



Diplomarbeit

Höhere Technische Bundeslehranstalt Leonding
Abteilung für Informatik

AKKalytics

Eingereicht von: **Felix Kronsteinerr, 5AHIF**
Adzamija Alen, 5AHIF
Klatzer Emanuel, 5AHIF

Datum: **4. April 2018**

Betreuer: **Gerhard Aistleitner**

Projektpartner: **MIC**

Declaration of Academic Honesty

Hereby, I declare that I have composed the presented paper independently on my own and without any other resources than the ones indicated. All thoughts taken directly or indirectly from external sources are properly denoted as such.

This paper has neither been previously submitted to another authority nor has it been published yet.

Leonding, April 4, 2018

Felix Kronsteinerr, Adzamija Alen, Klatzer Emanuel

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorgelegte Diplomarbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Gedanken, die aus fremden Quellen direkt oder indirekt übernommen wurden, sind als solche gekennzeichnet.

Die Arbeit wurde bisher in gleicher oder ähnlicher Weise keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Leonding, am 4. April 2018

Felix Kronsteinerr, Adzamija Alen, Klatzer Emanuel

Zusammenfassung

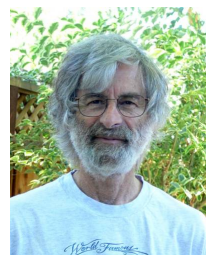
Here it is described what the thesis is all about. The abstract shall be brief and concise and its size shall not go beyond one page. Furthermore it has no chapters, sections etc. Paragraphs can be used to structure the abstract. If necessary one can also use bullet point lists but care must be taken that also in this part of the text full sentences and a clearly readable structure are required.

Concerning the content the following points shall be covered.

1. *Definition of the project:* What do we currently know about the topic or on which results can the work be based? What is the goal of the project? Who can use the results of the project?
2. *Implementation:* What are the tools and methods used to implement the project?
3. *Results:* What is the final result of the project?

This list does not mean that the abstract must strictly follow this structure. Rather it should be understood in that way that these points shall be described such that the reader is animated to dig further into the thesis.

Finally it is required to add a representative image which describes your project best. The image here shows Leslie Lamport the inventor of \LaTeX .



Zusammenfassung

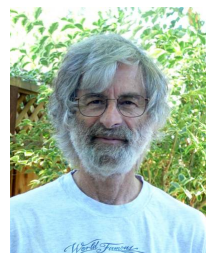
An dieser Stelle wird beschrieben, worum es in der Diplomarbeit geht. Die Zusammenfassung soll kurz und prägnant sein und den Umfang einer Seite nicht übersteigen. Weiters ist zu beachten, dass hier keine Kapitel oder Abschnitte zur Strukturierung verwendet werden. Die Verwendung von Absätzen ist zulässig. Wenn notwendig, können auch Aufzählungslisten verwendet werden. Dabei ist aber zu beachten, dass auch in der Zusammenfassung vollständige Sätze gefordert sind.

Bezüglich des Inhalts sollen folgende Punkte in der Zusammenfassung vorkommen:

- *Aufgabenstellung*: Von welchem Wissenstand kann man im Umfeld der Aufgabenstellung ausgehen? Was ist das Ziel des Projekts? Wer kann die Ergebnisse der Arbeit benutzen?
- *Umsetzung*: Welche fachtheoretischen oder -praktischen Methoden wurden bei der Umsetzung verwendet?
- *Ergebnisse*: Was ist das endgültige Ergebnis der Arbeit?

Diese Liste soll als Sammlung von inhaltlichen Punkten für die Zusammenfassung verstanden werden. Die konkrete Gliederung und Reihung der Punkte ist den Autoren überlassen. Zu beachten ist, dass der/die LeserIn beim Lesen dieses Teils Lust bekommt, diese Arbeit weiter zu lesen.

Abschließend soll die Zusammenfassung noch ein Foto zeigen, das das beschriebene Projekt am besten repräsentiert. Das folgende Bild zeigt Leslie Lamport, den Erfinder von \LaTeX .



Acknowledgments

If you feel like saying thanks to your grandma and/or other relatives.

