

Multiplattform-fähiges User Interface und Datenmanagement für die Wetterstation Arbon (Fachmodul)

Ladina Bilgery Thomas Wieling

17. Dezember 2017

Inhaltsverzeichnis

1	Darstellungsprobleme auf der Webseite	2
1.1	Adobe Flash: Workaround mit Nachteilen	2
1.2	Wetterdaten ohne Responsive Design	3
1.3	Verwirrende Graphen	4
1.4	Unsichere Sturmwarnung mit Öffnungszeiten	4
2	Datenmanagement mit Eigenheiten	6
2.1	Datenfluss	6
2.2	Datenmanagement und Datensicherung	8
2.3	API	9
3	Sensoren	9
3.1	Pegelmesser	9
3.2	Wassertemperatursensoren	10
4	Webcam-Steuerung	10
4.1	Positionsabhängige Zoombeschränkung	11
4.2	Warteschlange für Webcam-Steuerung	12
5	Erweiterungen	12
5.1	Individueller Benachrichtigungs-Service	12
5.2	Überprüfung der Windprognose-Genauigkeit	13
5.3	Berechnung und Darstellung der Wellenhöhe	13
5.4	Verlauf der Wassertemperatur in Abhängigkeit der Tiefe	13
6	Anforderungen	13
6.1	User Interface (UI)	14
6.2	Datenbank (DB)	16
6.3	Sensoren (TD)	17
6.4	Webcam (CA)	18
6.5	Nicht Funktionale Anforderungen (NF)	18
7	Projektmanagement	18
7.1	Vorgehensmodell	18
7.2	Entwicklungsprozess	19
7.3	Projektplan für die Bachelor-Arbeit	20
7.4	Dokumentation	20
7.5	Risikoanalyse	22
8	Rechtliche Ansprüche	22
9	Schluss	25

Einführung ins Thema

Die Wetterstation Arbon wurde 2005 als Lehrlingsarbeit des Berufsbildungszentrums Arbon auf Initiative der Technischen Gesellschaft Arbon (TGA) aufgebaut und in Betrieb genommen. Sie besteht aus mehreren Wettersensoren und einer Webcam, die auf einer Plattform auf dem See draussen montiert sind. Die Messwerte werden auf der Webseite¹ der Wetterstation Arbon angezeigt.



Abbildung 1: Installation und Webseite der Wetterstation Arbon

Was damals modern war, ist heute veraltet. Sowohl auf der Hardwareseite, als auch auf der Webseite gibt es diverses Reparatur- beziehungsweise Modernisierungspotential. Während des Fachmoduls führten wir eine Ist-Aufnahme der Wetterstation Arbon durch. Im Fokus lag sowohl die Hardware als auch die Software. Der Übersicht halber und damit wir die Arbeiten besser untereinander aufteilen konnten, haben wir die Themen in die vier Blöcke: Webseite, Datenbank und Webcam unterteilt. (Abb. 2)



Abbildung 2: Aufteilung in Arbeitsblöcke

Bei jedem Block führten wir eine Analyse der der ist-Situation durch und identifizierten die vorhandenen Probleme. Dieser Bericht zeigt jeweils pro Block auf, wo die Problemstellen liegen, wie diese behoben werden können und was die Anforderungen an die Lösung ist.

¹ <https://www.wetter-arbon.ch>

1 Darstellungsprobleme auf der Webseite

Die Webseite der Wetterstation Arbon besteht neben der Homepage aus zwölf Unterseiten. Für uns wichtig sind all jene, die mit den Sensordaten, der Webcam, oder der Datebank in Verbindung stehen. (hervorgehoben in Abb.3). Im folgenden werden diese Seiten und deren Probleme genauer erläutert.



Abbildung 3: Sitemap der Webseite

1.1 Adobe Flash: Workaround mit Nachteilen

Um die Daten des Wettertransmitters auszulesen, wird Weather Display² verwendet. Dieses Programm liest die Daten des Wettertransmitters und stellt sie anderen Programmen zum Beispiel in Form eines Text-Files zur Verfügung. Weather Display Live (WDL)³ liest nun eben dieses Text-File und erstellt damit eine Adobe Flash Animation wie in Abb.4 dargestellt (gelb markiert).

Adobe Flash war eine einfache Möglichkeit animierte Grafiken auf Webseiten darzustellen und wurde praktisch von allen Browsern, nach Installation des Plug-ins, unterstützt. Diverse Sicherheitslücken und der Umstand, dass es sich um eine proprietäre d.h. closed-source Software handelte, führten dazu, dass Apple entschied Adobe Flash auf ihrem Mobile-Betriebssystem iOS nicht mehr zu unterstützen. [5]

Sämtliche Adobe Flash Animationen können somit nicht auf iPhone und iPad angezeigt werden. Da ein Grossteil der schweizer Bevölkerung jedoch genau diese Mobilgeräte verwendet, wurde für die Wetterstation folgender Workaround geschaffen: Der Browser prüft zuerst, ob das Gerät Adobe Flash unterstützt. Wenn ja wird die normale Applikation geladen, wenn nicht wird ein Printscreen der Applikation geladen (Code: 1)

Listing 1: Adobe Flash workaround für iOS

```
1 if (swfobject.hasFlashPlayerVersion("1")) {document.write('
2 <iframe src="https://www.wetter-arbon.ch/WDL/index.html"></iframe>');
3 else {document.write('
4 ')}
```

Der Nachteil dieses Workarounds ist jedoch, dass die Anzeige weder dynamisch noch interaktiv ist. Um die aktuellen Werte zu erhalten muss die Seite jeweils neu geladen werden. Die interaktiven Elemente sind unbrauchbar d.h. die Änderung von Einheiten, Anzeige von Rekordwerten und weiteren Graphen ist nicht möglich.

² <http://www.weather-display.com/index.php>

³ <http://www.weather-display.de>

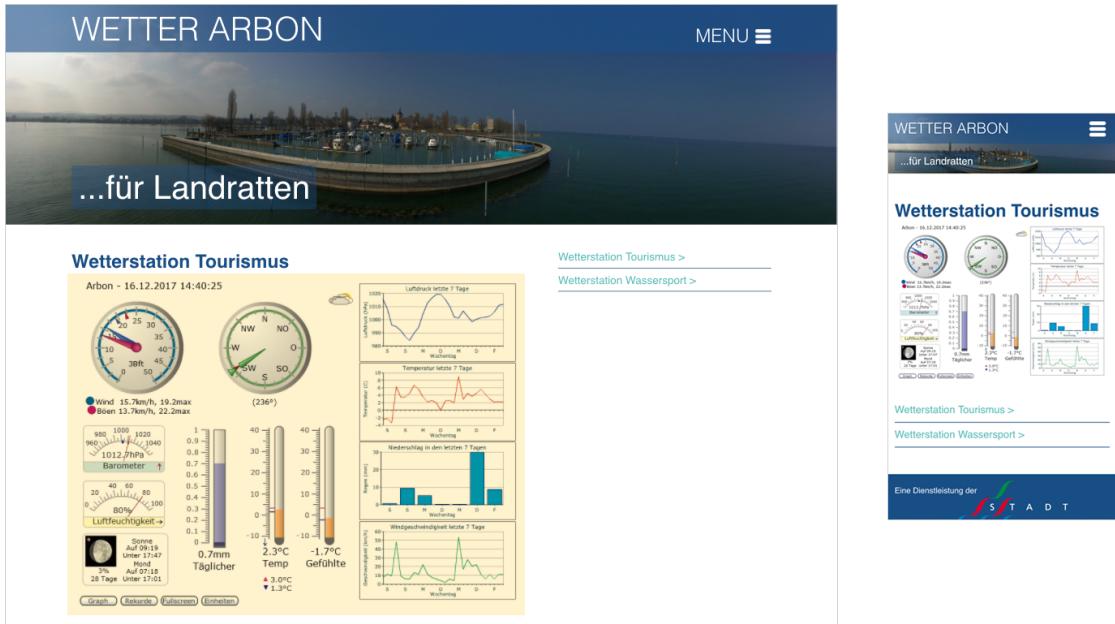


Abbildung 4: Responsive Design; Problem Flash-Applikation

Mit dem Aufkommen der Smartphones wurde Flash immer weiter verdrängt durch HTML5 und Javascript. Dadurch haben auch andere Hersteller von mobilen Geräten nachgezogen und auf Flash möglichst verzichtet.

