

# Auswertung und Kombination verschiedener Datenquellen und deren Visualisierungsmöglichkeiten an einer Medienfassade

Abschlussarbeit zur Erlangung des akademischen Grades  
Master of Science (M. Sc.)  
an der Hochschule für Technik und Wirtschaft Berlin

**Autor:** Robin Schlegel  
**Matrikelnummer:** 522402  
**Fachbereich:** Wirtschaftswissenschaften II  
**Studiengang:** Angewandte Informatik  
**1. Prüfer:** Prof. Dr. Jürgen Sieck  
**2. Prüfer:** Prof. Dr.-Ing. Thomas Jung

Berlin, den 1. Februar 2013

# Danksagung

An dieser Stelle möchte ich mich herzlich bei denjenigen Personen bedanken, die mich bei der Erstellung dieser Masterarbeit unterstützt haben.

Zunächst möchte ich Herrn Prof. Dr. Jürgen Sieck für die Betreuung des Themas und die Begleitung während der Erstellung danken; weiterhin meinem zweiten Betreuer Prof. Dr.-Ing. Thomas Jung für die hilfreichen Tipps und Prof. Dr.-Ing. Volker Quaschnig für die Hilfe bei Fragen bezüglich der Wetterstation.

Mein Dank gilt insbesondere meiner gesamten Familie für die Unterstützung; vor allem meiner Schwester Jessica Schlegel, die stets mit Verbesserungsvorschlägen zur Stelle war, meiner Mutter und meinem Vater, der die Abgabe dieser Arbeit leider nicht mehr miterleben kann. Weiterhin gilt mein Dank all meinen Freunden für die moralische Unterstützung, ebenso Anja Godolt für die unzähligen Korrekturvorschläge und nicht zuletzt im Besonderen Alexandra Schuchardt für die wichtige Unterstützung in der Zeit.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Hintergrund: Die Errichtung des Forschungs- und Weiterbildungszentrums für Kultur und Informatik . . . . .	1
1.2	Zielstellung . . . . .	1
1.3	Aufbau und Struktur . . . . .	2
<b>2</b>	<b>Grundlagen</b>	<b>4</b>
2.1	Visualisierung . . . . .	4
2.1.1	Das RGB-Farbmodell . . . . .	4
2.1.2	Visualisierung von Daten . . . . .	5
2.1.3	Technologien zur Visualisierung auf einer Medienfassade . . . . .	5
2.2	Auswertung von Daten . . . . .	7
2.2.1	Data Mining . . . . .	8
2.3	Meteorologie . . . . .	8
2.3.1	Meteorologische Größen . . . . .	8
2.3.2	Wetterprognose . . . . .	10
2.4	Astronomie . . . . .	10
2.4.1	Astronomische Koordinatensysteme . . . . .	11
2.4.2	Sonnenposition . . . . .	14
2.4.3	Globalstrahlung . . . . .	18
2.5	Photovoltaik . . . . .	22
2.5.1	Photovoltaik-Anlagen . . . . .	22
2.5.2	Berechnung der erzeugten Leistung . . . . .	23
<b>3</b>	<b>Analyse</b>	<b>26</b>
3.1	Aufbau und Lage des FKI-Gebäudes . . . . .	26
3.1.1	Aufbau der Medienfassade . . . . .	27
3.1.2	Aufbau der Photovoltaik-Anlage . . . . .	29

3.2	Aufbau der Wetterstation der HTW Berlin . . . . .	31
3.3	Anforderungsanalyse . . . . .	32
3.4	Analyse und Bewertung verschiedener Sensordaten . . . . .	33
3.4.1	Mögliche reale Sensoren . . . . .	33
3.4.2	Mögliche virtuelle Sensoren . . . . .	35
3.4.3	Analyse der Sensordaten auf Kombinationsmöglichkeiten . . . . .	35
3.5	Analyse und Bewertung der Prognose der Stromerzeugung . . . . .	36
3.5.1	Prognose der Globalstrahlung . . . . .	36
3.5.2	Prognose der Stromerzeugung . . . . .	38
3.6	Analyse der Visualisierungstechnologien einer Medienfassade . . . . .	39
3.6.1	Visualisierung auf einer Medienfassade . . . . .	39
3.6.2	Visualisierung auf der Medienfassade des FKI-Gebäudes . . . . .	39
<b>4</b>	<b>Entwurf</b>	<b>41</b>
4.1	Entwurf der Systemarchitektur . . . . .	41
4.2	Modellierung des Dashboards . . . . .	42
4.2.1	Entwurf der Oberfläche des Dashboards . . . . .	42
4.3	Entwurf der Widgets . . . . .	44
4.3.1	Widgets für die PV-Anlage . . . . .	44
4.3.2	Widgets für das Wetter . . . . .	46
4.3.3	Weitere Widgets . . . . .	49
4.3.4	Entwurf der Interaktion . . . . .	52
4.4	Algorithmus für die Prognose des erzeugten Solarstroms . . . . .	53
4.4.1	Berechnung der Globalstrahlung auf die PV-Anlage zu einem bestimmten Zeitpunkt . . . . .	53
4.4.2	Berechnung der Leistung der PV-Anlage zu einem bestimmten Zeitpunkt . . . . .	60
4.4.3	Maximale Leistung der PV-Anlage . . . . .	60
4.4.4	Möglichkeiten durch die Prognose der Stromerzeugung . . . . .	61
<b>5</b>	<b>Proof of Concept</b>	<b>63</b>
5.1	Widget-Klasse . . . . .	63
5.2	Widget-Launcher . . . . .	64
5.2.1	Warteschlange . . . . .	64
5.2.2	Screens . . . . .	66

5.3	Implementierte Widgets . . . . .	66
5.3.1	Widget für die Prognose des Stroms . . . . .	67
5.3.2	Interaktion mit dem Widget für die Berechnung der Sonnenposition . . . . .	68
5.4	Konfiguration der Anwendung . . . . .	69
5.5	Schnittstellen . . . . .	70
5.6	Komponenten . . . . .	70
5.7	Verwendete Bibliotheken . . . . .	71
5.7.1	Simple DirectMedia Layer . . . . .	71
5.7.2	Boost . . . . .	73
5.7.3	libcurl . . . . .	73
5.7.4	JSON Spirit . . . . .	73
5.7.5	SOCI . . . . .	74
5.7.6	MD5 . . . . .	74
5.7.7	jQuery mobile . . . . .	74
5.7.8	Node.js und socket.io . . . . .	75
5.8	Fehlerbehandlung . . . . .	75
<b>6</b>	<b>Tests</b>	<b>77</b>
6.1	Testumgebung . . . . .	77
6.1.1	Testlabor . . . . .	77
6.2	Durchgeführte Tests . . . . .	79
6.2.1	Modultests . . . . .	79
6.2.2	Integrations- und Systemtests . . . . .	80
6.2.3	Tests für den Algorithmus für die Berechnung der Globalstrahlung . . . . .	80
<b>7</b>	<b>Ergebnis</b>	<b>83</b>
7.1	Fazit . . . . .	83
7.2	Ausblick . . . . .	84
	<b>Abbildungsverzeichnis</b>	<b>I</b>
	<b>Literaturverzeichnis</b>	<b>III</b>
	<b>Glossar</b>	<b>IX</b>

# 1 Einleitung

## 1.1 Hintergrund: Die Errichtung des Forschungs- und Weiterbildungszentrums für Kultur und Informatik

Hintergrund der vorliegenden Arbeit ist die Errichtung des Forschungs- und Weiterbildungszentrums für Kultur und Informatik (FKI) auf dem Spreecampus der Hochschule für Technik und Wirtschaft Berlin (HTW Berlin). Der Bau des Gebäudes hat im Dezember 2011 begonnen und wird voraussichtlich im Februar 2013 abgeschlossen sein. Die Seiten des Gebäudes werden eine spezielle Fassade erhalten. Die Nordseite wird mit einer Medienfassade ausgestattet, welche über drei unterschiedliche Technologien für die Visualisierung verfügen wird: LED-Kacheln, Rückprojektion und Ambient-Light. Die übrigen drei Seiten des Gebäudes werden mit einer Photovoltaik-Anlage bestückt, welche zur Erzeugung von Solarstrom dient.

Seit Ende 2010 verfügt die HTW Berlin zudem über eine Wetterstation. Diese befindet sich ebenfalls auf dem Spreecampus und ist auf dem Dach des Gebäudes G angebracht. Die Wetterstation ist mit neun Sensoren ausgestattet, um die Messung verschiedener Wetterdaten zu ermöglichen.

## 1.2 Zielstellung

Ziel dieser Arbeit soll es sein, die Daten der Medienfassade, der Photovoltaik-Anlage und der Wetterstation auszuwerten, zu kombinieren als auch zu visualisieren. Für die Ergebnisse soll eine Informationstafel (*Dashboard*) für die Medienfassade entwickelt werden, bei der auch die Nutzung weiterer Datenquellen in Betracht gezogen werden soll. Besonders die Fragestellung, inwiefern sich eine Kombination der Daten als sinnvoll erweist, soll einen Schwerpunkt darstellen. Zusätzlich soll auch die Möglichkeit untersucht werden, ob Messwerte und Prognosen aus anderen Datenquellen, wie beispielsweise dem Internet, sich für eine weitere Optimierung als sinnvoll erweisen.

Beispielhaft lässt sich die Kombination der Daten am folgenden Szenario veranschaulichen: Aufbauend auf einer Wetterprognose könnte eine Prognose für den erzeugten Strom in den nächsten Stunden oder Tagen erstellt werden. Eventuell könnten, unter Verwendung weiterer Datenquellen, die Ergebnisse der Prognose weiter optimiert werden. Das Ergebnis der Auswertung ließe sich einerseits auf der Medienfassade visualisieren, andererseits für das Gebäude nutzen. In diesem Zusammenhang ist die Regulierung der Medienfassade bei zukünftigem diesigem Wetter denkbar, bei dem folglich weniger Strom erzeugt wird.

Die über die verschiedenen Sensoren gewonnenen Daten sollen für unterschiedliche Visualisierungen auf der Medienfassade eingesetzt werden. Dabei sollen die spezifischen Charakteristika einer Medienfassade berücksichtigt werden. Für eine mögliche Visualisierung würden zum Beispiel die Daten der Wetterstation oder der Photovoltaik-Anlage in Frage kommen. Weiterhin könnten zusätzlich die Abfahrtszeiten der Tram, Unwetterwarnungen, der Speiseplan der Mensa oder wichtige Termine angezeigt werden.

Zum Erreichen dieser Ziele sind die folgenden Fragestellungen von Bedeutung:

- Welche Datenquellen und Sensoren sind für das System sinnvoll?
- Welche Daten können in welcher Art und Weise vorteilhaft kombiniert werden und welcher Nutzen kann aus der Kombination gewonnen werden?
- Wie lassen sich die Datenmengen aussagekräftig auf der gegebenen Medienfassade visualisieren und welche Problemstellungen gibt es dabei zu berücksichtigen?
- Welche Schlussfolgerungen für das System lassen sich aus den gewonnenen Prognosen ziehen?

Im Rahmen dieser Arbeit soll zusätzlich ein Prototyp entstehen, der die Funktionsweise der vorgestellten Konzepte und Algorithmen zeigt. Dieser soll so konstruiert werden, dass er auf der zukünftigen Medienfassade des FKI-Gebäudes einsetzbar ist.

### 1.3 Aufbau und Struktur

In Kapitel 2 wird zunächst auf die für diese Arbeit relevanten Grundlagen eingegangen. In diesem Zusammenhang werden verschiedene Aspekte der Visualisierung, Meteorologie und der Auswertung von Daten dargelegt. Ebenfalls werden die physikalischen Zusammenhänge der verwendeten Astronomie und Photovoltaik erläutert.