Inhaltsverzeichnis

1	Einleitung	5
1.1	Ausgangslage	5
1.2	Ziele	5
1.3	Übersicht	5
1.4	Basic Terminology	6
1.5	Related Work and Projects	6
1.6	Structure of the Thesis	6
2	Aufgabenstellung	7
3	Lösungsansatz	8
4	Architektur	9
5	Datenquellen und Ihr Aufbau	10
6	Extract	11
7	Transform	12
8	Load	13
9	Visualisierung	14
9.1	Begriffsklärung	14
9.2	Kapitel2	14
9.3	Kapitel3	14
9.4	Kapitel4	14
10	Die Arbeitsumgebung	15
10.1	AWS	15
10.2	Linux	15
10.3	Portainer	15
10.4	IntelliJ	15
10.5	Vim	15

10.6 Docker	15
11 Verwendete Technologien	16
11.1 Kapitel1	16
11.2 Kapitel2	16
11.3 Kapitel3	16
11.4 Kapitel4	16
12 Databus	17
13 Kotlin	18
14 Gradle	19
15 Kafka	20
16 Portainer	21
17 Load	22
18 NO SQL	23
19 Python	24
20 Spark	25
21 Was sind Visualisierungs Applikationen (FK)	26
21.1 Warum Visualisierungs tools	26
22 Der Unterschied zwischen Javascript und Zero Code	27
22.1 JavaScript Frameworks	27
22.1.1 D3	27
22.1.2 Google Charts	28
22.2 Zero Code Frameworks	28
22.2.1 Tableau	28
22.2.2 Infogram	29
22.2.3 Chartblock	30
22.2.4 Datawrapper	30
22.2.5 Qlik	31
22.2.6 Power Bi	31
22.2.7 Kibana	32
22.3 Andere Tools	33
22.3.1 Dash by Plotly	33
22.4 Vergleich der Visualisierungs Applikationen	34
22.4.1 Dritter Platz: Qlik	34
22.4.2 Zweiter Platz: Power Bi	34

22.4.3	Erster Platz: Kibana	35
23	Der Visualisierungsprozess	36
23.1	Einrichten von AWS	36
23.2	Installierung von Elasticsearch und Kibana	36
23.3	Die REST-API Schnittstelle von Elasticsearch	36
23.4	Die Erstellung von Grafiken mit Kibana	36
23.5	Probleme beim Visualisierungsprozess	36
24	Summary	37
A	Additional Information	42
B	Individual Goals	43

Kapitel 1

Einleitung

1.1 Ausgangslage

Common word processors do not prepare print-like documents in so far as these programs do not reflect the rules of professional printing which have been grown over centuries. These rules contain clear requirements for balancing page layouts, the amount of white space on pages, font-handling, etc. Donald Knuth's TeX package (see [?]) is a word processor which conforms to these printing rules. This package was enhanced by Leslie Lamport by providing more text structuring commands. He called his package LaTeX [?].

When preparing a thesis, we want not only to have our content on a top level, we also want to commit to a high level of formal criteria. Therefore, we request our students to use one of these professional printing production environments like TeX or LaTeX.

Furthermore students should train their scientific writing skills. This includes a clear and structured break-down of their ideas, a high-level and clear wording, and the training of transparent citations of ideas from other sources than from theirs. A good source for more information concerning technical and scientific writing can be found in [?].

1.2 Ziele

The general goals and objectives of the project are described here. Care must be taken that the goals documented here are purely project goals and have nothing to do with individual goals of the team members. If individual goals should be part of the thesis they are listed in appendix B.

1.3 Übersicht

Details of the diploma thesis have to be aligned between student and supervisor. This should be a basic structure to facilitate the first steps when students start to write their theses.



Abbildung 1.1: Don Knuth, the inventor of T_EX

Never forget to add some illustrative images. Images must not be messed up with your normal text. They are encapsulated in floating bodies and referenced in your text. An example can be seen in figure 1.1. As you can see, figures are placed by default on top of the page nearby the place where they are referenced the first time. Furthermore you can see that a list of figures is maintained automatically which can be included easily by typing the command `\listoffigures` into your document.

1.4 Basic Terminology

As usual the very basic terminology is briefly explained here. Most probably the explanations here only scratch a surface level. More detailed explanations of terminology goes into chapter ??.

