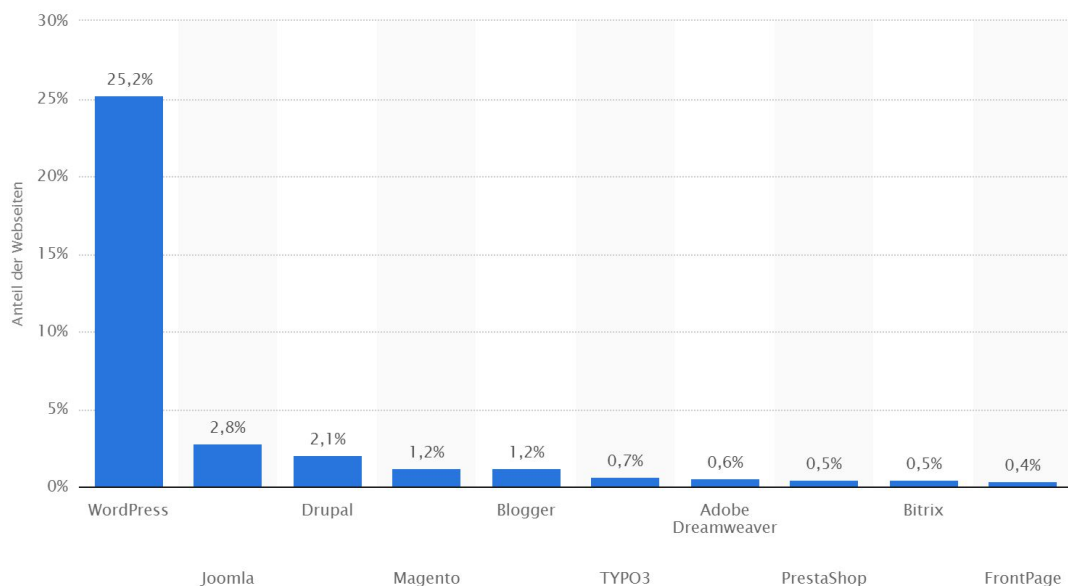


Abbildung 5.1: Nutzungsanteil CMS weltweit *Quelle:de.statista.com*

Die Zahlen wurden mit Werten von W3techs.com verglichen². Die Unterschiede sind für unsere Verwendung minimal und liegen im 10tels Prozentbereich. Beim Betrachten der Statistik fällt auf das Wordpress mit 25,2 mit Abstand am meisten genutzt wird. Alle dynamischen Webseiten unter den Top 10 basieren auf Systemen in PHP[^phpinfotag]. Adobe Dreamviewer und FrontPage sind keine Systeme welche auf dem Server betrieben werden. Sie sind Editoren welche auf dem jeweiligen Computer ausgeführt werden und danach mehrheitlich HTML, CSS

¹CMS Nutzungsstatistik von statista.com ("Top 10 CMS November 2015" 2015)

²CMS Nutzungsstatistik von w3techs.com ("Usage of Content Management Systems for Websites" 2015)

und Javascript Code an den Server ausliefern. Funktionalitäten werden mit den beiden Editoren manuell geschrieben.

Wodpress PlugIn

Erweiterungen im Wordpress nennen sich Plugins. Die Plugins können direkt über das CMS-Backend eingespielt werden. Alternativ können Sie natürlich manuell installiert werden. Zum Beispiel in dem man ein Plugin selber programmiert oder beim Hersteller oder über das Plugin-Verzeichnis von Wordpress[^plugin-verzeichnis] downloaded. Wordpress sammelt zugleich die aktiven Installationen der Plugins (sofern man als Entwickler den Informationsaustausch nicht unterbindet). Die Gesamtzahl wird im CMS-Backend Wordpress und auf Ihrer Plugin-Verzeichnis Webseite[^plugin-verzeichnis] veröffentlicht. Dank dieser Kennzahl kann nun die meist verbreiteten Plugins herausgefunden werden.