Aufbauend auf den Grundlagen folgt in Kapitel 3 eine Analyse der gegenwärtigen Situation. Hierbei wird einerseits der Aufbau des FKI-Gebäudes betrachtet, inklusive der

### **1.3 Aufbau und Struktur**

---

Medienfassade und der Photovoltaik-Anlage, und andererseits der Aufbau der Wetterstation. Im Anschluss folgt eine Analyse und Bewertung der realen und virtuellen Sensoren, sowie eine Analyse der möglichen Berechnung einer Prognose des erzeugten Stroms der Photovoltaik-Anlage. Abschließend wird die Visualisierung auf der Medienfassade genauer betrachtet.

Im darauf folgenden Kapitel 4 wird der Entwurf für die Anwendung beschrieben. Hierbei wird zunächst ein Konzept für die System- und Programmarchitektur vorgestellt, gefolgt von Skizzen für verschiedene Visualisierungen. Abschließend erfolgt ein Entwurf für die wichtigsten Algorithmen.

In Kapitel 5 werden die entscheidenden Probleme und deren Lösungen bei der Implementierung des Prototyps vorgestellt. Des Weiteren werden die Schnittstellen erläutert und die zusätzlich verwendeten Bibliotheken beschrieben. Außerdem wird das Konzept der Fehlerbehandlung erläutert. Darauf aufbauend werden in Kapitel 6 die durchgeführten Tests für den Prototyp dargelegt.

Abschließend folgen in Kapitel 7 eine Zusammenfassung und Bewertung der Ergebnisse sowie ein kurzer Ausblick.



## 2 Grundlagen

In diesem Kapitel werden die für diese Arbeit benötigten Grundlagen erläutert. Der Fokus liegt dabei einerseits auf der Visualisierung, andererseits auf dem erforderlichen Wissen aus den Bereichen der Meteorologie, Astronomie und Photovoltaik.

### 2.1 Visualisierung

Im Folgenden sollen die für diese Arbeit relevanten Bereiche der Thematik Visualisierung kurz vorgestellt werden: einerseits als Basis das RGB-Farbmodell, andererseits die Visualisierung von Daten, insbesondere an einer Medienfassade und die dafür möglichen Technologien.

#### 2.1.1 Das RGB-Farbmodell

Ein prinzipielles Verständnis für Farbmodelle ist die Grundlage für jegliche Visualisierung an einem digitalen Bildschirm. Andreas Holzinger beschreibt den Begriff Farbmodell wie folgt: „Ein Farbmodell dient dazu, Farben anhand einer bestimmten Farbskala zu beschreiben. Dieses kann nicht alle sichtbaren Farben beschreiben, sondern nur eine Untermenge daraus.“[Hol02, Seite 147] Bei der Farbdarstellung von Monitoren wird das *RGB-Farbmodell* verwendet. Auch dieses kann dementsprechend nicht alle sichtbaren Farben beschreiben, sondern nur eine Untermenge dieser. Bei dem RGB-Farbmodell handelt es sich um ein additives Farbmodell. Das heißt, dass eine Mischung der Primärfarben (**R**ot, **G**rün, **B**lau) mit voller Intensität Weiß ergibt. Des Weiteren lassen sich in dem RGB-Modell alle Farben durch diese drei Primärfarben beschreiben. In der Praxis erfolgt dies häufig mit einer Skala von 0,0 bis 1,0 oder von 0 bis 255. [Hol02, Seite 147 f.]

Schwierig bei der Nutzung des RGB-Farbmodells ist, dass es sich an der Hardware orientiert. Das hat zur Folge, dass eine Definition einer Farbe in diesem Modell auf zwei unterschiedlichen Monitoren in jeweils zwei anderen Farbwirkungen beim Betrachter

resultieren kann. Eine Angleichung verschiedener Bildschirme ist allerdings mit Hilfe einer Farbskala möglich.

### 2.1.2 Visualisierung von Daten

Bei der Visualisierung von Daten wird in der Literatur häufig auf die *Visualisierungspipeline* verwiesen, wie in Abbildung 2.1 dargestellt. Die Grundlagen bilden die Rohdaten, die über Messungen, Berechnungen oder aus Datenbanken bezogen werden. Diese Rohdaten werden im ersten Schritt aufbereitet. Das heißt, dass unter anderem soweit wie möglich Fehler korrigiert und ungültige Daten gefiltert werden. Außerdem werden charakteristische Eigenschaften, wie beispielsweise Extrema oder Statistiken, bestimmt und Datensätze vereinheitlicht. Der darauf folgende Schritt ist das Zuweisen (*Mappen*) der Daten. Dabei werden die aufbereiteten Daten in ein Bild (2D oder 3D) visualisiert, welches schließlich im letzten Schritt zu einem Bild auf dem Monitor gerendert wird. [Krö00]

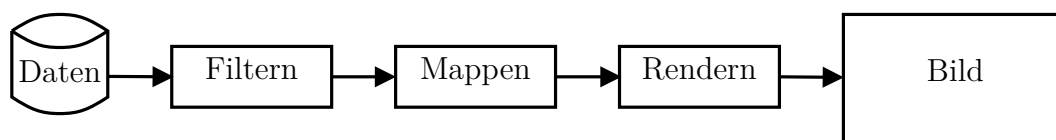


Abbildung 2.1: Die Visualisierungspipeline [Krö00]

### 2.1.3 Technologien zur Visualisierung auf einer Medienfassade

Medienfassaden können in ihrer Form und Gestaltung sehr unterschiedlich aufgebaut sein. Es gibt viele verschiedene Technologien, die jeweils ihre eigenen Vor- und Nachteile haben. Dadurch wird der Entwickler für die Visualisierung auf einer Medienfassade vor neue Herausforderungen gestellt, auf die im Detail in Unterabschnitt 3.6.1 eingegangen wird. Im Folgenden soll zunächst ein kurzer Überblick über die große Variation an Technologien geschaffen werden [BS12, Seite 3 ff.]:

**Aufprojektion:** Bei der Aufprojektion werden ein oder mehrere Beamer verwendet, um Elemente direkt auf die Fassade des Gebäudes zu projizieren. Besonders bei Lichtfestivals sind diese Installationen sehr beliebt. Eine gleichmäßige Ausleuchtung ohne Schattenwurf kann besonders bei gewölbten Fassaden anspruchsvoll zu realisieren sein. Die Ergebnisse sind meistens sehr beeindruckend anzuschauen, können jedoch nur bei Nacht gezeigt werden.

**3D Video Mapping:** Eine noch recht neue Möglichkeit der Aufprojektion ist das 3D Video Mapping. Dabei werden mehrere Beamer verwendet, die eine Animation oder ein Video auf eine plastische Ebene projizieren, wie zum Beispiel eine Häuserfassade. Dazu werden zunächst die genauen Maße des zu bespielenden Objektes ermittelt. Danach wird ein 3D Modell am Computer erstellt und das zu projizierende Material an dieses angepasst. [car11]

**Rückprojektion:** Bei der Rückprojektion werden die Beamer, im Gegensatz zur Aufprojektion, im Gebäude platziert. An den Fenstern werden spezielle Folien oder Leinwände angebracht, auf die von innen projiziert wird. Problematisch ist die teilweise aufwändige Installation im Innenraum des Gebäudes, wenn der Raum weiterhin genutzt werden soll, während die Rückprojektion läuft. Außerdem sind die meisten Installationen nur in der Nacht für den Betrachter gut zu erkennen.

**Leuchtdioden-basierte Lösungen:** Besonders häufig werden Leuchtdioden (LEDs) für die Umsetzung einer Medienfassade verwendet. Sie haben viele Vorteile, beispielsweise eine hohe Helligkeit, die auch einen Betrieb bei Tag ermöglicht, sowie ein großes Farbspektrum und eine lange Lebenszeit. Dabei werden die LEDs in verschiedenen Formen wie Streifen, Kacheln oder einem Netz verbaut.

**Organische LEDs:** Organische LEDs (OLEDs) sind die potenziellen Nachfolger der LEDs. Sie erlauben eine formbare und flexible Konstruktion, die völlig neue Einsatzmöglichkeiten eröffnet. Zurzeit sind die OLEDs noch nicht ausgereift, in Zukunft könnten sie jedoch eine attraktive Alternative darstellen.

**Elektrolumineszenz-Folie:** Die Elektrolumineszenz-Folie (EL-Folie) ist eine Folie, die durch das Anlegen einer Spannung zum Leuchten gebracht werden kann. Sie hat den großen Vorteil, dass sie auf nahezu allen Oberflächen montiert werden kann und keine spezielle Wartung benötigt. Die Folien versagen nicht plötzlich, sondern verlieren sukzessive ihre Leuchtkraft. Außerdem verbrauchen sie im Vergleich zu anderen Technologien wenig Strom und sind äußerst robust. Der Nachteil an den Folien ist, dass eine Folie immer nur eine konkrete Farbe haben kann. Diese kann nur durch ein stufenloses Dimmen noch leicht modifiziert werden.

**LCD:** Speziell angepasste LCD-Monitore sind ebenfalls für den Einsatz an einer Medienfassade vorstellbar. Diese müssen jedoch wasserfest sein, einen weiten Blickwinkel aufweisen und auch bei intensiver Sonneneinstrahlung noch gut lesbar sein. Problematisch ist die Größe von LCDs, da sich nur kleine Monitore im Bereich von 20 bis 30 Zoll preisgünstig produzieren lassen. Daher werden teilweise mehrere LCDs mit extra dünnen Rändern in einem Raster angebracht, wodurch eine sehr hohe Pixelanzahl erreicht werden kann.

**Elektronische Tinte:** Elektronische Tinte (eInk) ist besonders durch die Verwendung in E-Book-Readern bekannt geworden. Sie wird inzwischen jedoch auch in Anzeigen für Fahrgastinformationen oder kleineren Displays verwendet. Vorteilhaft sind ein geringer Strombedarf und eine sehr gute Lesbarkeit aus verschiedenen Winkeln oder bei unterschiedlichen Lichtverhältnissen. Die Nachteile sind, dass die Darstellung von Farbe noch nicht ausgereift ist und die Schaltzeiten für bewegte Bilder noch zu hoch sind. Außerdem ist die Technologie nur tagsüber nutzbar, da sie im Normalfall über keine eigene Lichtquelle verfügt.

**Laserprojektion:** Laserprojektion ist zurzeit noch nicht ausgereift. Die Technologie würde viele Vorteile bringen, wie eine sehr scharfe Projektion, auch auf unebenen Flächen, sowie eine große Reichweite. Bedenklich sind jedoch die Aspekte einer möglichen Gefährdung für das Auge und der noch sehr hohe Stromverbrauch sowie der äußerst teure Preis der Geräte.

**Weitere Leuchtmittel:** Neben den bisher genannten gibt es noch weitere Leuchtmittel, die besonders in den Anfängen häufig für Medienfassaden verwendet wurden. Insbesondere sind Neonlampen, Leuchtstoffröhren oder Halogenlampen im Einsatz. Die Vorteile sind ein geringer Preis in der Anschaffung und die unterschiedlichen Arten von Lichtspektren und Lichtintensitäten. Nachteilig ist hingegen, dass die Lebensdauer der Leuchtmittel im Vergleich zu beispielsweise LEDs gering ist und somit höhere Wartungskosten entstehen.

Eine detaillierte Beschreibung und ein Vergleich der verschiedenen Technologien ist in der Arbeit „Medienfassaden - Möglichkeiten von Visualisierung und Interaktion“ [BS12, Seite 3 ff.] zu finden.