1.5 Related Work and Projects

Here a survey of other work in and around the area of the thesis is given. The reader shall see that the authors of the thesis know their field well and understand the developments there. Furthermore here is a good place to show what relevance the thesis in its field has.

1.6 Structure of the Thesis

Finally the reader is given a brief description what (s)he can expect in the thesis. Each chapter is introduced with a paragraph roughly describing its content.

Kapitel 2

Aufgabenstellung

Kapitel 3

Lösungsansatz

Kapitel 4

Architektur

Kapitel 5

Datenquellen und Ihr Aufbau

Kapitel 6

Extract

Kapitel 7

Transform

Kapitel 8

Load

Kapitel 9

Visualisierung

9.1 Begriffsklärung

9.2 Kapitel2

9.3 Kapitel3

9.4 Kapitel4

Kapitel 10

Die Arbeitsumgebung

10.1 AWS

10.2 Linux

10.3 Portainer

10.4 IntelliJ

10.5 Vim

10.6 Docker

Kapitel 11

Verwendete Technologien

11.1 Kapitel1

11.2 Kapitel2

11.3 Kapitel3

11.4 Kapitel4

Kapitel 12

Databus

Kapitel 13

Kotlin

Kapitel 14

Gradle

Kapitel 15

Kafka

Kapitel 16

Portainer

Kapitel 17

Load

Kapitel 18

NO SQL

Kapitel 19

Python

Kapitel 20

Spark

Kapitel 21

Was sind Visualisierungs Applikationen (FK)

Visualisierungs Applikationen sind eine Möglichkeit Daten als Grafiken widerzuspiegeln. Die Daten können auf vielfältige Art und Weise zusammengestellt werden. Nicht jede Visualisierungs Applikation kann dieselben Datenquellen benutzen und verarbeiten. In Microsoft Word kann man zum Beispiel ausschließlich Microsoft Excel Tabellen für die Diagramme einfügen. In Spezialisierteren Applikationen können weitaus mehr Datenquellen verwendet werden wie Daten direkt aus einer Datenbank auslesen oder mit einer REST Schnittstelle von einem Server zu bekommen.

21.1 Warum Visualisierungs tools

Viele Unternehmen sammeln täglich riesige Mengen an Daten über ihre Kunden, Mitarbeiter, den Markt und viele andere Dinge, aber dieses Wissen ist einzeln betrachtet in bedeutungslosen Zahlen versteckt. Kombiniert man aber diese Daten könnten sie analysiert und visualisiert werden, um benutzerfreundliche und verständliche Diagramme zu erstellen, die helfen können, Marktveränderungen vorherzusagen oder den Gewinn durch intelligente Kostenreduzierung zu erhöhen. Dies könnte entweder von einem Betreiber genutzt werden, der ansprechende Diagramme für seine Präsentationen und Berichte verwendet, oder von Datenanalysten, die diese Diagramme studieren und nützliche Informationen ableiten, um Fragen zu beantworten wie "Warum steigt der Markt in Großbritannien, obwohl es den Brexit gibt, und wie können wir das, was in dieser Region funktioniert, in andere Regionen bringen?"

Kapitel 22

Der Unterschied zwischen Javascript und Zero Code

Die Entscheidung ob mit JavaScript Frameworks oder „Zero Code“ Frameworks gearbeitet werden soll muss bereits früh im Projekt geklärt werden weil es enorme Auswirkungen hat auf den Aufwand der eingeplant werden muss.

22.1 JavaScript Frameworks

JavaScript ist eine Skriptsprache und wurde erschaffen um CSS und HTML zu erweitern. Es kann verwendet werden um Websites Interaktiver zu machen sodass der Benutzer nicht jedes Mal eine vollständige neue Seite laden muss wenn er zum Beispiel eine neue Filteroption einer Liste auswählt. JavaScript kann aber noch viel mehr. Vgl JAVASCRIPT !!! In unserem Fall könnte man JavaScript verwenden um interaktive Grafiken auf einer Website zu platzieren. Dies hätte den Vorteil das der Kunde das Diagramm nach seinen Wünschen anpassen könnte um die Daten anzuzeigen welche er gerade benötigt. Nachteile bei der Verwendung von diversen JavaScript Frameworks sind zum Beispiel der Große Aufwand der von Nöten ist weil man einen vollständigen Server mit Datenbankzugriff selber ausimplementieren muss. Die Interaktivität muss man dabei ebenfalls selber integrieren. Um einen besseren Überblick über die vorhandenen Libraries zu bekommen habe ich folgende untersucht:

22.1.1 D3

„D3.js ist eine JavaScript-Bibliothek zur Manipulation von Dokumenten auf Basis von Daten. D3 hilft Ihnen, Daten mit Hilfe von HTML, SVG und CSS zum Leben zu erwecken. Die Betonung von D3 auf Webstandards gibt Ihnen die vollen Möglichkeiten moderner Browser, ohne sich an ein proprietäres Framework zu binden, und kombiniert leistungsstarke Visualisierungskomponenten mit einem datengesteuerten Ansatz zur DOM-Manipulation.“ [Bos19]

22.1.2 Google Charts

Google Charts ist sehr ähnlich zu D3(22.1.1)

22.2 Zero Code Frameworks

Wie die Überschrift bereits vermuten lässt handelt es sich bei „Zero Code“ Frameworks um Applikationen die bereits vollständig implementiert worden sind. Der Arbeitsaufwand hält sich dementsprechend auch in gewissen Grenzen. Was aber nicht bedeutet das gar keine Zeit investiert werden muss denn zusätzlich zu dem Installations Vorgang muss auch eine Datenbank- oder eventuell REST-Schnittstelle implementiert werden. Nachdem unser Arbeitgeber MIC die Vorgabe gab das wir mit einem „Zero Code“ Framework arbeiten sollen folgt nun eine detailliertere Form der getesteten Frameworks:

22.2.1 Tableau

Tableau ist ein kostenpflichtiges Datenvisualisierungstool, das für große Daten in Unternehmen optimiert ist. Obwohl es eine überwältigende Menge an Möglichkeiten hat und dies einige Zeit zum Erlernen benötigt, ist es besonders für Nicht-Techniker sehr einfach zu bedienen. Einmal richtig aufgesetzt können Datenfelder mit Hilfe von Drag n Drop einfach in das Diagramm gezogen werden und die Daten erscheinen im Diagramm. Es hilft auch bei der Auswahl der besten Grafik für jede Datenquelle indem es Vorschläge anzeigt. Hat der Benutzer ein Diagramm fertig kann er es auch in die Cloud stellen damit alle berechtigten Nutzer darauf zugreifen können.

Vorteile

- Einfaches Einfügen der Daten durch Drag n Drop
- Visualisierungen einfach über die Cloud teilen
- Auf verschiedensten Geräten verwendbar

Nachteile

- Vergleichsweise Teuer
- Nur als Desktop App verfügbar
- Schwierig zu Erlernen

Tableau Preise

Ein weiterer Wunsch von MIC war es das die verwendeten Technologien Gratis sind oder nicht viel kosten. Tableau erreichte diese Anforderung eindeutig nicht was die folgende Abbildung 22.1 über die Preispolitik zeigt. Zusätzlich kann man noch extra Pakete dazu bestellen



Abbildung 22.1: Tableau Preispolitik

22.2.2 Infogram

Infogram ist ein Online-Werkzeug zur Erstellung von Grafiken und Diagrammen. Es bietet 5 verschiedene Datenbanken (MySQL, Postgre, Amazon Redshift, Oracle und MS SQL Server), aber keine REST-Schnittstelle, da es hauptsächlich dazu dient, hübsche Grafiken für Präsentationen und Berichte zu erstellen. Außerdem kann man Daten per Upload importieren, mit Google Drive, Drop Box und Json.