Der wichtigste Grund ist das Adobe ab 2020 Flash nicht mehr weiterentwickelt und keine Updates mehr herausgeben wird. Dies aufgrund der oben genannten Tatsachen und das zur heutigen Zeit vielmals Flash umgangen wird und andere Lösungen gebraucht werden. Aufgrund von dies wird ab 2020 auch kein Browser mehr das Plugin zulassen und Seiten welche Flashinhalte benutzen sind nicht mehr Vollständig verfügbar. Somit ist auch die Wetter-Arbon Seite von dieser Tatsache betroffen. [4]

1.2 Wetterdaten ohne Responsive Design

Die Webseite der Wetterstation ist mit dem Content-Management-System (CMS) *Openfile64Light* der Firma Screenbox erstellt. Dieses unterstützt grundsätzlich responsive Design. Eigene Änderungen oder dynamische Inhalte, werden in Openfile64Light als sogenannte Applikationen behandelt und in die Seite eingebettet. Unterstützt die eingebettete Applikation kein responsive Design, so wird dieser Teil einfach linear skaliert. Dies führt dazu, dass die Anzeige der Wetterstationsdaten auf einem Mobilgerät kaum mehr lesbar sind. (Abbildung 4)

1.3 Verwirrende Graphen

Die Graphen auf den beiden Wetterseiten verwirren teilweise mehr, als das sie einem klar eine die vergangenen Werte zeigen. Das Ziel in der BA ist, dass die Aussagen der Graphen auf dem ersten Blick klar sind.

Automatische y-Skalierung von Graphen

Die Graphen, bspw. der Windanzeige mit ihrer Richtung sowie die Stärke, sind schwer lesbar, da die Skalierung des Graphen automatisch die, je nach Windstärke, wechselt. Das Bild 5 zeigt auf wie die Skalierung der Y-Achse bei unterschiedlichen Achsen wechselt. Beim linken Graph geht die Skalierung von 0 bis 30 Knoten, beim rechten von 0 bis 14 Knoten. Das Problem ist, dass ein schnelles ablesen der Anzeige nicht möglich ist.

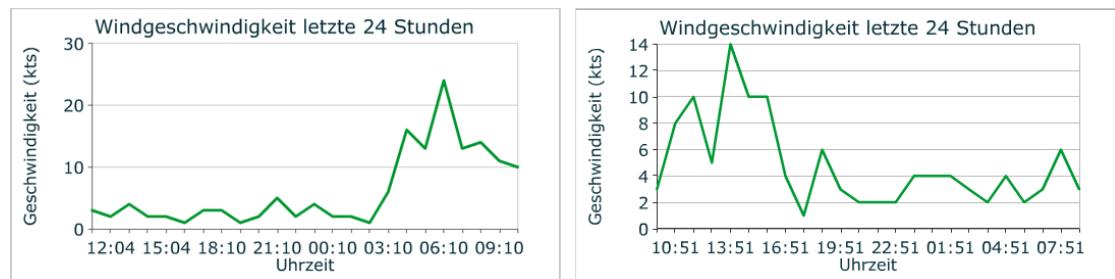


Abbildung 5: Anzeige der Windgeschwindigkeit mit variabler y-Skalierung

Windrichtung

Bild der Windrichtungsproblematik

Erklärung der Problematik

1.4 Unsichere Sturmwarnung mit Öffnungszeiten

Die Seite mir der Sturmwarnung, siehe 6, wurde anfangs November umgestellt aus Sicherheitsgründen, welche in der Problemanalyse erläutert werden. Zum jetzigen Zeitpunkt wird nur noch ein Link zur Verfügung gestellt um auf die kantonale Sturmwarnseite zu kommen. Die Daten dieser Seite werden mit dem deutschen Wetterdienst in Stuttgart sowie Meteo Schweiz erstellt und dienen auf der Webseite nur als Information. Zu beachten ist hierbei, dass die Sturmwarnungen Bürozeiten haben. D.h. konkret vom 1. April bis 31 Oktober zwischen 6 und 22 Uhr und vom 1. November bis 31. März zwischen 7 und 20 Uhr. Der deutsche Wetterdienst und Meteo Schweiz unterscheiden zwei verschiedene Kategorien. Zum einen starke Windböen zwischen 25 und 33 Knoten, dies wird 40 Blitze pro Minute an den Leuchten signalisiert. Zum anderen Sturmböen von 34 und mehr Knoten, welche mit 90 Blitze pro Minute signalisiert werden. Zusätzlich zu den beiden Kategorien wird der Bodensee in 3 verschiedene Zonen unterteilt, West, Mitte und Ost, wobei Arbon in zur Zone Ost gehört. Wie schon erklärt ist das Problem hierbei

der HTTP Standard, viele der Webbrower stellen die Unterstützung dieses Standards langsam aber sicher ein und werden dann nur noch HTTPS unterstützen [3]. Deswegen ist die Sturmwarnung nur über einen Link aufrufbar.

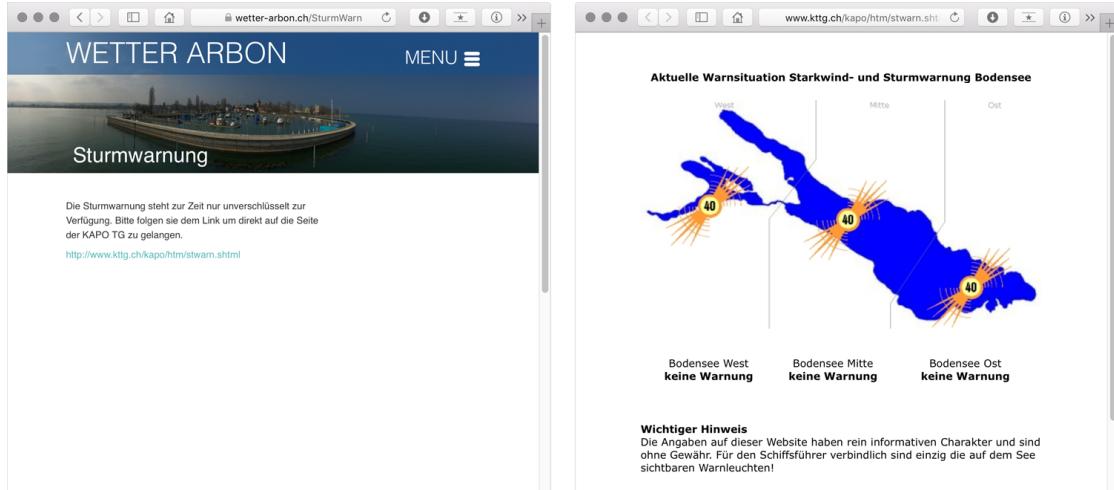


Abbildung 6: Sturmwarnung vom Kanton Thurgau

Das Problem der Webseite und vor allem der verschiedenen Applikationen ist, dass viele Geräte Flash nicht mehr oder in naher Zukunft nicht mehr unterstützen. Zusätzlich ist die Lösung mit dem Screenshot der aktuellen Verhältnisse auch keine optimale Lösung. Auch sind die Schreibfehler, welche entdeckt wurden bei näherer Betrachtung auch nicht Vorteilhaft. Weiter ist die Wetterapplikation nicht nach dem Prinzip responsive Design aufgebaut, welches in der heutigen Zeit ein wichtiger Bestandteil einer Webseite ist. Zusätzlich zum Flash, von der Gebrauch gemacht wird sind die Anzeigen auf der Touristik bzw. Wassersport Seite unschön.

Um die Webseite auf den neusten Stand der Technik zu bringen sollten folgende Änderungen durchgeführt werden. Die Flash-Software wird ausgemustert und die Applikation wird auf HTML5 und Javascript umgestellt. Die Webseite soll zudem im responsive Design entwickelt werden, damit auch auf mobilen Geräten die aktuelle Wetterlage sichtbar ist. Die dynamischen, sowie auch die teilweise statischen Anzeigen, werden wo möglich mithilfe der Javascript Bibliothek D3.js oder Google Charts erstellt, hiermit lassen sich ansehnliche und moderne Grafiken erstellen. Die Grafiken, sollten so gestaltet sein das auch Sehbehinderte Personen erkennen wie das Wetter momentan ist. Das heisst beispielweise, dass die Farben auch für Farbenblinde unterscheidbar sein sollten oder blinde Personen anhand eines Vorleseprogramms erkennen wie das Wetter ist. Ein weiterer Punkt ist die Auswahl der Einheiten, diese sollen nach dem ersten Besuch gespeichert bleiben beim Client mithilfe von Webstorage.