## 2.2 Auswertung von Daten

Bei der Auswertung von Daten ergeben sich die gleichen ersten beiden Schritte wie bei der Visualisierungspipeline (Unterabschnitt 2.1.2). In einem ersten Schritt werden die Rohdaten gesammelt, welche schließlich in einem zweiten Schritt aufbereitet werden. Dabei werden die Daten vereinheitlicht, gefiltert und korrigiert. Je nach Art der Daten wird neben der häufig verwendeten statistischen Auswertung auch Data Mining (siehe Unterabschnitt 2.2.1) verwendet.

### 2.2.1 Data Mining

Beim *Data Mining* geht es um die Erkennung von Mustern in elektronischen Daten. Die Menge der gespeicherten Daten ist in den letzten Jahren stark angestiegen. Es wird davon ausgegangen, dass sie sich alle 20 Monate verdoppelt. Aufgrund solch einer überwältigenden Menge an Daten wird eine Auswertung dieser immer komplizierter. Data Mining soll hier ansetzen, indem die Suche nach Mustern zumindest teilweise automatisiert erfolgt. [WFH11]

Für ein erfolgreiches Data Mining müssen die gesammelten Daten zunächst aufbereitet und gefiltert werden. Kommen die Daten aus mehreren Datenquellen, ist es notwendig, diese auf die gleiche Datenbasis zu bringen. Es muss weiterhin geprüft werden, inwiefern mit fehlenden oder ungenauen Werten umgegangen werden soll. Der gesamte Prozess des Data Mining bringt zwar inzwischen immer öfter gute Ergebnisse, ist allerdings sehr komplex und aufwändig. [WFH11]

## 2.3 Meteorologie

Die Meteorologie ist aufgrund der vielen unterschiedlichen physikalischen und chemischen Ereignisse, die aufeinandertreffen, ein sehr umfangreicher und überaus komplexer Themenbereich. Somit sollen im Folgenden nur die für diese Arbeit wichtigsten Elemente kurz und aufs Wesentliche vereinfacht erläutert werden.

### 2.3.1 Meteorologische Größen

Die nachstehenden meteorologischen Größen sind für diese Arbeit relevant:

**Luftdruck:** Die Erdatmosphäre ist aus verschiedenen Luftschichten aufgebaut. Die unteren Schichten werden von dem Gewicht der darüberliegenden Luftschichten zusammengedrückt. Der durch dieses Gewicht der Luft entstehende Luftdruck lässt sich mit Hilfe eines Barometers in Hektopascal [hPa] messen. Der Luftdruck ist am Boden am höchsten und wird mit zunehmender Höhe immer niedriger. Zusätzlich wird er von der Temperatur und der Luftfeuchtigkeit beeinflusst. [CF99]

**Windgeschwindigkeit:** Die Geschwindigkeit des Windes wird in Meter pro Sekunde [ $\frac{m}{s}$ ] mit einem sogenannten Anemometer gemessen. Hierbei handelt es sich um ein drehbares Kreuz, an dessen Enden kleine Halbkugeln befestigt sind. Diese

## 2.3 Meteorologie

---

werden vom Wind in Bewegung gesetzt, wodurch über eine Elektronik die Windgeschwindigkeit berechnet werden kann. Eine einfache Methode, um die Windgeschwindigkeit ohne Messgerät abzuschätzen, ist die *Beaufort-Skala*, in der die Windstärke in 13 Stufen eingeteilt wird. Jede Stufe hat dabei eine Beschreibung, inwiefern sich der Wind auf die Umgebung auswirkt. [Neu04]

**Windrichtung:** Die Windrichtung wird mit einer Windfahne gemessen. Diese ist meistens so aufgebaut, dass sich eine frei drehbare Fahnenfläche durch den Wind so dreht, dass das kürzere Ende in die Richtung zeigt, aus welcher der Wind kommt. Als meteorologische Größe entspricht die Windrichtung stets der Himmelsrichtung, aus welcher der Wind kommt, und nicht derjenigen, in welche der Wind bläst. [Neu04]

**Lufttemperatur:** Bei der Lufttemperatur handelt es sich um jene Temperatur, die in der bodennahen Atmosphäre, in der Regel in Höhe von zwei Metern, ohne den Einfluss der Sonnenstrahlung oder Bodenwärme gemessen wird. Zur Messung in Grad Celsius [°C] dient im einfachsten Falle ein Thermometer, häufig wird jedoch ein Hygro-Thermogeber verwendet, der neben der Lufttemperatur auch noch die Luftfeuchtigkeit misst. [Polb]

**Relative Luftfeuchtigkeit:** Unter Luftfeuchtigkeit wird der Wasserdampf in der Luft verstanden. Die relative Luftfeuchtigkeit in Prozent [%] ist die Menge an Wasserdampf in der Luft relativ zu der Höchstmenge, die sie bei einer bestimmten Temperatur aufnehmen kann. [CF99]

**Niederschlag:** Niederschlag bezeichnet einerseits Wasser, welches von Wolken herabregnet, andererseits Wasser, das sich durch Kondensation (der Übergang eines Stoffes vom gasförmigen in den flüssigen Aggregatzustand) oder Resublimation (der Übergang eines Stoffes vom gasförmigen direkt in den festen Aggregatzustand) an Objekten absetzt. Der Niederschlag entsteht, sobald kondensierte Teilchen in der Wolke eine gewisse Größe erreicht haben und sie aufgrund der Schwerkraft aus der Wolke in Richtung Erdboden fallen. Niederschlag kann in verschiedenen Formen vorkommen: Regen, Niesel, Schnee, Graupeln oder Hagel. Das kondensierte Wasser kommt in Form von Tau und das resublimierte in Form von Reif vor. Zur Messung des Niederschlags wird ein Niederschlagsgeber verwendet, der die Menge an Niederschlag in Millimeter [mm] angibt. Diese sind häufig

## 2.4 Astronomie

---

beheizbar, damit auch im Winter der Niederschlag in Form von Schnee gemessen werden kann. [Polc, Mül05]

**Bewölkungsgrad:** Der Bewölkungsgrad gibt an, wie viele Teile des Himmels mit Wolken bedeckt sind. Der Grad wird in Achtel angegeben, wobei *sonnig* einem Bewölkungsgrad von  $\frac{0}{8}$ , *heiter*  $\frac{1}{8}$  bis  $\frac{3}{8}$ , *wolkig*  $\frac{4}{8}$  bis  $\frac{6}{8}$ , *stark bewölkt*  $\frac{7}{8}$  und *bedeckt* einem Wert von  $\frac{8}{8}$  entspricht. Die Messung kann manuell oder auch per Kamera mit Hilfe von Bildverarbeitung erfolgen. [Pola]

### 2.3.2 Wetterprognose

Die Wetterprognose ist ein sehr umfangreicher Prozess, mit dessen Hilfe die Atmosphäre zu einer bestimmten Zeit an einem bestimmten Ort prognostiziert werden soll. Hierbei wird zwischen der kurzfristigen (24 Stunden), mittelfristigen (3 bis 10 Tage) und langfristigen Wettervorhersage (mehr als 10 Tage) unterschieden. Je länger der Vorhersagezeitraum ist, desto geringer ist jedoch die Zuverlässigkeit. Liegt beispielsweise die Zuverlässigkeit bei der kurzfristigen Prognose bei ungefähr 91 Prozent, liegt sie bei einer Prognose über drei Tage schon nur noch bei etwas über 70 Prozent (Stand 2003). [WSF03]

Grundsätzlich muss zwischen zwei unterschiedlichen Arten der Wettervorhersage unterschieden werden. Einerseits gibt es die sogenannte *numerische Wettervorhersage*. Bei dieser wird aus den Messdaten der Wetterstationen sowie aus Radar- und Satellitenbildern der atmosphärische Zustand bestimmt. Dabei werden Werte wie Wind, Temperatur, Luftdruck und Luftfeuchte registriert. Mit Hilfe komplexer Gleichungen und leistungsstarker Computersysteme wird daraus der Zustand der Atmosphäre für die nächste Zeit berechnet. [Mal06]

Die zweite Art der Wettervorhersage ist die *lokale und regionale Wettervorhersage*. Diese baut auf der Numerischen auf und bestimmt regionale Erscheinungen wie Wolken, Niederschlag, Nebel usw. [Mal06]

## 2.4 Astronomie

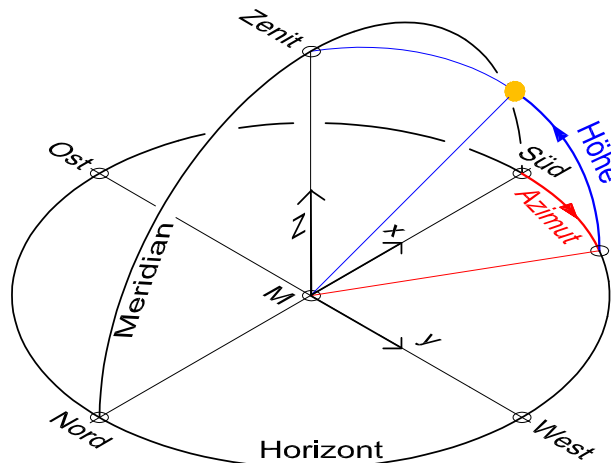
Für die Berechnung der Leistung einer Photovoltaik-Anlage zu einem gegebenen Zeitpunkt sind einige astronomische Grundkenntnisse notwendig, die in diesem Abschnitt näher erläutert werden.

### 2.4.1 Astronomische Koordinatensysteme

Astronomische Koordinatensysteme ermöglichen die Angabe der Position eines Himmelskörpers. Es existieren verschiedene Systeme, wobei im Folgenden nur die beiden für diese Arbeit relevanten Systeme, das Horizont- und das Äquatorsystem, kurz erläutert werden sollen. Prinzipiell lassen sich folgende Merkmale aufzeigen [UB02]:

1. Der *Horizont* mit den Himmelsrichtungen Nord, Ost, Süd und West.
2. Der *Zenit* senkrecht über und der *Nadir* senkrecht unter dem Betrachter auf dem Himmelskörper.
3. Der *Meridian*, der durch den Himmelspol, Zenit, Südpunkt, Nadir und Nordpunkt geht.

#### Das Horizontsystem



**Abbildung 2.2:** Astronomisches Koordinatensystem des Horizonts (auch Horizontsystem) [Wet09a]

In Abbildung 2.2 ist das Horizontsystem dargestellt. Dieses hat den Betrachter als Zentrum und ist dementsprechend an diesen gebunden. Die Position eines Objekts wird über zwei Winkel definiert: Das *Azimut* und die *Höhe*. Beide Koordinaten sind zeitabhängig. [Geb]

**Azimutwinkel  $a$ :** Der Azimutwinkel  $a$  beschreibt, wie weit ein Objekt aus der Nordbeziehungsweise Südseite herausgedreht ist [Kon08, Seite 5]. Da in der Literatur keine einheitliche Definition für den Azimutwinkel existiert, wird in der vorliegenden Arbeit die folgende verwendet:



## 2.4 Astronomie

---

Norden =  $\pm 180^\circ$

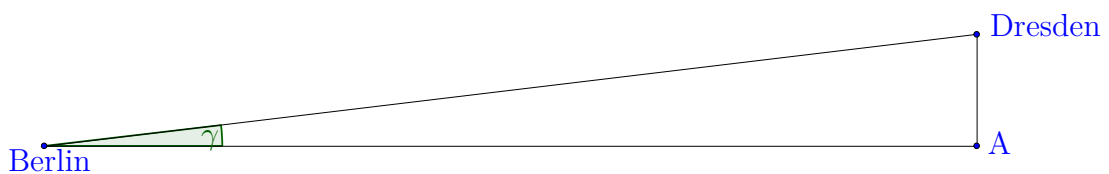
Süden =  $0^\circ$

Osten = Positiv

Westen = Negativ

**Höhenwinkel  $\gamma$ :** Der Höhenwinkel  $\gamma$  ist der Winkel zwischen dem Objekt und der Horizontebene. [Geb]

Sei beispielsweise der Bezugspunkt in Berlin mit einer Höhe von 34 m über Normalnull (Höhe über dem Meeresspiegel) und in Dresden das Objekt mit 113 m über Normalnull. Der Höhenwinkel  $\gamma$  ist nun der Winkel zwischen Berlin und Dresden, wie in Abbildung 2.3 dargestellt.