Vorteile

- Sehr gut verwendbar für Präsentationen
- Einfach und Intuitiv zu benutzen
- Online Teamarbeit an einem Document
- Unglaublich schöne Grafiken

Nachteile

- Vergleichsweise Teuer
- Keine Analysefunktionen
- Nur eine kleine Auswahl an Funktionen und Grafiken
- Keine REST-Schnittstelle

Infogram Preise

Infogram ist für seine Funktionen vergleichsweise teuer, ähnlich zu Tableau was in folgender Abbildung zu sehen ist 22.2

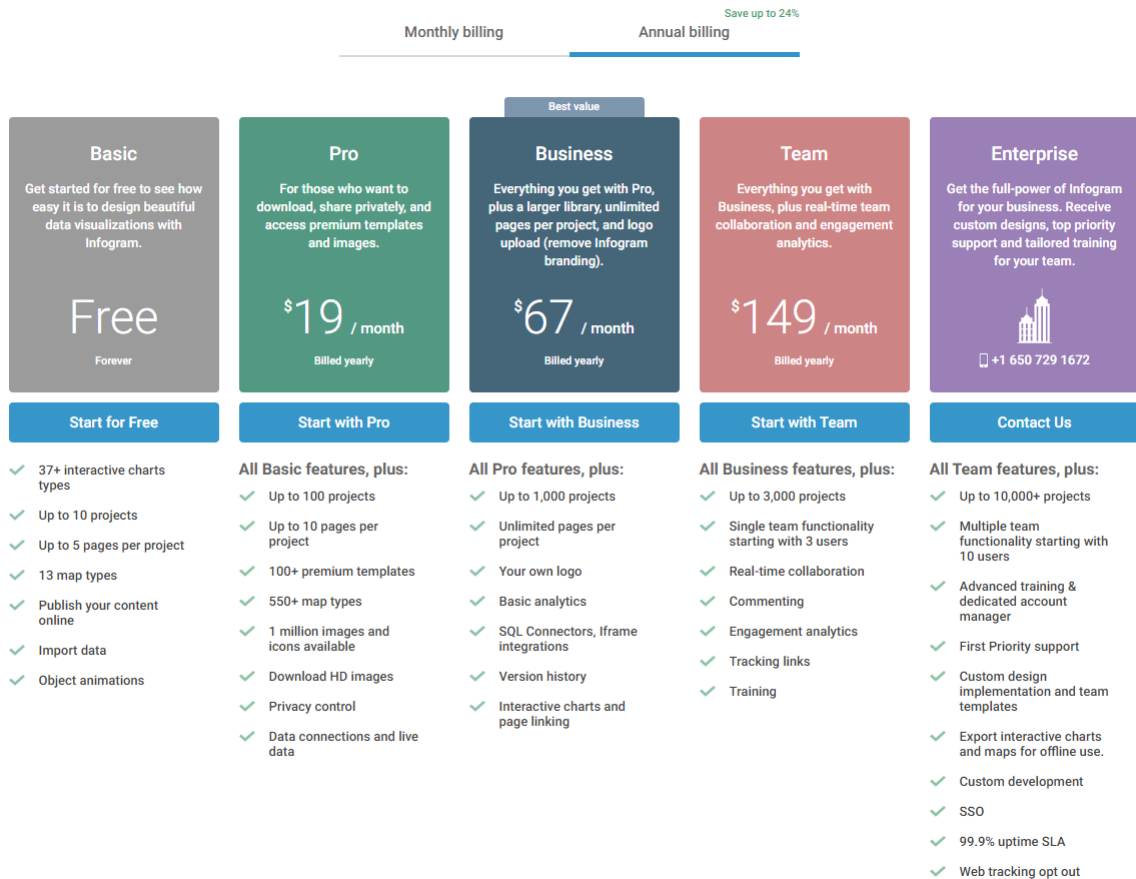


Abbildung 22.2: Infogram Preispolitik

22.2.3 Chartblock

Chartblock ist ein sehr einfaches Werkzeug, mit dem man Diagramme in einer Wizard-Form(Schritt-für-Schritt Anleitung) erstellen kann. Zur Zeit der Entscheidung welches Tool verwendet werden soll hatte es keine Möglichkeit daten außer mit csv, Excel oder Manuell zu importieren, daher ist es für unseren Anwendungsfall derzeit nicht relevant

22.2.4 Datawrapper

Datawrapper ist ähnlich zu Chartblock jedoch liegt der Fokus auf Diagramme für Journalisten. Es hat ebenso keine Möglichkeit Daten aus einer Datenbank oder einer REST-Schnittstelle zu entnehmen weswegen es ebenfalls nicht infrage kommt.