2 Datenmanagement mit Eigenheiten

2.1 Datenfluss

Damit alle erklärten Vorgänge besser verständlich sind, wurde ein Ablaufdiagramm erstellt mit der heutigen Situation und der zukünftigen erwünschten Situation. Die heutige Situation, braun, sieht folgendermassen aus. Die Daten werden von der Wetterstation an das Weather-Display gesendet. Dieses verarbeitet die Daten und schreibt sie in eine Textdatei, sowie im minuten Takt in die Datenbank. Das Weather-Display Live verarbeitet die Daten der Textdatei in eine dynamische Anzeige und erstellt gleichzeitig ein Bild davon. Die Webseite entscheidet dann, ob die dynamische oder die Bild-Anzeige angezeigt wird. Bei der mobilen Webseite wird in jedem Fall vom Bild Gebrauch gemacht. In Zukunft, blau, soll der Datenfluss bis zur Textdatei gleich sein. Anschliessend soll aber ein Javascript Anzeigen von den verarbeiteten Daten erstellen und auf die responsive Webseite angezeigt werden.

Für die Webseite und die Wetterstation, hat es vier verschiedene Datenbanken. Diese werden in diesem Kapitel einzeln behandelt und erklärt wie Sie zusammenhängen bzw. welche Rolle sie für die Webseite spielen. Die vier Datenbanken heissen:

- igwetter meteotmpl
- igwetter wettertest
- igwetter wp0
- igwetter openfile64Light

Die Datenbank igwetter meteotmpl beinhaltet alle relevanten Datenpunkte, d.h. von der Temperatur bis zur Windrichtung. In dieser Datenbank sind jedoch keine Daten vorhanden und dient momentan nur als Template.

Die zweite Datenbank welche nicht aktiv ist, heisst igwetter wp0, diese wurde für eine kurze Zeit für eine Wordpress-Seite benutzt.

Die dritte Datenbank ist die igwetter wettertest. Diese ist im Gegensatz zu den vorherigen beiden Datenbanken im Gebrauch. In der Tabelle wx data sind die Daten ab dem 25.02.2015 bis zum jetzigen Zeitpunkt gespeichert. Daten zwischen dem 14.07.2012 und 25.02.2015 sind nicht in der Datenbank hinterlegt. Vor diesem Zeitpunkt bis zum 25.02.2005 sind die täglichen Minimum, sowie Maximum Daten in der Tabelle tblgestern gespeichert.

Für das CMS ist die Datenbank opfile64Light relevant. Für die BA, sowie das Fachmodul ist nur die Tabelle applications interessant, dort werden die Applikationen unter einem bestimmten Namen abgespeichert und aufgerufen. Die Webseite weiss dann welche Datei sie öffnen muss, damit die Applikation läuft.

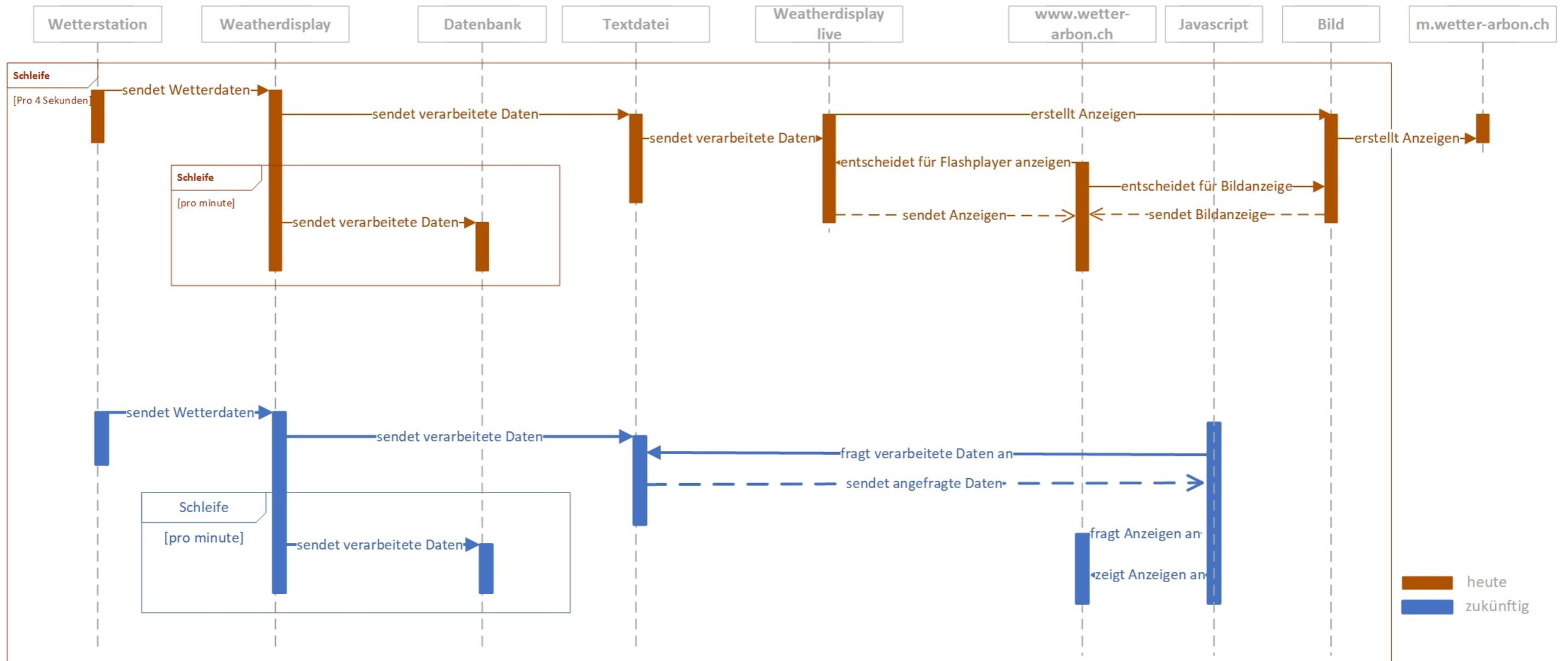


Abbildung 7: Ablauf von der Datenerfassung bis zur Anzeige

Die Weather-Display Software speichert zusätzlich alle Daten, welche von der Wetterstation sind, in ein .txt File. Das Weather-Display live nimmt die Daten direkt aus diesem File um die Anzeigen zu erstellen. Die Daten werden in 4 verschiedenen Textfiles mit unterschiedlichen Funktionen gespeichert.

- clientraw.txt
- clientrawextra.txt
- clientrawhour.txt
- clientrawdaily.txt

Die clientraw.txt Datei enthält die aktuellen Wetterdaten der Station. Der Intervall der Aktualisierung dieser Datei wird in der Datei wdlconfig.xml eingestellt. Im Fall von Arbon wird die Datei alle 5 Sekunden aktualisiert. Hier werden auch die restlichen Parameter bzw. Einheiten in der die Daten gespeichert werden sollen eingestellt. Die clientrawextra.txt enthält die historischen Extremwerte. Die Datei clientrawhour.txt enthält die aufgezeichneten Daten der letzten Stunde im Minutentakt. [6]

2.2 Datenmanagement und Datensicherung

Täglich fallen 93600 Datenpunkte an, diese Daten werden alle seit 2015 gespeichert und nicht ausgedünnt. Das Problem hierbei ist folgendes schaut man die Datenbanken zum jetzigen Zeitpunkt an, scheint es chaotisch zu sein. Im Grunde werden nur die igwetter openfile64Light und die igwetter test Datenbank benutzt. Da es aber mehrere Datenbanken gibt, ist es auf dem ersten Blick nicht sichtbar, was wo gemacht wird und welche Datenbank für wofür zuständig ist. Ein weiterer Punkt sind die riesigen Mengen an Daten, diese verzögert eine Abfrage in der Datenbank enorm. Des weiteren wird nicht nur eine Datenbank sondern auch .txt Files benutzt um ein Backup zu erstellen, sowie die aktuellen Daten zu speichern. Die historischen Daten in der Datenbank sind nicht vollständig, zum einen gibt es für den gesagten Zeitraum zwischen 2012 und 2015 keine Daten und zum anderen werden die seit dem erstellten Daten nicht auch in die "historische" Tabelle tblgestern abgelegt.