**Abbildung 2.3:** Beispiel für den Höhenwinkel  $\gamma$ ; Berlin ist der Bezugspunkt, Dresden das Objekt

### Das Äquatorsystem

Des Weiteren gibt es noch das Äquator-Koordinatensystem, wie in Abbildung 2.4 dargestellt. Bei diesem System werden die Koordinaten *Deklination* und *Stundenwinkel* genannt. Im Äquatorsystem sind diese Koordinaten eines Objektes zeitunabhängig. [Geb]

**Stundenwinkel  $\tau$ :** Der Stundenwinkel  $\tau$  ist eine Koordinate eines Himmelskörpers im Äquator-Koordinatensystem (siehe Abbildung 2.4). Er wird durch folgende Beziehung beschrieben [Mel97]:

$$\tau = (12 \text{ h} - \sigma) \cdot 15^\circ/\text{h} \quad (2.1)$$

mit:  $\sigma$  = Wahre Ortszeit (Sonnenzeit)

Die Abbildung 2.5 verdeutlicht den Zusammenhang zwischen Stundenwinkel und Stundenmaß. Dabei wird ein scheinbarer Sonnenkreis um die Erde vorgestellt, welcher in

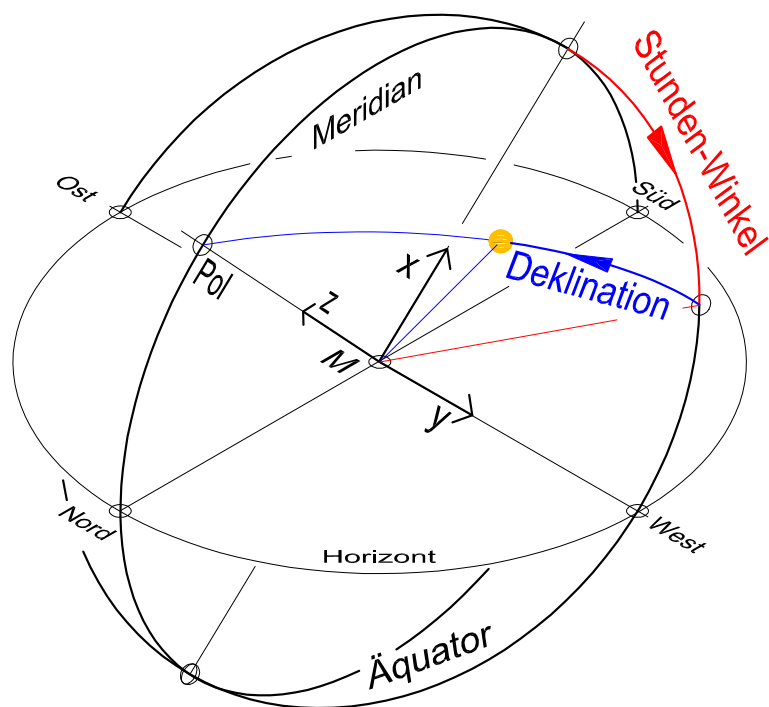


Abbildung 2.4: Das Äquator-Koordinatensystem [Wet09b]

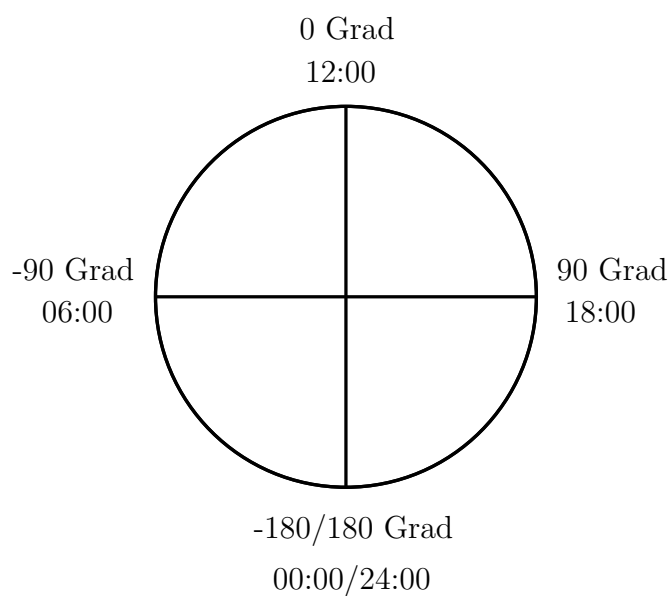


Abbildung 2.5: Zusammenhang zwischen Stundenmaß und Stundenwinkel

## 2.4 Astronomie

---

$360^\circ$  (von  $-180^\circ$  bis  $+180^\circ$ ), beziehungsweise 24 Stunden eingeteilt ist. Beispielsweise entspricht ein Stundenmaß von 10 einem Stundenwinkel von  $-30^\circ$ .

**Deklination  $\delta$ :** Die zweite Koordinate im Äquator-Koordinatensystem ist die Deklination  $\delta$ . Sie ist wie folgt definiert [Mel97, Seite 11]:

$$\sin \delta = 0,3978 \cdot \sin(x - 77,51^\circ + 1,92^\circ \cdot \sin(x)) \quad (2.2)$$

mit:  $x = 0,9856^\circ \cdot J - 2,72^\circ$

$J =$  Tag des Jahres

### 2.4.2 Sonnenposition

Zahlreiche Berechnungen sind abhängig von der Position der Sonne. Der aktuelle Sonnenstand lässt sich für jeden Ort der Erde durch zwei Winkel bestimmen: einerseits durch die *Sonnenhöhe (Elevation)* und andererseits durch das *Sonnenazimut*. [Qua09, Seite 62]

#### Sonnenhöhenwinkel $\gamma_s$

Der Sonnenhöhenwinkel  $\gamma_s$  wird als Winkel zwischen dem Horizont und dem Sonnenmittelpunkt aus Sicht des Betrachters definiert [Qua09, Seite 62] und wie folgt berechnet [Mel97]:

$$\sin \gamma_s = \sin \delta \cdot \sin b + \cos \delta \cdot \cos b \cdot \cos \tau \quad (2.3)$$

$$= \cos \psi_z \quad (2.4)$$

mit:  $\delta =$  Deklination

$b =$  Breitengrad

$\tau =$  Stundenwinkel

$\psi_z =$  Sonnenzenitwinkel

Dabei gilt für den Sonnenzenitwinkel  $\psi_z$ , der Winkel zwischen dem Zenit und dem Sonnenmittelpunkt aus Sicht des Betrachters:

$$\psi_z = 90^\circ - \gamma_s \quad (2.5)$$

### Sonnenazimutwinkel $a_S$

Das Sonnenazimut ist der Winkel zwischen dem Vertikalkreis durch den Sonnenmittelpunkt und der geografischen Südrichtung [Qua09, Seite 62]. Für den Sonnenazimutwinkel  $a_S$  gilt [Mel97]:

$$\cos |a_S| = \frac{\sin \gamma_S \cdot \sin b - \sin \delta}{\cos \gamma_S \cdot \cos b} \quad (2.6)$$

mit:  $\gamma_S$  = Sonnenhöhenwinkel

$b$  = Breitengrad

$\delta$  = Deklination

Hierbei ist  $a_S$  am Vormittag positiv und am Nachmittag negativ.

### Die wahre Sonnenzeit $\sigma$

Die wahre Sonnenzeit  $\sigma$  (oder auch wahre Ortszeit) ist die Zeit, die von einer Sonnenuhr angezeigt wird. Dabei entsprechen 12 Stunden dem höchsten Stand der Sonne auf ihrer Kreisbahn am Himmel. Die wahre Sonnenzeit verläuft aufgrund der ungleichförmigen Bahnbewegung der Erde nicht gleichmäßig. [UB02]

Sie lässt sich wie folgt berechnen [Drü12, Seite 11]:

$$\sigma = \text{Standardzeit} + 4 \cdot (\varphi_{\text{Lokal}} - \varphi_{\text{St}}) + E \quad (2.7)$$

mit:  $\varphi_{\text{Lokal}}$  = geografischer Längengrad des Standorts

$\varphi_{\text{St}}$  = geografischer Längengrad des Standardmeridian der lokalen Zeitzone  
(15° für die mitteleuropäische Zeitzone)

$E$  = Zeitgleichung

Um einen gleichmäßigen Zeitverlauf zu erreichen, wurde die *mittlere Sonnenzeit* eingeführt. Bei dieser wird von einer mittleren Sonne ausgegangen, die den Äquator gleichförmig in derselben Zeit umläuft und insgesamt die gleiche Zeit wie die wahre Sonne für den Umlauf der Ekliptik benötigt [UB02]. Die Differenz zwischen der wahren Sonnenzeit und der mittleren Sonnenzeit beschreibt die Zeitgleichung  $E$  [Drü12, Seite 11]:

$$E = 9,87 \cdot \sin(2 \cdot B) - 7,53 \cdot \cos(B) - 1,5 \cdot \sin(B) \quad (2.8)$$

mit:  $B = (J - 81) \cdot 0,989$

$J$  = Tag des Jahres

### Einfallswinkel $\psi$

Der Einfallswinkel  $\psi$  der Strahlung auf eine geneigte Fläche ist wie nachstehend festgelegt [Mel97, Seite 9]:

$$\cos \psi = \sin \gamma_S \cdot \cos \gamma_E + \cos \gamma_S \cdot \sin \gamma_E \cdot \cos(a - a_S) \quad (2.9)$$

mit:  $\gamma_S$  = Sonnenhöhenwinkel

$\gamma_E$  = Neigungswinkel der bestrahlten Fläche

$a$  = Azimutwinkel

$a_S$  = Sonnenazimutwinkel

### Beispiel für die Berechnung der Sonnenposition

Das FKI-Gebäude befindet sich an der Position Längengrad  $\varphi_{Lokal} = 13,53^\circ$  und Breitengrad  $b = 52,46^\circ$ . Dementsprechend ist der geografische Längengrad des Standardmeridian der lokalen Zeitzone  $\varphi_{St} = 15^\circ$  für die mitteleuropäische Zeitzone. Es soll nun die Sonnenposition aus Sicht des FKI-Gebäudes am 17. Juni 2012 (169. Tag im Jahr) um 10:00 Uhr berechnet werden. Zuerst wird die wahre Sonnenzeit  $\sigma$  ermittelt, für die zunächst  $B$  und  $E$  bestimmt werden müssen:

$$\begin{aligned} B &= (J - 81) \cdot 0,989 \\ &= (169 - 81) \cdot 0,989 \\ &= \underline{\underline{87,032}} \\ E &= 9,87 \cdot \sin(2 \cdot B) - 7,53 \cdot \cos(B) - 1,5 \cdot \sin(B) \\ &= 9,87 \cdot \sin(2 \cdot 87,032) - 7,53 \cdot \cos(87,032) - 1,5 \cdot \sin(87,032) \\ &= \underline{\underline{-0,8671}} \end{aligned}$$

Mit Hilfe der Ergebnisse kann folglich die wahre Sonnenzeit  $\sigma$  bestimmt werden:

$$\begin{aligned} \sigma &= \text{Standardzeit} + 4 \cdot (\varphi_{Lokal} - \varphi_{St}) + E \\ &= 600 + 4 \cdot (13,53 - 15) - 0,8671 \\ &= 593,2529 \text{ min} = \underline{\underline{9,8875 \text{ h}}} \\ &= 9:53 \text{ Uhr} \end{aligned}$$

## 2.4 Astronomie

---

Die wahre Sonnenzeit  $\sigma$  ist die Basis für alle weiteren Berechnungen bezüglich der Sonnenposition. Als nächstes wird der Stundenwinkel  $\tau$  errechnet:

$$\begin{aligned}\tau &= (12 \text{ h} - \sigma) \cdot 15^\circ/\text{h} \\ &= (12 \text{ h} - 9,8875 \text{ h}) \cdot 15^\circ/\text{h} \\ &= \underline{\underline{31,6875^\circ}}\end{aligned}$$

Da die Uhrzeit  $< 12:00$  Uhr ist, gilt:  $\tau = \tau \cdot (-1) \Rightarrow \tau = -31,6875^\circ$ .