22.2.5 Qlik

Qlik ist ein Tool mit Schwerpunkt Data Analytics. Es kann aber auch Diagramme zeichnen. Qlik kann Daten aus allen gängigen Quellen verarbeiten und es schlägt dabei auch vor wie Datensätze verbunden werden könnten was einiges an Aufwand spart. Auch bei der gemeinsamen Erstellung und Verteilung von Daten hilft Qlik. Sodass auch Nicht-Techniker die Informationen Nützen und verwenden können

Vorteile

- Code freie Daten Analysefunktionen mit Künstlicher Intelligenz
- Auf allen gängigen Plattformen benutzbar
- Legt großen Wert auf Zusammenarbeit
- Viele Datenimport Möglichkeiten darunter REST
- Hohe Skalierbarkeit mit Multi-Nodes
- Interaktive Diagramme

Nachteile

- Der Preis ist nicht ersichtlich, weil er für jeden Kunden einzeln berechnet wird
- Es gibt Programme die deutlich schönere Grafiken und Diagramme erzeugen

22.2.6 Power Bi

PowerBI ist das Business-Intelligence-Werkzeug von Microsoft. Es ist einfach zu bedienen und hat sehr schöne Diagramme out-of-the-box. Leider konnte ich die REST-API nicht ausprobieren, weil man das online registrieren muss und Admin-Berechtigungen von MIC benötigt. Es gibt aber Testdaten mit denen man zumindest den Diagramm Teil der Applikation ausprobieren kann. Die Benutzung ist sehr einfach auch ohne technische Erfahrung und die Diagramme sind wirklich sehr schön.

- Interaktive Diagramme
- Sehr Intuitiv in der Benutzung auch ohne Technische Erfahrung
- Sehr schöne Diagramme
- Es können Datenabfragen mit Volltextsuche gemacht werden (nur auf Englisch)
- Eigene Analysefunktionen geschrieben in Python oder R
- Auf allen gängigen Plattformen benutzbar

Nachteile

- Der Preis ist nicht ersichtlich, weil er für jeden Kunden einzeln berechnet wird
- Es gibt Programme die deutlich schönere Grafiken und Diagramme erzeugen

Die Tatsache das Power Bi fast ausschließlich in der Cloud läuft kann sowohl Vorteile als auch Nachteile bieten. Ein klarer Vorteil dabei wäre das sich MIC nicht eigene Server bauen oder vorhandene Server belasten müsste. Ein Nachteil wäre die starke Abhängigkeit von dem Cloud

Power Bi Preispolitik

Microsoft verlangt für seinen Enterprise Service nicht Pro Benutzer sondern Pro benutzer Rechen- und Speicherkapazität. Wie in Folgender Abbildung ersichtlich wird gibt es aber auch ein Pro Benutzer Abonnement:

The image shows a comparison of two Power BI pricing tiers. On the left, under the heading 'Benötigen Sie Self-Service-BI?', is the 'Power BI Pro wählen' section. It lists features: 'Self-Service und moderne BI in der Cloud', 'Zusammenarbeit, Veröffentlichung, Freigabe und Ad-hoc-Analysen', and 'Vollständig von Microsoft verwaltet'. Below this, a yellow box highlights 'Power BI Pro' with a price of '\$9.99' and the note 'Monthly price per user with annual subscription'. A button labeled 'Erste Schritte >' is at the bottom. On the right, under the heading 'Benötigen Sie erweiterte Analysen, Big Data-Unterstützung sowie lokale und Cloudberichte?', is the 'Power BI Premium hinzufügen' section. It lists features: 'Enterprise BI, Big Data-Analysen, Cloud- und lokale Berichterstellung', 'Erweiterte Steuerungsmöglichkeiten für Verwaltung und Bereitstellung', 'Dedizierte Cloud Computing- und Speicherressourcen', and 'Ermöglicht allen Benutzern die Nutzung von Power BI-Inhalten'. Below this, a yellow box highlights 'Power BI Premium' with a price of '\$4,995' and the note 'Monthly price per dedicated cloud compute and storage resource with annual subscription'. A button labeled 'Beratung anfordern >' is at the bottom.