Möglichkeit ein Wordpress PlugIn zu erweitern

Wordpress basiert auf einem sogenannten Hook-System. Übersetzt man "Hook" vom Englischen ins Deutsch erhält Haken, Aufhänger oder Greifer. Ein Hook ist im Wordpress eine definierte Code-stelle bei der man seinen eigenen Code einhängen kann. Der PlugIn Entwickler definiert diese Hooks um anderen Plugins oder Funktionalitäten zu erlauben sein PlugIn zu erweitern. Auch der Core vom Wordpress enthält solche Hooks. Dadurch soll verhindert werden, dass PlugIn's oder der Core von Wordpress direkt umgeschrieben werden muss und dann nicht mehr einfach so unabhängig upgedatet werden kann. Um unsere Schnittstelle einzubinden, könnten wir auf den ersten Blick also solche Hooks verwenden. Dieser "Hook"/Haken hat lustigerweise auch einen Haken: Der PlugIn-Entwickler kann selbständig bestimmen ob und wo er solche Hooks einsetzen will und welche Möglichkeiten dann zur Verfügung stehen. Kommerzielle PlugIn's verfolgen vielfach den Weg möglichst verschlossen zu agieren um mögliche Erweiterungen monetär umzusetzen und so eine Abhängigkeit zu erzeugen. Im nächsten Kapitel ist analysiert ob und wie die grossen Social-Media Module von Wordpress Hooks einsetzen.

Wordpres Social-Media Hooks [^phpinfotag] Die Information wurde von den jeweiligen Hersteller- bzw. Communitywebseiten bezogen. [^plugin-verzeichnis] Das Pluginverzeichnis befindet sich unter <https://de.wordpress.org/plugins/> #Studie

5.2 Hauptziel der Studie

In den vergangenen Kapiteln wurden immer wieder verschiedene Authentifizierungsarten erwähnt und beschrieben. Diese verschiedenen Möglichkeiten gilt es mit einander zu vergleichen.

6 ProofOfConcept

7 Fazit

A Glossar

ORM ORM steht für object-relational mapping und ist eine Technik mit der Objekte einer Anwendung in einem relationalen Datenbanksystem abgelegt werden kann.

Github Github ist ein Cloud basierter SourceCode Verwaltungsdienst für Git. <https://github.com>

B Verzeichnisse

Neues Verzeichnisse

B.1 Abbildungsverzeichnis

3.1	Beispiele von Captchas <i>Quelle:drupal.org</i>	12
3.2	Aktive Nutzer Weltweit	14
3.3	Anzahl Schweizer Nutzer	14
5.1	Nutzungsanteil CMS weltweit <i>Quelle:de.statista.com</i>	23

B.2 Quellenverzeichnis

B.3 Tabellenverzeichnis

2.1 Soll/Ist Analyse	10
--------------------------------	----

Bradner, S. 1997. "Key Words for Use in RFCs to Indicate Requirement Levels." <https://www.ietf.org/rfc/rfc2119.txt>.

Burling, Stacey. 2012. "CAPTCHA: The Story Behind Those Squiggly Computer Letters." <http://phys.org/news/2012-06-captcha-story-squiggly-letters.html>.

Duden. 2014. Vol. 26. Dudenredaktion.

Hanik, Filip. 2015. "Kiss." <https://people.apache.org/~fhanik/kiss.html>.

"[Http://authentifizierung.org/](http://authentifizierung.org/)" 2015. <http://authentifizierung.org/>.

Interactive, Goldbach. 2015. "Nutzerzahlen Der Wichtigsten Plattformen." <https://twitter.com/revogt/>.

"Interview Mit Shaul Olmert." 2015. https://www.youtube.com/watch?v=X_fQ1uG9rFY.

Kriha, Walter, and Roland Schmitz. 2009. *Sichere Systeme*. Xpert.press. Berlin, Heidelberg: Springer Berlin Heidelberg.

Millischer, Sven. 2015. "Die Digitale Revolution." handelszeitung.ch/digitalisierung/hz-sonderausgabe-die-digitale-revolution-874557.

"NET-Matrix-Audit." 2004. news.admin.ch/message/index.html?lang=de&msg-id=13600.

"NET-Matrix-Audit." 2015. <http://netreport.net-matrix.ch/audit/>.

"PlayBuzz." 2015. <http://www.playbuzz.com>.

"ReCAPTCHA Digitization Accuracy." 2014. <http://www.google.com/recaptcha/digitizing>.

Rothman, Mike. 2015. "Default Deny." <https://securosis.com/blog/network-security-fundamentals-d>

Rouse, Margaret. 2015. "Authentifizierung - Definition." <http://www.searchsecurity.de/definition/Authentifizierung>.

"SMI (SocialMedia Institute)." 2015. <http://socialmedia-institute.com/>.

"Statistik Plattform." 2015. <http://de.statista.com/>.

Stern, Olaf. 2012. *Reglement Bachelorarbeit*. Zürcher Hochschule für Angewandte Wissenschaften.

"Top 10 CMS November 2015." 2015. <http://de.statista.com/statistik/daten/studie/320685/umfrage/nutzungsanteil-der-content-management-systeme-cms-weltweit/>.

"Usage of Content Management Systems for Websites." 2015. http://w3techs.com/technologies/overview/content_management/all.