Es folgt die Berechnung der Deklination  $\delta$ , wobei zuerst  $x$  bestimmt werden muss:

$$\begin{aligned}x &= 0,9856^\circ \cdot J - 2,72 \\ &= 0,9856^\circ \cdot 169 - 2,72 \\ &= \underline{\underline{163,8464^\circ}}\end{aligned}$$

Mit Hilfe von  $x$  kann nun die Deklination  $\delta$  errechnet werden:

$$\begin{aligned}\sin \delta &= 0,3978 \cdot \sin(x - 77,51^\circ + 1,92^\circ \cdot \sin(x)) \\ \sin \delta &= 0,3978 \cdot \sin(163,8464^\circ - 77,51^\circ + 1,92^\circ \cdot \sin(163,8464^\circ)) \\ \delta &= \arcsin(0,3981^\circ) \\ &= \underline{\underline{23,4594^\circ}}\end{aligned}$$

Damit sind die Koordinaten der Sonne bestimmt:

Der Stundenwinkel  $\tau = -31,6875^\circ$

Die Deklination  $\delta = 23,4594^\circ$

Mit Hilfe dieser kann jetzt der Sonnenhöhenwinkel  $\gamma_S$  berechnet werden:

$$\begin{aligned}\sin \gamma_S &= \sin \delta \cdot \sin b + \cos \delta \cdot \cos b \cdot \cos \tau \\ &= \sin(23,4594^\circ) \cdot \sin(52,46^\circ) + \\ &\quad \cos(23,4594^\circ) \cdot \cos(52,46^\circ) \cdot \cos(31,6875^\circ) \\ \gamma_S &= \arcsin(0,7913^\circ) \\ &= \underline{\underline{52,3062^\circ}}\end{aligned}$$

Zum Schluss wird noch der Sonnenazimutwinkel  $a_S$  bestimmt, womit alle wichtigen Werte für die Sonnenposition berechnet sind:

$$\begin{aligned}\cos |a_S| &= \frac{\sin \gamma_S \cdot \sin b - \sin \delta}{\cos \gamma_S \cdot \cos b} \\ &= \frac{\sin(52,3062^\circ) \cdot \sin(52,46^\circ) - \sin(23,4594^\circ)}{\cos(52,3062^\circ) \cdot \cos(52,46^\circ)} \\ |a_S| &= \arccos(0,6156^\circ) \\ a_S &= \underline{\underline{52,0067^\circ}}\end{aligned}$$

### 2.4.3 Globalstrahlung

Die Globalstrahlung  $E_G$  bezeichnet die gesamte Strahlung der Sonne auf eine beliebige Fläche der Erde. Sie setzt sich aus den folgenden Strahlungsteilen zusammen [Mel97, Seite 5 ff.]:

**Direkte Strahlung  $E_D$ :** Die direkte oder auch terrestrische Strahlung ist abhängig von der extraterrestrischen Strahlung. Bei dieser handelt es sich um die Intensität der Sonnenstrahlung oberhalb der Erdatmosphäre. Innerhalb der Atmosphäre wird diese durch Streu- und Absorptionsvorgänge (Trübungsfaktor) abgeschwächt. Die direkte Strahlung auf eine Fläche auf der Erde ist daher die extraterrestrische Strahlung in Abhängigkeit eines Trübungsfaktors der Atmosphäre sowie der Sonnenposition und des daraus folgenden Einfallswinkels.

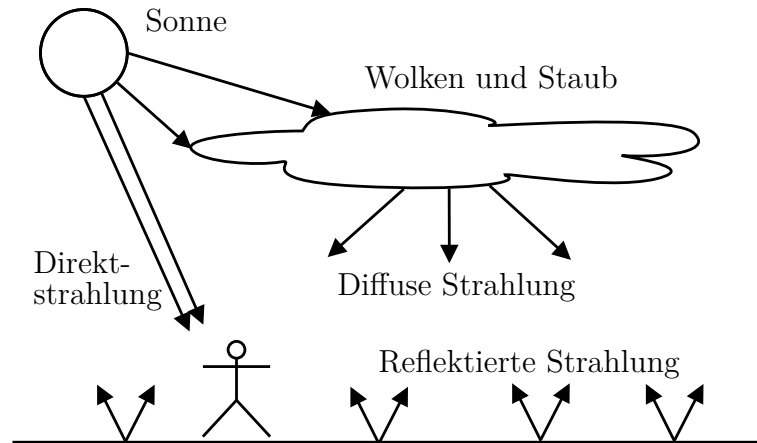
**Diffuse Strahlung  $E_H$ :** Die diffuse Strahlung oder auch Himmelstrahlung wird durch die Streuvorgänge der Atmosphäre erzeugt.

**Reflektierte Strahlung  $E_R$ :** Die reflektierte Strahlung ist der Anteil der Strahlung, der vom Boden reflektiert wird. Sie hängt stark von der Bodenbeschaffenheit der Umgebung ab. Beispielsweise reflektiert eine Eisfläche sehr gut, wohingegen Wohnumgebung weniger gut reflektiert.

Zusammenfassend visualisiert Abbildung 2.6 die drei Bestandteile der Globalstrahlung. Diese ergeben zusammenaddiert die Globalstrahlung auf die Erdoberfläche:

$$E_G = E_D + E_H + E_R \quad (2.10)$$

Um die einzelnen Elemente der Globalstrahlung zu berechnen, werden neben der Sonnenposition und der Position der bestrahlten Fläche noch weitere Faktoren benötigt, die im Weiteren näher erläutert werden.



**Abbildung 2.6:** Die drei Bestandteile der Globalstrahlung: direkte, diffuse und reflektierte Strahlung [Mel97, Seite 13]

### Extraterrestrische Strahlung $E_0$

Die Extraterrestrische Strahlung  $E_0$  (Solarkonstante) beschreibt die Intensität der Sonnenstrahlung oberhalb der Erdatmosphäre [Mel97, Seite 5]. Aufgrund des sich ändernden Abstands zwischen Erde und Sonne ist sie vom aktuellen Tag des Jahres abhängig und wird wie folgt ermittelt [Drü12, Seite 14]:

$$E_0 = \bar{E}_0 \cdot \left( 1 + 0,033 \cdot \cos \left( \frac{2 \cdot \pi \cdot J}{365} \right) \right) \quad (2.11)$$

mit:  $\bar{E}_0$  = Mittlere Solarkonstante =  $1367 \frac{\text{W}}{\text{m}^2}$   
 $J$  = Tag des Jahres

### Terrestrische Strahlung $E_D$

Die terrestrische Strahlung (oder auch Direktstrahlung) auf die Erdoberfläche lässt sich wie nachstehend berechnen [Mel97, Seite 8 f.]:

$$E_D = E_0 \cdot \exp \left( - \frac{T_R}{0,9 + 9,4 \cdot \sin \gamma_S} \right) \quad (2.12)$$

mit:  $E_0$  = Extraterrestrische Strahlung  
 $T_R$  = Trübungsfaktor  
 $\gamma_S$  = Sonnenhöhenwinkel



## 2.4 Astronomie

---

### Trübungsfaktor $T_R$

Beim Durchgang der Strahlung durch die Atmosphäre kommt es zu Absorptions- und Streuverlusten, welche im *Trübungsfaktor* zusammengefasst werden. Die Tabelle 2.1 zeigt für die einzelnen Monate des Jahres die über Jahrzehnte gemessenen mittleren Trübungsfaktoren. [Mel97, Seite 6f.]

**Tabelle 2.1:** Mittlere gemessene Trübungsfaktoren für die einzelnen Monate des Jahres und die unterschiedlichen Gebietsarten [Mel97, Seite 7]

Gebiet	Jan.	Feb.	März	April	Mai	Juni
Hochgebirge	1,8	1,9	2,1	2,2	2,4	2,7
Flachland	2,1	2,2	2,5	2,9	3,2	3,4
Stadt	3,1	3,2	3,5	3,9	4,1	4,2
Gebiet	Juli	Aug.	Sept.	Okt.	Nov.	Dez.
Hochgebirge	2,7	2,7	2,5	2,1	1,9	1,8
Flachland	3,5	3,3	2,9	2,6	2,3	2,2
Stadt	4,3	4,2	3,9	3,6	3,3	3,1

### Direkte Strahlung auf eine beliebig geneigte Fläche $E_{D,g}$

Der Anteil der Direktstrahlung auf eine beliebig geneigte Fläche  $E_{D,g}$  ist von dem Einfallswinkel dieser abhängig [Mel97, Seite 11 f.]:

$$E_{D,g} = E_D \cdot \cos \psi \quad (2.13)$$

mit:  $E_D$  = Direkte Strahlung

$\psi$  = Einfallswinkel

### Diffuse Strahlung auf eine horizontale Fläche $E_{H,h}$

Die diffuse Strahlung auf eine horizontale Fläche wird wie folgt bestimmt [Mel97, Seite 11 f.]:

$$E_{H,h} = \frac{1}{3} \cdot (E_0 - E_D) \cdot \sin \gamma_S \quad (2.14)$$

mit:  $E_0$  = Extraterrestrische Strahlung

$E_D$  = Direkte Strahlung

$\gamma_S$  = Sonnenhöhenwinkel

### Diffuse Strahlung auf eine beliebig geneigte Fläche $E_{H,g}$

Die folgende Formel verfolgt einen isotropen Ansatz. Das heißt, dass davon ausgegangen wird, dass aus allen Himmelsrichtungen stets der gleiche Anteil an Himmelsstrahlung vorhanden ist. Eine grobe Abschätzung für die diffuse Strahlung auf eine beliebig geneigte Fläche nach dem isotropen Ansatz kann wie nachstehend berechnet werden [Qua09, Seite 67]:

$$E_{H,g} = E_{H,h} \cdot \frac{1}{2} \cdot (1 + \cos \gamma_E) \quad (2.15)$$

mit:  $E_{H,h}$  = Diffuse Strahlung auf eine horizontale Fläche

$\gamma_E$  = Neigungswinkel der Fläche

### Reflektierte Strahlung $E_R$

Für die Bestimmung der reflektierten Strahlung gibt es verschiedene Ansätze. Im Folgenden wird ein isotroper Ansatz gewählt, da die anisotropen Berechnungen nur eine vernachlässigbar bessere Genauigkeit ergeben [Qua09, Seite 68]. Die reflektierte Strahlung  $E_R$  lässt sich dementsprechend wie folgt berechnen [Mel97, Seite 12]:

$$E_R = \rho_B \cdot (E_{D,h} + E_{H,h}) \cdot \sin^2 \left( \frac{\gamma_E}{2} \right) \quad (2.16)$$

mit:  $\rho_B$  = Reflexion des Bodens (siehe Tabelle 2.2)