Das zeigt auch auf wie wichtig eine gute Strukturierung der Datenbank ist. Hierfür wird vorgeschlagen die Datenbank igwetter openfile64Light so zu belassen, da diese für das CMS zuständig ist. Es wird eine neue Datenbank erstellt welche klarer strukturiert wird, hierbei sollten neue Tabellen entstehen, wobei Daten ausgedünnt werden können um keinen unnötigen Speicherplatz zu belasten. Die Haupttabelle soll alle aktuelle Daten enthalten. Zusätzlich soll Tabelle erstellt werden mit zukünftigen und vorhandenen historischen Daten. Um die Übersichtlichkeit zu gewähren wird eine Tabelle mit Maximal sowie Minimal Daten erstellt. Die Zeitabstände, in der die Daten gelöscht bzw. zu historischen Daten werden, müssen noch mit den Mitgliedern der IG-Wetter abgesprochen werden. Ein weiterer Punkt auf der Liste sollten die zukünftigen Backups sein, d.h. diese sollten nicht als .txt sonder auch als .sql File gespeichert sein damit im Falle eines Datenverlustes die Datenbank einfach wiederherzustellen ist.

2.3 API

REST-Service?

API für Badi (Luft- und Wassertemperatur)

Das Kapitel API wurde aus Zeitgründen noch nicht erstellt. Dies wird bis zu definitiven Abgabe noch nachgeholt.

3 Sensoren

Die Wetterstation Arbon verfügt über fünf Sensoren bzw. Sensor-Einheiten: Webcam, Kombi-Wetter-Transmitter, Wassertemperatur-Sensor und Pegelsensor. Auf der Plattform im See draussen befindet sich lediglich ein Schaltschrank mit Datenwandlern und keine Auswerteeinheit. Sämtliche Daten werden per TCP/IP an den Server geschickt. Abbildung 8 zeigt den schematischen Aufbau der Komponenten im Schaltschrank und die angeschlossenen Sensoren. Die Stromversorgung ist der Übersicht halber nicht dargestellt.

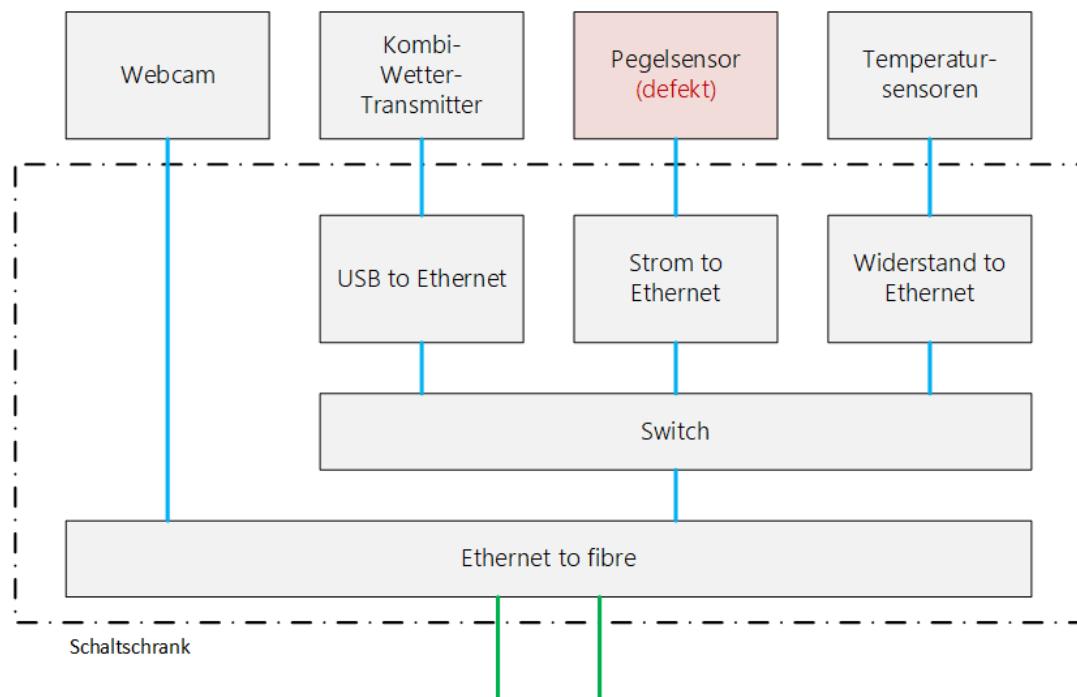


Abbildung 8: Hardware-Aufbau der Wetterstation Arbon

3.1 Pegelmesser

Der bisherige Pegel-Sensor nutzte das Prinzip der hydrostatischen Druckmessung. Der Sensor ist nun aber defekt und muss ersetzt werden. Neben der hydrostatischen Druck-

messung kommen weitere potentielle Messprinzipien in Frage. Sie alle erfüllen die Grundsatzforderung bezüglich Messdistanz und Robustheit. Während der Bachelor-Arbeit wird der passende Pegelsensor getestet und ausgewählt. Möglich ist:

- Hydrostatische Druckmessung
- Ultraschall-Distanzmessung
- Radar-Distanzmessung
- Time-of-flight-Distanzmessung

3.2 Wassertemperatursensoren

Die Wassertemperatur wird definitionsgemäß einen Meter unterhalb der Wasseroberfläche gemessen. Die Wetterstation Arbon verwendet eine Widerstandskaskade aus PT100-Widerständen. Diese sind in einem Kunststoffrohr im Abstand von 20cm angeordnet. Abhängig vom gemessenen Pegel kann so der richtige Temperatursensor für die Wassertemperatur ausgewählt werden. Von den zehn verbauten Sensoren ist einer defekt. Da die Reparatur allerdings sehr aufwändig ist, und der Wert durch die beiden Nachbarwiderstände interpoliert werden kann, wird der Widerstand nicht ersetzt. Für uns besteht diesbezüglich kein Handlungsbedarf.

4 Webcam-Steuerung

Zur Wetterstation Arbon gehört eine schwenk- und zoombare Webcam. Diese ist über ein Applikations-Plugin in die Webseite integriert. In der Titelleiste sind die wichtigsten Wetterdaten aufgeführt, wobei die Einheit der Windgeschwindigkeit jeweils alle dreissig Sekunden zwischen km/h und Knoten wechselt. Die Webseite der Webcam ist in Abbildung 9 links dargestellt.

Der Benutzer hat die Möglichkeit die Webcam nach oben/unten und nach links/rechts zu bewegen. Sechs voreingestellte Positionen stehen als Shortlinks zur Verfügung. Diese Positionen sind in der Betriebseigenen Software konfiguriert. Die freie Positionierung erfolgt über Pfeile, sowie die Plus und Minus am Bildrand, je nachdem welcher Button geklickt wird, sendet die Webseite das Kommando mittels HTTP an die Webcam. Die Zusammenstellung der URL geschieht über ein Javascript, wie man dem Code in Listing 2 entnehmen kann.

Listing 2: Positionsänderung der Webcam

```
1 <div class="container webcam" id="webcam_585">
2   
3 </div>
4 <script type="text/javascript">
5   function changeWebCam(command) {
6     var urlAddition;
7     switch (command) {
8       case 'up':
```

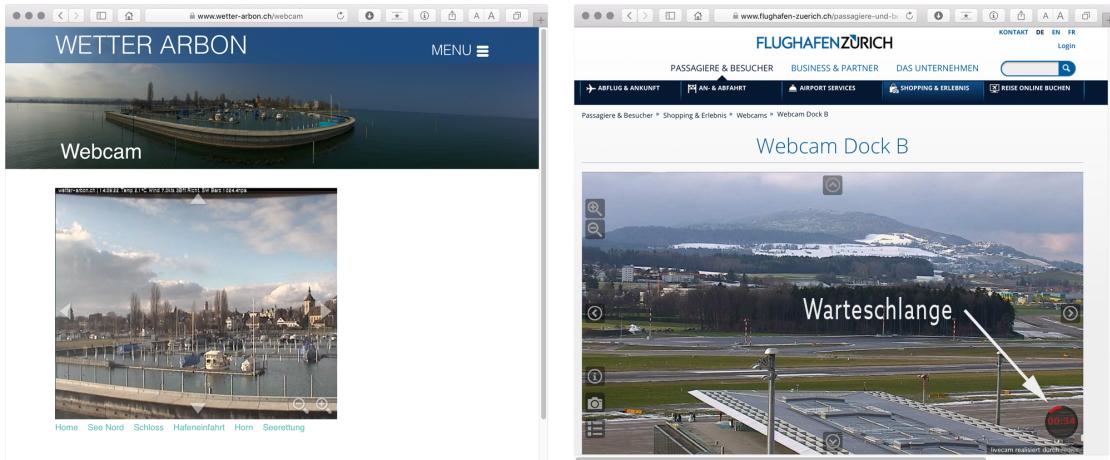


Abbildung 9: Webcam Arbon und Beispiel einer Warteschlange

```

9      case 'down':
10     case 'left':
11     case 'right':
12     case 'home':
13       urlAddition = 'move=' + command;
14       break;

16     case 'zoomIn':
17       urlAddition = 'rzoom=2500';
18       break;

20     case 'zoomOut':
21       urlAddition = 'rzoom=-2500';
22       break;

24     case 'Hafeneinfahrt':
25       urlAddition = 'gotoserverpresetname=' + command;
26       break;
27   }
28   console.log('changeWebCam');
29   $.get('https://webcam.wetter-arbon.ch/axis-cgi/com/ptz.cgi?camera=1&' + urlAddition);
30 }
```