Option	Benötigen Sie Self-Service-BI?	Benötigen Sie erweiterte Analysen, Big Data-Unterstützung sowie lokale und Cloudberichte?
Power BI Pro wählen	<ul style="list-style-type: none">• Self-Service und moderne BI in der Cloud• Zusammenarbeit, Veröffentlichung, Freigabe und Ad-hoc-Analysen• Vollständig von Microsoft verwaltet	Power BI Premium hinzufügen <ul style="list-style-type: none">• Enterprise BI, Big Data-Analysen, Cloud- und lokale Berichterstellung• Erweiterte Steuerungsmöglichkeiten für Verwaltung und Bereitstellung• Dedizierte Cloud Computing- und Speicherressourcen• Ermöglicht allen Benutzern die Nutzung von Power BI-Inhalten
Power BI Pro	\$9.99 Monthly price per user with annual subscription	Power BI Premium \$4,995 Monthly price per dedicated cloud compute and storage resource with annual subscription
Erste Schritte >		Beratung anfordern >

Abbildung 22.3: PowerBI Preispolitik

22.2.7 Kibana

Kibana ist ein grafisches Werkzeug, das auf Elasticsearch basiert. Elasticsearch ist eine Dokumentbasierte NoSQL Datenbank welche durch Features wie Volltextsuche und extrem schnelle Suchabfragen glänzt. Eine weitere tolle Eigenheit ist das Kibana sowohl als auch Elasticsearch schnell über Docker aufgesetzt werden können und sofort Startbereit sind und Daten über REST-Schnittstelle oder Zugriff auf externe Datenbanken

importiert werden können. Kibana bietet aber auch out-of-the-box die Möglichkeit Musterdatensätze zu verwenden um einen ersten Eindruck zu erhalten. Kibana dient nicht nur zur Visualisierung, sondern auch zur Überwachung von Abfragen auf Daten. So kann z.B. kontrolliert werden von wem die Daten benutzt werden.

- Umfassendes Berechtigungssystem
- Sehr einfach aufzusetzen
- Schöne Diagramme
- Es können Datenabfragen mit Volltextsuche gemacht werden (nur auf Englisch)
- Sehr schnell durch die Elasticsearch Suchmaschine
- REST-Schnittstelle verfügbar
- Hochgradig Interaktive Diagramme
- Sehr gute Skalierbarkeit dank Multi-Nodes
- Opensource und Großteils Gratis

Nachteile

- Erweiterte Sicherheitsfunktionen, Multi-Nodes, Machine Learning und Support kosten etwas
- Diagramme erstellen dauert etwas ist etwas komplexer als bei anderen Visualisierungs Programmen

22.3 Andere Tools

22.3.1 Dash by Plotly

Dash ist eine Open-Source-Bibliothek für R und Python. Es wird kein JavaScript benötigt, aber Python. Mit Python schreibt man sowohl das Layout der Website sowohl als auch die Logik des Servers dahinter. Somit erreicht es einen ähnlich großen Funktionsumfang wie oben genannte JavaScript Frameworks mit dem bedeutenden Unterschied das Python mit wenigen Zeilen Code sehr viel mehr bewirken kann als JavaScript. Jedoch muss hier jedes Diagramm und Dashboard von einem Programmierer Entwickelt werden. Einfache Kunden können somit nur schwer Änderungen an dem Diagramm vornehmen oder gar neue erstellen.

Vorteile

- Gratis benutzung der Bibliotheken. Nur der Support ist kostenpflichtig
- Kann gut in Websites eingebunden werden
- REST-Schnittstelle kann selber geschrieben werden
- Die Interaktivität der Diagramme wird von den Dash Bibliotheken unterstützt =, Weniger Aufwand bei der Implementierung

Nachteile

- Implementierungen notwendig
- Analysefunktionen müssen selber geschrieben werden
- Nicht-Techniker können keine eigenen Diagramme erstellen

22.4 Vergleich der Visualisierungs Applikationen

Es gibt noch viel mehr Visualisierungsprogramme aber die wohl wichtigsten / bekanntesten sind hier gelistet. Eine paar der oben gelisteten Programme sind Nischenprodukte, es wird bestimmte Einsatzzwecke für Applikationen wie z.B. Chartblock geben jedoch werden diese nicht für die folgende Top 3 reichen. Zur Erinnerung hier die Anforderungen von MIC für ein Visualisierungs Tool. REFERENZ EINFÜGEN