$E_{D,h}$  = Direkte Strahlung auf eine horizontale Fläche

$E_{H,h}$  = Diffuse Strahlung auf eine horizontale Fläche

$\gamma_E$  = Neigungswinkel

**Tabelle 2.2:** Reflexionsgrade verschiedener Oberflächen  $\rho_B$  [Mel97]

Gebiet	Bodenreflexion
Felder mit Schneebedeckung	0,61 – 0,73
Eis/schneebedecktes Wasser	0,68
offenes Wasser	0,16
Wohngebiete	0,21 – 0,45

## 2.5 Photovoltaik

Das Wort Photovoltaik setzt sich aus dem griechischen Wort *phos* für Licht und dem Namen des Physikers Alessandro Volta zusammen. Dabei wird unter Photovoltaik der Vorgang der direkten Stromerzeugung aus Licht verstanden. Die Grundlage ist der sogenannte *innere Fotoeffekt*, welcher die Freisetzung von positiven und negativen Ladungsträgern in einem Festkörper durch Lichteinstrahlung beschreibt. [Kon08, Hei]

### 2.5.1 Photovoltaik-Anlagen

Im einfachsten Fall besteht eine Photovoltaik-Anlage (PV-Anlage), wie in Abbildung 2.7 gezeigt, aus einem Solarmodul, einem Wechselrichter und einem Stromzähler, sowie der benötigten Verkabelung zwischen den Komponenten. Trifft Sonnenlicht auf das Solarmodul, wird auf Basis des inneren Fotoeffekts Strom erzeugt. Da es sich bei diesem Strom um Gleichstrom (DC) handelt, wird ein Wechselrichter benötigt, der diesen in Wechselstrom (AC) umwandelt. Normalerweise verfügen Photovoltaik-Anlagen nicht nur über ein Solarmodul, sondern über eine Vielzahl von Modulen. Bei größeren Anlagen werden diese Module in sogenannte *Stränge* (*Strings*) gegliedert. Außerdem verfügen die Wechselrichter über einen *MPP-Regler* (MPP = Maximum Power Point). Dieser sorgt dafür, dass die Solarmodule stets an ihrem bestmöglichen Betriebspunkt betrieben werden. Größere Anlagen verfügen teilweise noch über weitere Leistungsoptimierer, Überwachungsfunktionen oder ein Sonnenstandsnachführungssystem. [Fri12, Seite 1 f.]

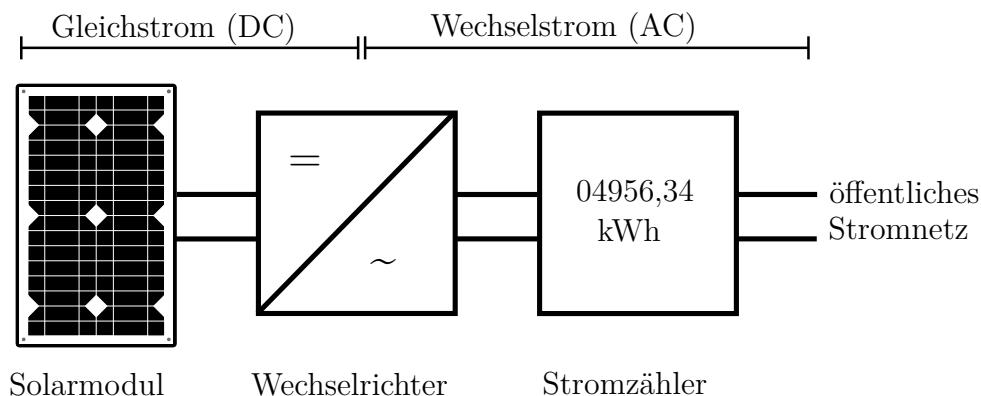


Abbildung 2.7: Minimaler Aufbau einer Photovoltaik-Anlage [Fri12, Seite 1]

### 2.5.2 Berechnung der erzeugten Leistung

Die Kenndaten einer Solarzelle beschreiben normalerweise nur die Leistung unter Standardtestbedingungen. Diese Leistung wird in realen Einsatzszenarien jedoch durch verschiedene Faktoren beeinflusst [Qua12]:

$$P = \eta \cdot E_G \cdot A \cdot (1 + \alpha \cdot (\vartheta_M - 25 \text{ °C})) \cdot f \quad (2.17)$$

mit:  $\eta$  = Wirkungsgrad der Solarzelle

$E_G$  = Globalstrahlung auf die Solarzelle

$A$  = Fläche der Solarzelle

$\alpha$  = Temperaturkoeffizient

$\vartheta_M$  = Modultemperatur

$f$  = Verlustfaktor

#### Verlustfaktor $f$

Bereits bevor das Sonnenlicht die Solarzelle erreicht hat, entstehen die ersten Verluste durch Verschattung. Diese kann durch Vögel, Bäume oder Nachbargebäude entstehen, wohingegen die Bewölkung nicht betrachtet wird. Weitere Verluste können zum Beispiel durch die Verschmutzung der Fassade, Wechselrichterverluste oder Leitungsverluste vorkommen. Häufig wird ein geschätzter, pauschaler Wert von 10 % bzw. ein Verlustfaktor von 0,9 angenommen [Qua12]. [Fri12, Seite 5 f.]

#### Die Temperatur eines Solarmoduls $\vartheta_M$

Die Leistung einer Solarzelle ist nicht konstant, sondern sinkt proportional zum Anstieg der Temperatur. Die Stärke der Temperaturauswirkung ist vom Material der Solarzelle abhängig und wird durch den Temperaturkoeffizienten  $\alpha$  beschrieben. Typisch sind dabei Werte in der Größenordnung von  $-0,4 \text{ \% pro } ^\circ\text{C}$ . Beispielsweise würde bei einem Temperaturanstieg von  $25 \text{ }^\circ\text{C}$  die Leistung um  $10 \text{ \%}$  sinken. Dabei ist zu beachten, dass dies die Angaben für Standardtestbedingungen mit  $25 \text{ }^\circ\text{C}$  Modultemperatur sind. Diese Temperatur kann jedoch normalerweise nur im Winter erreicht werden. Ist die Modultemperatur dennoch unter  $25 \text{ }^\circ\text{C}$ , wird sogar eine Verbesserung der Leistung erreicht. [Qua09, Seite 190 f.]

Mit Hilfe der Umgebungstemperatur und der Proportionalitätskonstante kann die aktuelle Temperatur der Solarzelle bestimmt werden. Die *Proportionalitätskonstante* ist

## 2.5 Photovoltaik

---

abhängig von der Einbauvariante der Module und beschreibt die Auswirkung einer guten oder schlechten Hinterlüftung der Solarzelle. [Qua09, Seite 190 f.]

Die Modultemperatur  $\vartheta_M$  wird wie folgt errechnet [Qua09, Seite 188 ff.]:

$$\vartheta_M = \vartheta_U + c \cdot \frac{E_G}{1000 \frac{\text{W}}{\text{m}^2}} \quad (2.18)$$

mit:  $\vartheta_U$  = Umgebungstemperatur

$c$  = Proportionalitätskonstante

$E_G$  = Globalstrahlung auf die Solarzelle

Die folgende Modultemperatur  $\vartheta_M$  ergibt sich beispielsweise bei einer Proportionalitätskonstante  $c = 35 \text{ }^\circ\text{C}$ , was für ein in die Fassade integriertes Modul mit guter Hinterlüftung steht, einer Umgebungstemperatur  $\vartheta_U = 20,1 \text{ }^\circ\text{C}$  und einer Globalstrahlung  $E_G = 950 \frac{\text{W}}{\text{m}^2}$ :

$$\begin{aligned} \vartheta_M &= \vartheta_U + c \cdot \frac{E_G}{1000 \frac{\text{W}}{\text{m}^2}} \\ &= 20,1 \text{ }^\circ\text{C} + 35 \text{ }^\circ\text{C} \cdot \frac{950 \frac{\text{W}}{\text{m}^2}}{1000 \frac{\text{W}}{\text{m}^2}} \\ &= \underline{\underline{53,35 \text{ }^\circ\text{C}}} \end{aligned}$$

### Der Wirkungsgrad eines Solarmoduls $\eta$

Der Wirkungsgrad beschreibt das Verhältnis von nutzbringender gewonnener Energie zu aufgewendeter Energie [Qua09, Seite 16] und wird bei Solarmodulen folgendermaßen bestimmt [Qua12]:

$$\eta = \eta_{STC} \cdot \left( 1 + 0,0825 \cdot \ln \frac{E_G}{1000 \frac{\text{W}}{\text{m}^2}} \right) \quad (2.19)$$

mit:  $\eta_{STC}$  = Wirkungsgrad unter Standardtestbedingungen

$E_G$  = Globalstrahlung auf die Solarzelle

Dabei ist der Wirkungsgrad unter Standardtestbedingungen  $\eta_{STC}$  (STC = Standard Test Conditions) für Solarzellen wie folgt definiert [Qua09, Seite 188]:

$$\eta_{STC} = \frac{P_{STC}}{E \cdot A} \quad (2.20)$$

## 2.5 Photovoltaik

---

mit:  $P_{STC}$  = Leistung unter Standardtestbedingungen

$E$  = Bestrahlungsstärke auf die Solarzelle

(unter Standardtestbedingungen ist  $E = 1000 \frac{\text{W}}{\text{m}^2}$ )

$A$  = Fläche der Solarzelle

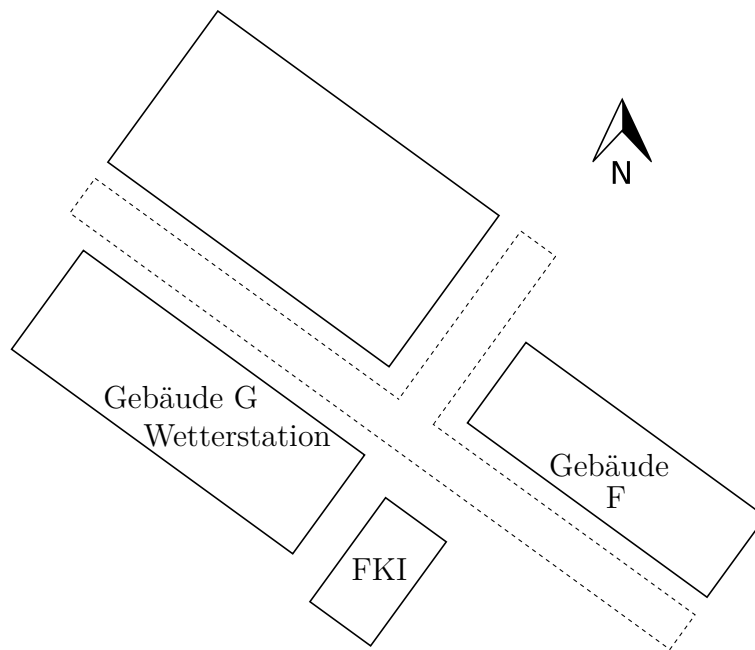
Beispielsweise ergibt sich der Wirkungsgrad  $\eta_{STC}$  bei einer gegebenen Leistung unter Standardtestbedingungen  $P_{STC} = 340 \text{ W}$  und einer Fläche  $A = 4,236 \text{ m}^2$ :

$$\begin{aligned}\eta_{STC} &= \frac{P_{STC}}{E \cdot A} \\ &= \frac{340 \text{ W}}{1000 \frac{\text{W}}{\text{m}^2} \cdot 4,236 \text{ m}^2} \\ &= \underline{\underline{8,026 \%}}\end{aligned}$$

# 3 Analyse

In diesem Kapitel erfolgt zunächst eine Betrachtung des FKI-Gebäudes mit der Medienfassade, der PV-Anlage und der benachbarten Wetterstation. Darauf aufbauend folgt die Definition einer Anforderungsanalyse sowie eine Analyse und Bewertung der möglichen realen und virtuellen Sensoren. Abschließend werden die Möglichkeiten der Visualisierung auf der Medienfassade genauer untersucht.