4.1 Positionsabhängige Zoombeschränkung

In der betriebseigenen Software lassen sich viele Parameter konfigurieren, unter anderem der Zoomfaktor. Das Problem ist hier, dass der Zoomfaktor in der betriebseigenen Software nur allgemein eingestellt werden kann, das heisst die Beschränkung gilt immer, egal ob die Kamera auf eine Wohnung zeigt oder auf den See raus. Aus Perönlichkeitsschutz-Gründen musste deshalb der Zoomfaktor auf die 4-fache Vergrösserung limitiert werden, möglich wäre aber eine 216-fache Vergrösserung. Daraus wird deutlich, dass die Webcam eigentlich viel mehr Potential hat. Die Idee ist nun die Limitierung des Zooms so zu konfiguriert, dass diese möglichst dynamisch ist. Das heisst, dass je nach Position die

Zoom-Limitierung ändert. Der Zoom soll, vor allem auf den See hinaus, in vollem Umfang genutzt werden können. Die Ausrichtung der Kamera kann mit einer einfachen HTTP GET Anfrage abgerufen werden. So kann mittels Javascript der Zoom beschränkt oder erweitert werden

4.2 Warteschlange für Webcam-Steuerung

Die Möglichkeit der Webcam-Steuerung übers Web ist zwar sehr attraktiv, hat aber auch seine Nachteile. Zum Beispiel wenn mehrere Personen gleichzeitig auf die Webcam zugreifen. Zur Zeit ist es so dass, die HTTP-Request der Reihe nach abgearbeitet werden. Es ist also möglich, dass sich die Benutzer gegenseitig in der Bedienung stören, was unter Umständen recht mühsam ist. Das Prinzip der Warteschlange kann hier Abhilfe schaffen. Jeder Benutzer erhält eine besitzmte Zeit den alleinigen Zugriff auf die Steuerung der Webcam. So eine Lösung setzt zum Beispiel der Flughafen Zürich ein, wie in Abbildung 9 rechts dargestellt.

5 Erweiterungen

In diesem Kapitel geht es nicht um die Verbesserung von bestehenden Problemen, sondern um die Erweiterung des Funktionsumfangs der Wetterstation Arbon. Es ist eine Auflistung möglicher Erweiterungen, die allerdings keine Priorität haben.

5.1 Individueller Benachrichtigungs-Service

Mit einem Benachrichtigungs-Service soll dem Benutzer die Möglichkeit gegeben werden, dass er zeitnah über Wetteränderungen informiert wird und somit keine Warnung oder sein perfektes Segelwetter verpasst. Dafür wurden drei verschiedene Möglichkeiten ausgewählt und mit der Nutzwertanalyse ausgewertet. Ziel bei allen Möglichkeiten ist es, dass der Benutzer die Möglichkeit hat Alarmkriterien selbst zu bestimmen. Werden die gewählten Alarmkriterien erreicht bzw. wird eine Sturmwarnung herausgegeben, wird der Benutzer benachrichtigt. Für die Evaluierung der Notifications wurde eine Nutzwertanalyse (Tabelle 1) erstellt. Dies ist eine gute Möglichkeit, um verschiedene Lösungsansätze zu bewerten. Der Nachteil hierbei ist jedoch, dass die Bewertung sehr subjektiv ist. Aus der Nutzwertanalyse geht hervor, dass die Benachrichtigung per E-Mail und Facebook Messenger die Lösungsansätze mit der höchsten Punktzahl sind. Der grösste Vorteil der beiden möglichen Lösungen sind, dass sie kostenlos sind. Der Nachteil an Facebook Messenger ist, dass nicht davon ausgegangen werden kann, dass jeder Benutzer ein Facebookprofil hat.

Nutzwertanalyse					
Möglichkeiten	Kosten	Einfachheit	Aufwand	Anpassbarkeit	Support
SMS	1	4	3	3	5
E-Mail	5	4	5	5	1
FacebookMessenger	5	4	3	4	1

Tabelle 1: Nutzwertanalyse verschiedner Notifikations-Möglichkeiten

5.2 Überprüfung der Windprognose-Genauigkeit

Es gibt diverse Anbieter von Windprognosen für den Bodensee wie zum Beispiel Windfinder⁴ und SRF Meteo⁵. Vorhersagen sind bekanntlich Hochrechnungen und mit gewissen Unsicherheiten behaftet. Interessant ist nun zu wissen wie gut die Windvorhersagen mit den Wind-Messwerten der Wetterstation Arbon übereinstimmen. Während der Bachelorarbeit soll nun eine Vergleichsgrafik erstellt werden, welche die Vorhersage den Messwerten gegenüber stellt.

5.3 Berechnung und Darstellung der Wellenhöhe

Sobald ein funktionstüchtiger Pegelsensor installiert ist, können die Pegeldaten auch für andere Zwecke verwendet werden, zum Beispiel zur Berechnung der Wellenhöhe. Dies ist insbesondere für Motorboot-Fahrer interessant.

5.4 Verlauf der Wassertemperatur in Abhängigkeit der Tiefe

Die Wetterstation Arbon verfügt über mehrere Temperatursensoren, die im Abstand von 20 Zentimeter die Wassertemperatur messen. Die Idee ist, die Temperaturschichtung des Wassers bestimmen zu können.

6 Anforderungen

Die im folgenden aufgelisteten Anforderungen sind in fünf Blöcke unterteilt: User Interface, Sensoren, Webcam und nicht-funktionale Anforderungen. Jede Anforderung besitzt eine eindeutige Identifizierungsnummer, Titel, Beschreibung der Anforderung, Wichtigkeit und einen Beschrieb wie der Nachweis erfolgen soll. Die Wichtigkeit ist MUSS, SOLL oder KANN. MUSS-Anforderungen sind absolut zwingend für die Umsetzung der Arbeit. SOLL-Anforderungen bringen einen erheblichen Mehrwert und KANN-Anforderungen sind eher unwichtig und können gegebenenfalls auch weggelassen werden.

⁴ <https://www.windfinder.com/forecast/arbon>

⁵ <https://www.srf.ch/meteo/surf-und-segelwetter/detail/06621>

6.1 User Interface (UI)

UI 010	Flash-less Webseite (FA)
Anforderung	Sämtliche Webseiten der Wetterstation Arbon funktionieren ohne direkte bzw. indirekte Verwendung von Adobe Flash.
Wichtigkeit	MUSS
Test	Sämtliche Webseiten der Wetterstation Arbon können von folgenden Browsern angezeigt werden, ohne dass Adobe Flash aktiviert bzw. installiert ist: Safari (Mobile & Desktop), Google Chrome (Mobile & Desktop), Firefox, Edge und Internet Explorer.
UI 020	Einheiten
Anforderung	Für die Anzeige der Messwerte werden folgende Einheiten verwendet: Temperatur in C, Luftdruck in hPa, Windrichtung mindestens in Grad, Niederschlagsmenge in mm, Relative Luftfeuchtigkeit in %
Wichtigkeit	MUSS
Test	Die Messwerte werden in C, hPa, Grad, mm und % angezeigt.
UI 030	Wetterdaten für Wassersportler
Anforderung	Die Anzeige der Wettertransmitterdaten erfolgt in nautischen Einheiten d.h. die Windgeschwindigkeit wird in Knoten angegeben und parallel dazu in Beaufort. Graphen zeigen den Verlauf von Luftdruck, Windgeschwindigkeit und Windrichtung der letzten 24h auf.
Wichtigkeit	MUSS
Test	Die aktuelle Windgeschwindigkeit wird auf der Wassersport-Seite in Knoten und Beaufort angegeben. Die x-Achse der Graphen zeigt die letzten 24h.
UI 040	Wetterdaten für Tourismus
Anforderung	Die Anzeige der Wettertransmitterdaten erfolgt in allgemein verständlichen Einheiten d.h. die Windgeschwindigkeit wird in km/h angeben. Graphen zeigen den Verlauf von Temperatur, Niederschlag und Windchill der letzten sieben Tage auf.
Wichtigkeit	MUSS
Test	Die Windgeschwindigkeit wird in km/h angezeigt. Die x-Achse der Graphen zeigt die letzten sieben Tage.
UI 050	Responsive Design
Anforderung	Die Werte der Wetterstation sind unabhängig von der Bildschirmgrösse übersichtlich und lesbar dargestellt. Horizontales Scrollen ist nicht erforderlich.
Wichtigkeit	SOLL
Test	Die Webseite der Wetterdaten wird mit einem iPhone 5, iPad und Desktop so dargestellt, dass kein horizontaler Scrollbalken auftritt.