22.4.1 Dritter Platz: Qlik

Qlik ist nicht schlecht, aber im Vergleich zu Kibana und PowerBI ist das Einrichten von Beziehungen und das Erstellen von Diagrammen etwas komplexer. Wenn man z.B. Daten von einer REST-Verbindung erhält, erzeugt es einige Spalten mehr als im Original-Datensatz, was sehr irritierend ist und für einen erhöhten Aufwand in der REST-Schnittstellenimplementierung sorgt. Ein weiterer wichtiger Punkt ist, dass Qlik-Benutzer sich über den hohen RAM-Bedarf beschwerten welcher bei der geplanten Hochskalierung der Hardwareressourcen zu bedeutenden Kostenerhöhungen führen wird. Somit fällt Qlik trotz der guten Analytischen Funktionen auf Platz 3.

22.4.2 Zweiter Platz: Power Bi

Die Entscheidung zwischen PowerBI und Kibana war eine schwere Wahl, weil PowerBI wirklich schön und einfach zu bedienen ist. Darüber hinaus hat es einige Features wie die "Voll Text ERKLÄRUNG REFERENZIEREN Abfragen und On-Click-Analyse", die sie sehr gut machen. Leider konnte ich die REST-API aufgrund eines komplexen Azure (Cloud von Microsoft) Registrierungsmechanismus nicht überprüfen. Ein entscheidender Faktor war die Auslieferung (Deployment) von PowerBI. Es gibt nur ein zusätzliches Tool

namens PowerBI Report Server“, das nur dazu dient, Berichte innerhalb einer Firma zu verbreiten, die vorher in PowerBI Desktop gemacht wurden. Dieses eine Programm läuft auf einem Server in der Firma die es einsetzt. Der ganze Rest Verarbeitung der Visualisierung und der Analytischen Funktionen übernimmt die Microsoft Cloud Azure. Dadurch wird das Produkt nicht nur deutlich teurer, sondern die Firma wird auch sehr viel stärker an Microsoft gebunden.

22.4.3 Erster Platz: Kibana

Kibana ist gleich in den meisten Punkten mit PowerBI, aber es gibt zwei wichtige Punkte, die Kibana ein wenig besser machen:

1. Kibana und Elasticsearch sind öut-of-the-boxfür den Einsatz auf Servern konzipiert, was die Bereitstellung und den Zugriff von vielen verschiedenen Geräten aus erleichtert, ohne dass eine Desktop-Applikation installiert werden muss.
2. Besonders Elasticsearch ist (wie beworben) sehr schnell, da es für den Anwendungsfall gebaut ist für Analysefunktionen niedrige Zugriffszeiten zu bieten.

Kapitel 23

Der Visualisierungsprozess

23.1 Einrichten von AWS

23.2 Installierung von Elasticsearch und Kibana

23.3 Die REST-API Schnittstelle von Elasticsearch

23.4 Die Erstellung von Grafiken mit Kibana

23.5 Probleme beim Visualisierungsprozess

Kapitel 24

Summary

Here you give a summary of your results and experiences. You can add also some design alternatives you considered, but kicked out later. Furthermore you might have some ideas how to drive the work you accomplished in further directions.

Literaturverzeichnis

- [Bos19] Mike Bostock. D3.js - Data-Driven Documents, July 2019. URL: <https://d3js.org/>.

Abbildungsverzeichnis

1.1	Don Knuth, the inventor of T _E X	6
22.1	Tableau Preispolitik	29
22.2	Infogram Preispolitik	30
22.3	PowerBI Preispolitik	32

Tabellenverzeichnis

Project Log Book

Date	Participants	Todos	Due
------	--------------	-------	-----

Anhang A

Additional Information

If needed the appendix is the place where additional information concerning your thesis goes. Examples could be:

- Source Code
- Test Protocols
- Project Proposal
- Project Plan
- Individual Goals
- ...

Again this has to be aligned with the supervisor.

Anhang B

Individual Goals

This is just another example to show what content could go into the appendix.