## 3.1 Aufbau und Lage des FKI-Gebäudes



**Abbildung 3.1:** Position des FKI-Gebäudes auf dem Campus der HTW Berlin (Auszug)

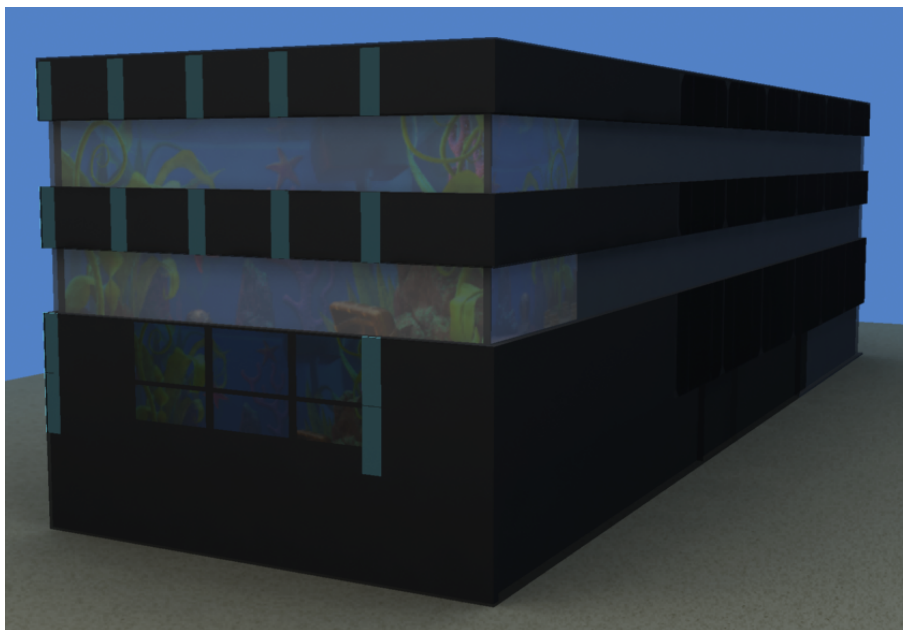
Die Fassade des Gebäudes wird mit unterschiedlichen Technologien ausgebaut. Einerseits wird sich auf der Nordseite des Gebäudes die Medienfassade befinden, andererseits wird auf den weiteren drei Seiten eine Photovoltaik-Anlage installiert sein. Das Gebäu-

### 3.1 Aufbau und Lage des FKI-Gebäudes

---

de befindet sich auf den Koordinaten  $52,455186^\circ$  Breite und  $13,525380^\circ$  Länge<sup>1</sup>. Es ist in einem stark frequentierten Bereich neben dem Mensagebäude G platziert, wie in Abbildung 3.1 dargestellt.

#### 3.1.1 Aufbau der Medienfassade



**Abbildung 3.2:** Visualisierung der Medienfassade am Modell des FKI-Gebäudes

Wie in der Abbildung 3.2 dargestellt, wird die Medienfassade mit drei unterschiedlichen Visualisierungstechnologien realisiert: LEDs, Rückprojektion und Ambient-Light (siehe Unterabschnitt 2.1.3). Das Gebäude verfügt außerdem über eine wenige Zentimeter von der Wand abgesetzte Glasfassade (siehe Abbildung 3.3). Bei der Nutzung der LEDs oder des Ambient-Lights muss berücksichtigt werden, dass dieses Glas leicht getönt ist, denn diese beiden Technologien werden hinter der Glasfassade an der Wand des Gebäudes montiert.

Der in der Abbildung zentrale Bereich unter der mittleren Fensterreihe ergibt mit mehreren LED-Kacheln eine zusammenhängende LED-Wand. Diese wird aus 12 Reihen mit jeweils 36 *Grid 25-Modulen* von der Firma *Freitag Technologies* bestehen. Ein Modul hat eine Pixelanzahl von  $8 \times 8$  Pixeln [Fre], wodurch eine Reihe dementsprechend  $36 \times 8 = 288$  Pixel in der Breite und eine Spalte  $8 \times 12 = 96$  Pixel in der

---

<sup>1</sup>Quelle: <http://maps.google.de/>





**Abbildung 3.3:** Foto des FKI-Gebäudes (29.01.2013)

Höhe hat. Daraus ergibt sich die Gesamtanzahl von **288 x 96** Pixeln, bzw. ein Bildformat von **3 : 1**. Die Module der LED-Wand werden über DVI ansteuerbar sein. [Sch12]

Des Weiteren werden die Fenster des Gebäudes voraussichtlich mit Rückprojektion über Beamer bespielt. Hierbei werden vier HD-Beamer pro Etage zum Einsatz kommen, welche die komplette Fensterfläche beleuchten. Zusätzlich wird die Rückprojektion noch auf die ersten Fenster an den länglichen Seiten des Gebäudes erweitert. Es wird jeweils ein weiterer Beamer pro Seite und Etage verwendet. Dies bietet die Möglichkeit, Elemente auf getrennten Häuserseiten darzustellen. Die Rückprojektion wird jedoch nur bei Nacht zum Einsatz kommen, weil tagsüber die Räumlichkeiten als Büroräume genutzt werden. Die nachts benötigte Rückprojektionsfolie kann tagsüber eingerollt werden. Um einen möglichst reibungslosen Betrieb zu ermöglichen, werden die Beamer über eine zentrale Stelle ein- und ausschaltbar sein. [Sch12]

Als dritte Komponente wird Ambient-Light verwendet. *Ambient-Light* ist in diesem Zusammenhang eine Bezeichnung für LED-Leuchten, die mit dem Zweck eingesetzt werden, ein Umgebungslicht zu erzeugen. Dieses Umgebungslicht soll eine Glaskachel auf der Fassade füllen. Sowohl zwischen den beiden Fensterreihen als auch links und rechts von der LED-Wand werden diese Bereiche einzeln ansteuerbar in verschiedenen Farben zu beleuchten sein. Die Steuerung der LED-Leuchten wird per *DMX* möglich sein.

### 3.1 Aufbau und Lage des FKI-Gebäudes

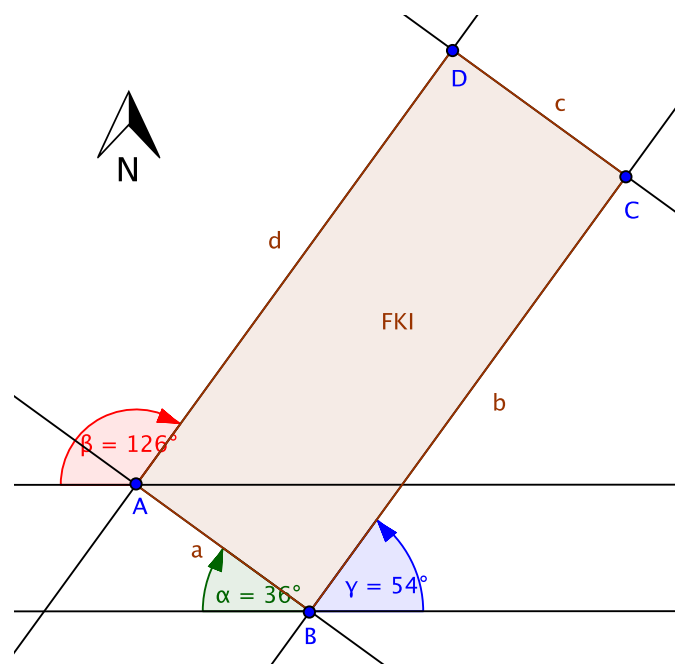
---

#### Starten einer Anwendung auf der Medienfassade

Neben dem Dashboard sollen noch viele weitere Anwendungen auf der Medienfassade laufen. Daher ist geplant, dass die Medienfassade über eine Art *Launcher* verfügen wird, der die Steuerung der verschiedenen Anwendungen vornimmt. Der Launcher verfügt über eine gewisse Anzahl von Anwendungen, für die jeweils eine Standard-Laufzeit konfiguriert ist. Zudem ist eine der Anwendungen als Standard definiert. Das heißt, sie wird immer gestartet, wenn gerade keine andere Anwendung läuft. Das Dashboard als rein informatives Programm würde sich für diese Standard-Anwendung anbieten, da viele der anderen geplanten Programme eine Interaktion erfordern.

Der erste Planungsschritt sieht vor, dass die Anwendungen manuell durch eine dazu berechtigte Person gestartet werden. Es gibt allerdings bereits Überlegungen, eine Art Abstimmung einzuführen. Dann könnten die Betrachter der Fassade mitbestimmen, welches Programm als nächstes gestartet wird.

#### 3.1.2 Aufbau der Photovoltaik-Anlage

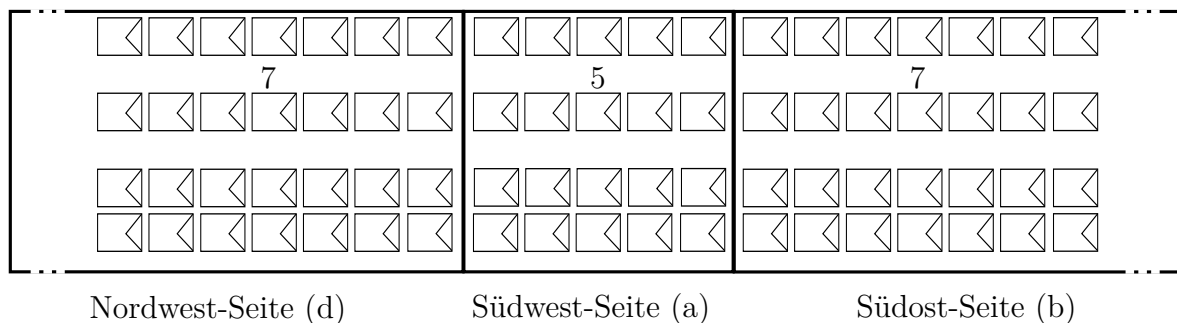


**Abbildung 3.4:** Position des FKI-Gebäudes mit Azimutwinkel der Fassadenseiten

Die Photovoltaik-Anlage erstreckt sich über drei Seiten des Gebäudes. In Abbildung 3.4 ist die genaue Positionierung des Gebäudes zu sehen. Die Module der PV-Anlage sind

### 3.1 Aufbau und Lage des FKI-Gebäudes

Schüco Fassadenmodule der *ProSol TF+ Serie*, welche auf den Seiten a (Südwest), b (Südost) und d (Nordwest) montiert sind. Diese sind direkt in die Glasfassade integriert und haben dementsprechend eine Neigung von  $90^\circ$ . Durch diese Integration in die abgesetzte Glasfassade verfügen die Module über eine gute Hinterlüftung zur Kühlung. Die eingezeichneten Winkel  $\alpha = 36^\circ$ ,  $\beta = 126^\circ$  und  $\gamma = 54^\circ$  sind die Azimutwinkel der jeweiligen Fassadenseite. Die exakte Aufteilung der Solarmodule auf der Fassadenseite ist in der Abbildung 3.5 zu sehen und der Tabelle 3.1 zu entnehmen.