UI 060	Samplerate
Anforderung	Die Sample-Rate der Graphen ist kleiner gleich zehn Minuten auf der Wassersport-Seite und kleiner gleich eine Stunde auf der Tourismus-Seite.
Wichtigkeit	SOLL
Test	Pro Graph sind für die Wassersport-Seite mindestens $24*6=144$ Punkte eingezeichnet, für die Tourismus-Seite mindestens $7*24=168$ Werte.
UI 070	Fixe Y-Achse
Anforderung	Für die Graphen auf der Tourismus und Wassersport-Seite wird eine fixe Y-Achs-Skalierung verwendet.
Wichtigkeit	SOLL
Test	Unabhängig von den Messwerten ist die Skalierung der y-Achse sämtlicher Graphen auf der Tourismus- und Wassersport-Seite konstant.
UI 080	Anzeige Windrichtung
Anforderung	Die Anzeige der Windrichtung kann sich kontinuierlich ändern, ohne dass in der Momentananzeige bzw. im Graphen ein Sprung entsteht.
Wichtigkeit	SOLL
Test	Wenn sich der Wind einem um 360 Grad dreht ist auf der Anzeige und im Graphen keine Sprung erkennbar.
UI 090	Barrierefreiheit
Anforderung	Die Anzeige der Wetterstation soll sowohl mit rot/grün Sehschwäche, als auch für sehbehinderte Menschen verständlich sein.
Wichtigkeit	SOLL
Test	Die Seiten werden mit einem Online-Color-Checker und einem Screen-Reader auf deren Verständlichkeit überprüft
UI 100	Notification
Anforderung	Der User kann sich selbst eine Notifikation einrichten. Er erhält eine Nachricht sobald der von ihm definierte Wert der Wetterstation unter- bzw. überschritten wird
Wichtigkeit	KANN
Test	Der User richtet sich eine Notification ein für Wassertemperatur grösser als 20 Grad und Windgeschwindigkeit grösser 10 Knoten und erhält für jede Anweisung eine separate Nachricht, sobald diese erfüllt ist.
UI 110	Favicon
Anforderung	Wenn die Webseite auf dem Homescreen eines Mobilgerätes abgespeichert wird, ist das Favicon der Wetterstation Arbon abgebildet.
Wichtigkeit	KANN
Test	Der User speichert die Webseite auf einem iPhone und sieht das Wetterstation Arbon Favicon

6.2 Datenbank (DB)

DB 010	Abfrage-Seite
Anforderung	Auf der Webseite der Wetterstation Arbon gibt es eine eigene Seite, auf der vom User Datenbank-Abfragen ausgeführt werden können. Das Resultat wird jeweils graphisch dargestellt. Die Abfragen können auf sämtliche Messwerte der Wetterstation und über den Zeitraum seit Datenerfassung durchgeführt werden. Liegen für einen bestimmten Zeitraum keine Messwerte vor, werden keine Werte angezeigt, und es findet auch keine Interpolation statt.
Wichtigkeit	MUSS
Test	Der User macht zwei Abfragen: In der ersten Abfrage soll die Windgeschwindigkeit in km/h seit Messbeginn aufgezeichnet werden. In der zweiten Abfrage soll der Pegel im ersten Betriebsjahr aufgezeichnet werden. Die Werte werden korrekt in der Grafik abgebildet inkl. Messlücke.
DB 020	Schutz vor Missbrauch
Anforderung	Die Schnittstelle zur Datenbank d.h. die Datenbank-Abfrage ist gegen schädliche Zugriffe geschützt.
Wichtigkeit	MUSS
Test	Die Abfrage des Pegels seit Inbetriebnahme der Wetterstation wird als File exportiert und kann anschliessend in Excel geöffnet werden.
DB 030	Fehleingaben
Anforderung	Die Abfrage-Seite ist so ausgeführt, dass sie Fehleingaben verunmöglicht.
Wichtigkeit	MUSS
Test	Der User versucht eine unplausible Abfrage zu senden.
DB 040	Datenmanagement
Anforderung	Die Daten in der Datenbank werden periodisch ausgedünnt d.h. zusammengefasst.
Wichtigkeit	SOLL
Test	Messwerte, die älter als eine Woche sind, werden zusammengefasst zu maximal einem Wert pro Stunde.
DB 050	Daten-Export
Anforderung	Die Resultate der getätigten Abfragen können als Datei exportiert werden.
Wichtigkeit	KANN
Test	

6.3 Sensoren (TD)

TD 010	Pegel-Messer
Anforderung	Die Wetterstation Arbon erhält einen geeigneten Pegel-Sensor, welcher mit den bereits verbauten Komponenten betrieben werden kann. Die Kosten für Anschaffung und Betrieb des neuen Sensors lieben innerhalb des Budgets der Wetterstation Arbon.
Wichtigkeit	MUSS
Test	Der Pegel Arbon wird auf der Station gemessen und ist auf der Webseite ersichtlich.
TD 020	Schnittstelle zur Wassertemperatur
Anforderung	Die Wetterstation Arbon verfügt über eine öffentlich nutzbar Schnittstelle (API) für die Wassertemperatur, sodass z.B. das Seebad diese übernehmen kann.
Wichtigkeit	MUSS
Test	Die Wassertemperatur kann über ein API abgerufen werden.
TD 030	Sturmwarnung
Anforderung	Auf der Webseite wird die aktuelle Sturmwarn-Situation dargestellt. Falls es sich um eine eingebettete Seite handelt, muss diese https-fähig sein.
Wichtigkeit	SOLL
Test	Die aktuelle Sturmwarn-Situation ist für den User auf der Webseite der Wetterstation Arbon einsehbar, ohne Verlinkung auf fremde Seiten.
TD 040	Vergleich Windvorhersage und Windmessresultate
Anforderung	Auf der Webseite ist grafisch ersichtlich wie die Windgeschwindigkeits-Vorhersage und die gemessene Windgeschwindigkeit zueinander stehen.
Wichtigkeit	SOLL
Test	Die vorhergesagten und die gemessenen Windgeschwindigkeiten der letzten sieben Tage sind grafisch dargestellt.
TD 050	Wellenhöhen
Anforderung	Aus den Messwerten des Pegelsensors wird die durchschnittliche Wellenhöhe berechnet und angezeigt. Die Wellenhöhe wird in der Datenbank gespeichert.
Wichtigkeit	SOLL
Test	Der User sieht auf der Webseite die aktuelle Wellenhöhe. Der User führt eine Datenbankabfrage aus und sieht den Verlauf der Wellenhöhe über die letzten drei Monate.
TD 060	Strahlungssensor
Anforderung	Die Wetterstation Arbon erhält einen geeigneten Sonnenstrahlungs-Sensor. Die Kosten für Anschaffung und Betrieb des neuen Sensors lieben innerhalb des Budgets der Wetterstation Arbon.
Wichtigkeit	KANN
Test	Die Sonnenstrahlung wird auf der Station gemessen und ist auf der Webseite ersichtlich.

6.4 Webcam (CA)

CA 010	Warteschlange
Anforderung	Die Webcam verfügt über eine Warteschlange, sodass wenn mehrere User auf der Seite sind, jeder die Steuerung der Webcam für eine gewisse Zeit für sich alleine hat.
Wichtigkeit	SOLL
Test	Zwei User greifen zur gleichen Zeit auf die Webcam zu. Der zweite erhält die Steuerung, sobald die Zeit des ersten Users abgelaufen ist.
CA 020	Sekotrweise Zoombeschränkung
Anforderung	Die Zoomstufe ist abhängig von der Ausrichtung der Webcam. Richtung Land ist der Zoom beschränkt, Richtung See offen.
Wichtigkeit	KANN
Test	Der User zoomt einmal maximal Richtung Land und einmal maximal Richtung See. Die Zoomstufe Richtung Land ist kleiner, als Richtung See.

6.5 Nicht Funktionale Anforderungen (NF)

NF 010	Reaktionsgeschwindigkeit
Anforderung	Bei ausreichendem Netz werden die Messdaten innerhalb von drei Sekunden angezeigt. Datenbank-Abfragen werden innerhalb von fünf Sekunden angezeigt.
Wichtigkeit	SOLL
Test	Der User ruft die Seite auf und sieht die Messresultate innerhalb von drei Sekunden. Der User wählt die Wassertemperatur der letzten zwölf Monate und erhält fünf Sekunden nach absenden der Abfrage das Resultat.

7 Projektmanagement

Wir wollen das Projektmanagement schlank halten um möglichst viel Zeit in die Entwicklung der Artefakte stecken zu können. Dieser Grundgedanke hat uns bei der im Folgenden beschrieben Auswahl der Modelle und Prozesse geleitet.

7.1 Vorgehensmodell

Die Anforderungen an das Vorgehensmodell haben wir folgendermassen definiert:

- wenig administrativer Aufwand, schlank
- passend zur Projektgrösse
- kompatibel mit den NTB-Vorgaben (Aufteilung Fachmodul, Bachelor-Arbeit)

Schnell merkten wir, dass die heutzutage beliebten agilen Vorgehensmodelle wie XP oder Scrum für uns ein Overkill darstellen und aus mehrerer Hinsicht nicht geeignet sind. Bei der Bachelor-Arbeit sind die Anforderungen im Fachmodul-Bericht definiert

und ändern sich während der Bachelor-Arbeit nicht mehr. Die zu bearbeitenden Themen-Blöcke weisen untereinander nur sehr wenige Schnittstellen auf und können dadurch als eigenständige Teilprojekte das Modell durchlaufen. Unser Team besteht zudem nur aus zwei Personen, was den Koordinationsaufwand auf ein Minimum reduziert.

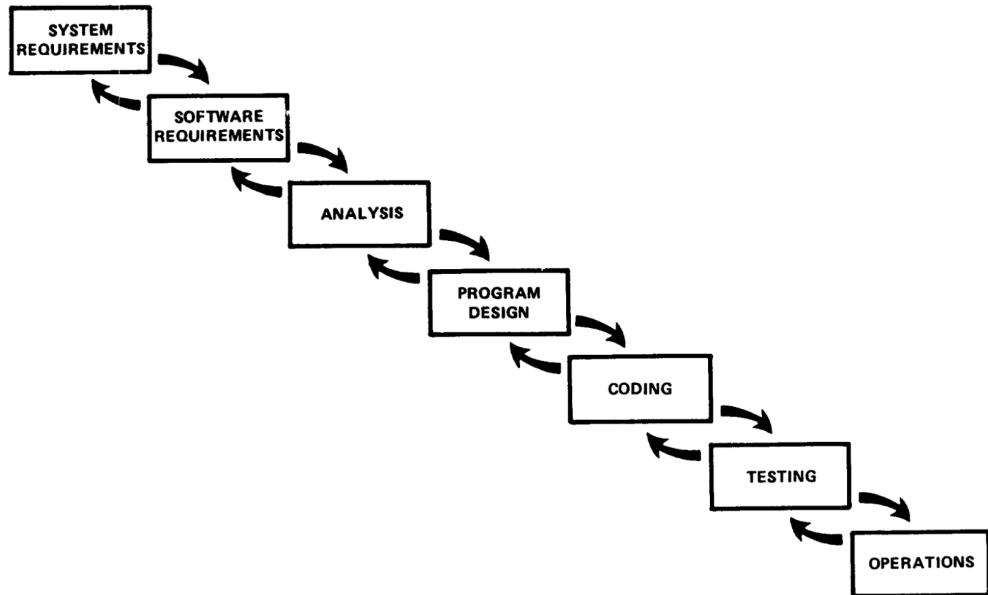


Abbildung 10: Vorgehensmodell nach Royce

Unsere Bedürfnisse deckt das Vorgehensmodell von Royce [7], welches in Abbildung 10 dargestellt ist, am besten ab. Es besteht grundsätzlich aus einem sequentiellen Ablauf der Entwicklungsphasen, berücksichtigt dabei aber auch die Notwendigkeit von Rücksprünge zur vorherigen Phase. Die ersten Phasen von der Definition der System Requirements bis zu den ersten Gedanken zum Thema "Program Design" behandeln wir im Fachmodul. Der zweite Teil mit der genauen Definition des Programm Designs bis zum Betrieb der Software findet anschliessend während der Bachelor-Arbeitszeit statt.

7.2 Entwicklungsprozess

Den Entwicklungsprozess führen wir mit Kanban. Kanban basiert auf dem Pull-Prinzip d.h. jeder, der im Projekt arbeitet, holt sich selbst einen neuen Arbeitsauftrag, sobald er mit einem fertig ist. Dies führt dazu, dass die Arbeiten speditiver abgewickelt werden und spart zudem die Stelle des Projektmanagers, der die Aufgaben verteilt.

David Anderson [2] hat das System Kanban, welches ursprünglich aus der Industrie kommt, auf die IT angepasst und dadurch das "Virtuelle Kanban System" entwickelt. Die grundlegenden Regeln daraus lauten:

- Jede Karte ist eine Aufgabe

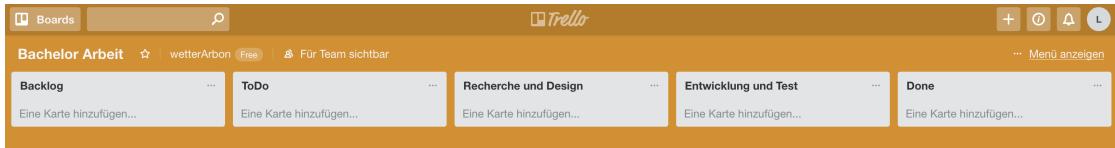


Abbildung 11: Kanban

- Die Aufgabe soll maximal 8-16h benötigen
- Pro Spalte sind die Anzahl Karten limitiert
- Eine neue Karte darf erst gezogen werden, wenn die vorherige fertig ist (Multitasking-Vermeidung)

7.3 Projektplan für die Bachelor-Arbeit

Der Zeitplan für die Bachelor-Arbeit ist in Abbildung 12 auf Seite 21 dargestellt. Im oberen Teil sind die allgemeinen Termine und Abwesenheiten aufgeführt. Der mittlere Teil zeigt die Arbeitsverteilung über das Semester und am Schluss kommen die Zeitaufwände für Doku und Meetings. Die Dokumentation wollen wir kontinuierlich erstellen, sodass wöchentlich ein entsprechender Block vorgesehen ist.

7.4 Dokumentation

Für die Bachelor-Arbeit verwenden wir unterschiedliche Dokumentationswerkzeuge. Bei der Auswahl haben wir darauf geachtet, dass die Tools kostenlos nutzbar und für sämtliche Plattformen verfügbar sind (Windows, Mac, iPad, usw.) Weiter war uns wichtig, dass die Tools untereinander kommunizieren können.

Versionierung und Zeiterfassung

Sämtliche Artefakte speichern wir auf *github*. Wir haben somit eine automatische Versionierung der Dokumente und können unabhängig voneinander an den Dokumenten arbeiten. Die Planung bzw. Darstellung des Entwicklungsprozesses erledigen wir mit *Trello*. Es ist ein intuitives Tool, welche diverse Integrationsmöglichkeiten mit den anderen Tools bietet. Für die Zeiterfassung verwenden wir *Toggl*, welches mittels Plugin direkt in Trello integriert werden kann.

Reporting; Kommunikation extern

Damit wir keine Besprechungsprotokolle verschicken müssen und dass alle Informationen für alle immer zugänglich sind, haben wir entschieden das Reporting in Form einer öffentlichen Webseite zu erstellen. *github* bietet mit *GitPages* einen Hosting-Service an, der genau dies ermöglicht. Der Vorteil von GitPages ist, dass wir sämtliche Daten in einem einzigen Ort bzw. Repository vereint haben. Damit wir uns nicht mit Formatierung

Nr. Aufgabe	Dauer [h]	F	März					April			Mai					Juni			Juli			August			September			734					
		KW08	KW09	KW10	KW11	KW12	KW13	KW14	KW15	KW16	KW17	KW18	KW19	KW20	KW21	KW22	KW23	KW24	KW25	KW26	KW27	KW28	KW29	KW30	KW31	KW32	KW33	KW34	KW35	KW36	KW37	KW38	KW39
1 Diverses																																	
Unterrichtsfrei																																	
Modulschlussprüfungen																																	
Ferien Ulrich																																	
Ferien Thomas																																	
Ferien Ladina																																	
2 Meilesteine																																	
Pre-Schlussbericht																																	
Abgabe BA																																	
Präsentation																																	
3 Auftrag gem. Mindmap																																	
Datenbank	120	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	12	108	
Pegelmessung	80	12	12																														80
Dashboard Design	60																																60
Dashboard Grafik	60																																72
Analyse Wetterprogose	120																															120	
Webcam	40																																36
Wassertemperatur	40																																36
Notifikation	40																																36
4 Dokumentation																																	114
Wochenreports	50	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	48	
Pre-Schlussbericht	30																																16
Schlussbericht	20																																20
Präsentation	30																																30
5 Meetings																																	72
Wöchentlich (intern)	50	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	52		
Mit Betreuer	12	2	2	2	2	2	2																									18	
BA-Präsentation	2																																2

Abbildung 12: Projektplan

u.a. herumschlagen müssen und uns auf den Inhalt konzentrieren können, verwenden wir *mkdocs* als Template Engine. Die Webseiten-Einträge können wir dadurch auf simple Art in Form von Markdown-Files erstellen.

Kommunikation teamintern

Innerhalb des Teams nutzen wir das Kommunikationstool *Slack*. Dieses ermöglicht uns Konversationen als Chat aufzuzeichnen und nach Themen zu gruppieren. Weiter lassen sich Dokumente austauschen. Sämtliche git-Posts werden von Slack automatisch geloggt und können, falls gewünscht, als push-Notification angezeigt werden. Das wöchentliche Team-Meeting findet über *Skype* statt, da wir den regelmässigen mündlichen Austausch aus zentralen Punkt erachten.

Dokumentation

Den Bericht werden wir in LaTeX verfassen. Wir haben uns für LaTeX entschieden, da wir uns auf den Inhalt konzentrieren können und das Layout automatisiert ist. Weiter ist LaTeX in der Wissenschaft weit verbreitet. Die Bachelor-Arbeit ist deshalb eine gute Gelegenheit uns in dieses Thema einzuarbeiten.

7.5 Risikoanalyse

Für die Risikoanalyse haben wir eine Liste der möglichen Risiken erstellt. Als Grundlage verwendeten wir das Risikolexikon aus dem Buch «IT-Risikomanagement leben!» [1]. Für jedes Risiko haben wir die Eintretenswahrscheinlichkeit und das Ausmass abgeschätzt. Gegenüber den herkömmlichen Risikobeurteilungen, haben wir allerdings die Auswirkungen auf Kosten und Terminverzug weggelassen, da sie in unserem Projekt nicht relevant sind und uns auf den Stundenaufwand und den Funktionsumfang beschränkt. Um die Auswirkung der einzelnen Risiken abschätzen zu können, haben wir eine Punkteskala mit entsprechenden Kriterien erstellt, wie in Tabelle 7.5 aufgeführt. Die komplette Risikoliste14 befindet sich auf Seite 24.

Wert [-]	Auswirkung bezüglich Umfang
10	Gesamter Block nicht funktionsfähig
8	Einzelne Funktion nicht umgesetzt
6	Bemerkbar, keine Funktionseinbusse
4	von eingeschränkter Benutzergruppe bemerkbar
2	von Kunden nicht bemerkbar

Die Risikomatrix zeigt auf grafische Weise wie kritisch die einzelnen Risiken aus der Risikoliste13 sind. Mindestens vier davon sind als hoch eingestuft und müssen im Rahmen der Bachelor-Arbeit reduziert werden.

8 Rechtliche Ansprüche

siehe separates Dokument

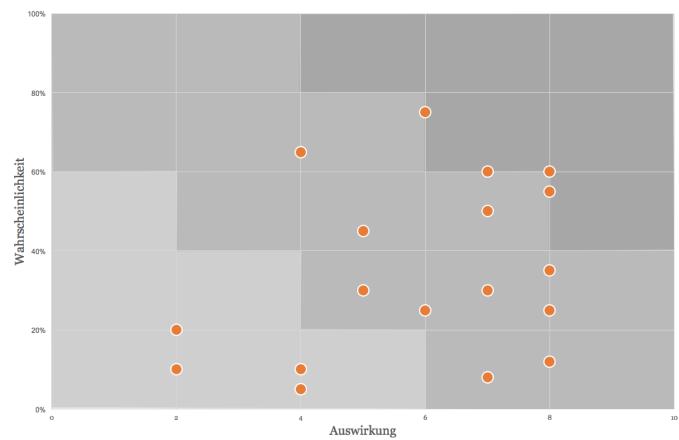


Abbildung 13: Risikomatrix

Risikoanalyse				Action Plan			
#	Risikoname / Risikosatz	Wahrscheinlichkeit	Auswirkung	Schadensausmass	Strategie	Status	Erläuterungen
001	Unklare Zielvorstellung	50%	7	3.5	Vermindern	in Arbeit	
002	Mangelnde Gebrauchstauglichkeit	30%	5	1.5	Vermindern	in Arbeit	
003	Mangelnde Anforderungsqualität	60%	7	4.2	Vermindern	in Arbeit	
004	Steigende Anforderungsumfang	8%	7	0.6	Vermeiden	in Arbeit	
005	Verändernde Anforderungen	12%	8	1.0	Vermindern	in Arbeit	
006	Einsatz neuer Technologien	45%	5	2.3	Akzeptieren	in Arbeit	
007	Entwicklung unnötiger Funktionen	5%	4	0.2	Vermeiden	in Arbeit	
008	Mangelhafte Schnittstelle zwischen Anwendungen	35%	8	2.8	Vermindern	in Arbeit	
009	Entwickler vernachlässigen Test	30%	7	2.1	Vermindern	in Arbeit	
010	Komplexe Datenmigration	55%	8	4.4	Akzeptieren	in Arbeit	
011	Mängel an Echtzeitverhalten	75%	6	4.5	Vermindern	in Arbeit	
012	Fehlende Benutzerakzeptanz	10%	4	0.4	Vermindern	in Arbeit	
013	Mangeldne Definition des Abnahmeverfahrens	25%	6	1.5	Vermindern	in Arbeit	
014	Unrealistische Zeitplnaung	65%	4	2.6	Vermindern	in Arbeit	
015	Mitarbeiter Ausfall	25%	8	2.0	Akzeptieren	in Arbeit	
016	Mangelnde Ressourcenverfügbarkeit	60%	8	4.8	Vermindern	in Arbeit	
017	Unzureichende Unterstützung durch Auftraggeber	20%	2	0.4	Vermindern	in Arbeit	
018	Mangelndes Temawork	10%	2	0.2	Vermindern	in Arbeit	

Abbildung 14: Risikoliste: Ausgangssituation

9 Schluss

Erkenntnisse, Einschätzungen
Vergleich FM-Planung zu FM-IST

Das Kapitel Schluss wurde aus Zeitgründen noch nicht erstellt. Dies wird bis zu definitiven Abgabe noch nachgeholt.

Literatur

- [1] Fabian Ahrendts. *IT-Risikomanagement leben! : wirkungsvolle Umsetzung für Projekte in der Softwareentwicklung*. Xpert.press. Springer, Berlin, 2008.
- [2] David J Anderson. *Kanban : evolutionäres Change Management für IT-Organisationen*. dpunkt, Heidelberg, 2011.
- [3] Richard Barnes. Deprecating non-secure http. <https://blog.mozilla.org/security/2015/04/30/deprecating-non-secure-http/>, 2015. [Online; abgerufen am 1. Dezember 2017].
- [4] Adobe Corporate Communications. Flash and the future of interactive content. <https://theblog.adobe.com/adobe-flash-update/>, 2017. [Online; abgerufen am 29. November 2017].
- [5] Steve Jobs. Thoughts on flash. <https://www.apple.com/hotnews/thoughts-on-flash/>, 2009. [Online; abgerufen am 29. November 2017].
- [6] Weather display live konfiguration. http://www.weather-display.de/english/Konfiguration_wdl_v2_04.pdf. [Online; abgerufen am 24. November 2017].
- [7] Winston W. Royce. Managing the development of large software systems. *Proceedings, IEEE Wescon*, pages 1–9, 1970.