**Abbildung 3.5:** Aufbau der Solarmodule an der Fassade des FKI-Gebäudes

**Tabelle 3.1:** Aufteilung der PV-Module an der Fassade

Seite	Ausrichtung		Anzahl Module			
	Himmelsrichtung	Azimutwinkel	EG	1. OG	2. OG	Gesamt
a	Südwest	$36^\circ$	10	5	5	20
b	Südost	$54^\circ$	14	7	7	28
d	Nordwest	$126^\circ$	14	7	7	28

Jedes Modul hat dabei folgende Kenndaten:

Proportionalitätskonstante $c$	$= 35 \text{ }^\circ\text{C}$ (geschätzt, siehe Tabelle 5.9 in [Qua09])
Leistung $P_{STC}$	$= 340 \text{ Wp} \pm 10 \%$ (Wp = Watt Peak)
Verlustfaktor $f$	$= 0,9$ (geschätzt)
Größe $A$	$= 2,566 \text{ m} \cdot 1,651 \text{ m} = 4,236 \text{ m}^2$
Anzahl der Module $n$	$= 76$
Art	$= \text{amorph/mikrokristallin}$
Bezeichnung	$= \text{ProSolTF+}$
Neigungswinkel $\gamma_E$	$= 90^\circ$
Gewicht	$= 124 \text{ kg}$

### 3.2 Aufbau der Wetterstation der HTW Berlin

---

Zusätzlich verfügt die Photovoltaik-Anlage über einen *Schüco Sunalyzer Web PR Datenlogger*. Dieser verfügt über eine Ethernet-Schnittstelle und protokolliert die nachstehenden Daten [SCH11]:

**Wechselrichter:** Für jeden Wechselrichter werden Daten wie die Gerätetemperatur [°C], der Tagesenergieertrag [kWh] oder die Spannung [V], Strom [A] und Leistung [W] für jeden Strang gesammelt. Die Datensätze sind immer mit einem Zeitstempel und einer Identifikations-Adresse des Wechselrichters für die exakte Zuordnung versehen.

**Generatoranschlusskasten:** Für den Generatoranschlusskasten (Verbinder mehrerer Stränge mit einem Wechselrichter) wird der Strom des jeweiligen Strangs [A] mit einem Zeitstempel gespeichert.

**Summendaten:** Zusätzlich wird ein Summenprotokoll erstellt. In diesem wird nach der erfolgreichen Abfrage der Wechselrichter ein Summendatensatz für die Gesamtanlage für die Leistung (AC und DC) [W] und den Tagesertrag [kWh] berechnet und gespeichert.

**Zusatzmesswerte:** Über weitere Sensoren werden zusätzliche Daten erfasst. Dazu zählen die Bestrahlungsstärke [ $\frac{W}{m^2}$ ], die Außentemperatur [°C], die Modultemperatur [°C] und der berechnete Energieertrag [kWh].

### 3.2 Aufbau der Wetterstation der HTW Berlin

Die Wetterstation befindet sich auf dem Gebäude G, welches neben dem neuen FKI-Gebäude steht. Seit Ende 2010 befindet sich dort eine *TDL-14-Wetterstation* der Firma Thies mit integriertem Datenlogger. Die Sensoren sind an einem drei Meter hohen Mast montiert, damit sie nicht verschattet und frei angeströmt werden können. Sämtliche Messwerte werden einmal pro Minute abgefragt und in einer Datenbank gespeichert. Die Daten können auf einer Webseite<sup>2</sup> aufbereitet in Form von Graphen abgerufen werden. [QE]

Die Wetterstation verfügt über folgende Sensoren [QE]:

**Windgeschwindigkeit:** Ein Schalenkreuzanemometer misst die Windgeschwindigkeit in Metern pro Sekunde [ $\frac{m}{s}$ ].

**Windrichtung:** Die Windrichtung wird mit einer Windfahne in Grad [°] gemessen, wobei 0° Norden ist.

---

<sup>2</sup><http://wetter.htw-berlin.de>

### 3.3 Anforderungsanalyse

---

**Lufttemperatur:** Ein Hygro-Thermogebler misst die Temperatur der Luft in Grad Celsius [°C].

**Relative Luftfeuchtigkeit:** Ebenfalls mit dem Hygro-Thermogebler wird die relative Luftfeuchtigkeit in Prozent [%] bestimmt.

**Luftdruck:** Mit Hilfe eines Barogebler wird der absolute Luftdruck in Hektopascal [hPa] gemessen.

**Bestrahlungsstärke:** Ein thermisches Pyranometer misst die Bestrahlungsstärke der Sonne in Watt pro Quadratmeter [ $\frac{W}{m^2}$ ].

**Bestrahlungsstärke (Si Sensor):** Eine zweite Messung der Bestrahlungsstärke erfolgt mit einem Silizium-Pyranometer. Dieser ist nur für Forschungszwecke in der Wetterstation vorhanden. [Qua12]

**Niederschlag:** Der Niederschlag wird über einen Niederschlagsgeber in Millimetern [mm] erfasst.

**Beleuchtungsstärke:** Die Beleuchtungsstärke wird mit einem Helligkeitssensor in Kiolux [kLux] gemessen.

Zusätzlich wird zu jeder Messung das Datum und die aktuelle Uhrzeit gespeichert. Die ersten Messdaten sind vom 01.10.2010, womit Daten von etwas über zwei Jahren zur Verfügung stehen. Sowohl die aktuellen als auch die historischen Messdaten können im CSV-Format über eine *REST*-Schnittstelle abgefragt werden.

### 3.3 Anforderungsanalyse

Die folgenden Punkte sollen im Rahmen der Umsetzung des Prototyps auf jeden Fall erreicht werden:

- Ansteuerung von realen und virtuellen Sensoren
- Kombination der Sensordaten
- Darstellung unterschiedlicher Widgets
- Nutzung der LED-Fläche
- Berücksichtigung der Eigenheiten der Medienfassade
- Automatischer Wechsel der Widgets (zum Beispiel nach einer gewissen Zeit)
- Entwurf einer einheitlichen Schnittstelle für die Widgets und Sensoren
- Erstellung eines Algorithmus für die Prognose der Stromerzeugung des Gebäudes
- Implementierung des Widgets für die prognostizierte Stromerzeugung
- Realisierung weiterer ausgewählter Widgets (insgesamt mindestens drei Stück)
- Schnittstelle für die Konfiguration der Anwendung

### 3.4 Analyse und Bewertung verschiedener Sensordaten

---

Unter einem *Widget* wird eine Komponente des Programms verstanden, die in irgendeiner Art und Weise auf der Medienfassade visualisiert wird. Zum Beispiel ist die Darstellung der Wetterprognose oder der Tram-Abfahrtszeiten jeweils als ein Widget zu verstehen.

Optional soll geprüft werden, inwiefern eine Interaktion mit dem Dashboard möglich ist, und wenn realisierbar exemplarisch an einem Widget demonstriert werden. Die Abbildung 3.6 zeigt ein Use-Case-Diagramm für das Dashboard. In diesem wird deutlich, dass es nur zwei verschiedene Akteure gibt: Administrator und Nutzer. Lediglich der Administrator wird über spezielle Schnittstellen verfügen, über die er das Dashboard administrieren kann.

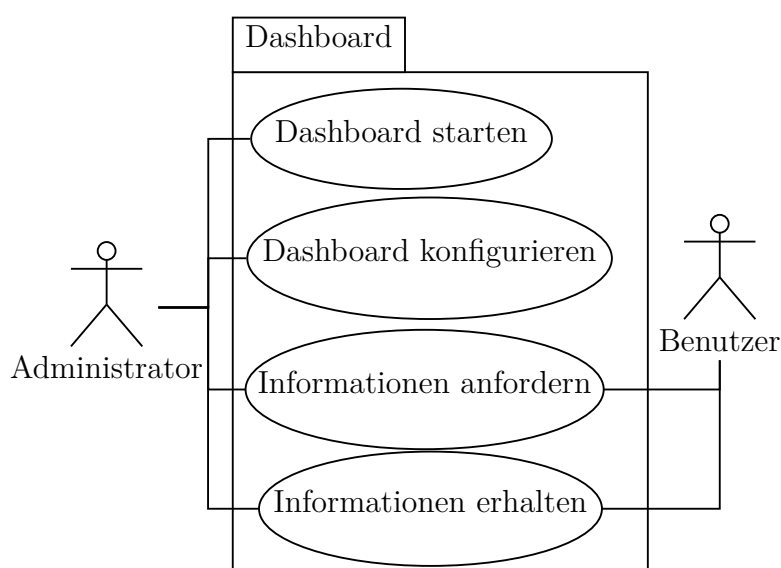


Abbildung 3.6: Use-Case-Diagramm für das Dashboard

## 3.4 Analyse und Bewertung verschiedener Sensordaten

Prinzipiell können viele unterschiedliche Sensoren zum Einsatz kommen. Zunächst soll eine Einteilung in reale und virtuelle Sensoren erfolgen. Darauf aufbauend erfolgt in Unterabschnitt 3.4.3 eine Analyse über eine mögliche Kombination der Sensordaten.

### 3.4.1 Mögliche reale Sensoren

Das Gebäude und die Umgebung bieten verschiedene Sensoren. Einerseits ist die Wetterstation auf dem Nebengebäude (siehe Abschnitt 3.2), andererseits die PV-Anlage des

### 3.4 Analyse und Bewertung verschiedener Sensordaten

---

FKI-Gebäudes (siehe Unterabschnitt 3.1.2) hervorzuheben. Die Medienfassade selbst könnte ebenso über verschiedene Sensoren verfügen:

**Mikrofon:** Es könnte ein Mikrofon installiert werden, damit die Umgebungsgeräusche wahrgenommen werden können. Im einfachsten Fall können Lautstärkeschwankungen registriert werden. Deutlich aufwändiger wäre eine Spracherkennung, aufgrund der vielen potenziellen Störgeräusche. Alternativ könnte auch das Mikrofon im Smartphone genutzt werden, wodurch mehrere Personen gleichzeitig mit der Fassade kommunizieren könnten. Die Erkennung der Sprache wäre hierbei ebenfalls einfacher zu realisieren.

**Kinect:** Des Weiteren könnte eine Kinect in die Fassade integriert werden, um Personen, die vor dieser stehen, zu verfolgen (tracken). Die Kinect ist ein Eingabegerät, welches dem Nutzer eine Steuerung mit Hilfe von Gesten ermöglicht. Das Gerät verfügt über eine Kamera, einen Infrarotsensor mit Chip zur Berechnung von Tiefenbildern und ein Array aus vier Mikrofonen. Der Vorteil bei der Nutzung einer Kinect ist, dass die Verfolgung von Personen und deren Bewegungen sehr gut funktioniert. Auch im Outdoor-Bereich bei Regen und wechselnder Bekleidung der Leute werden mit der Kinect gute Ergebnisse erzielt [BS12, Seite 45ff.]. Der Nachteil ist, dass die Kinect nur bei Nacht oder stark bewölktem Himmel eingesetzt werden kann, da sie mit Infrarotstrahlung arbeitet, die auch im Spektrum des Sonnenlichts enthalten ist.

**Kamera:** Für das Tracking kann ebenso eine Kamera verwendet werden. Im Gegensatz zu einer Kinect ist der Aufwand der Bildverarbeitung größer, dafür kann eine Kamera auch am Tag problemlos verwendet werden. Die Nutzung würde viele weitere Einsatzmöglichkeiten für die Medienfassade ermöglichen. Dabei müsste die Kamera voraussichtlich in einem Fenster platziert werden, da eine nachträgliche Montage in der Fassade sehr aufwändig wäre.

**RFID-Lesegerät:** Die Installation eines RFID-Lesegeräts wäre ebenso ein interessanter Aspekt. Dieses könnte für die Authentifizierung von Leuten über zum Beispiel die Mensakarte eingesetzt werden.

Neben diesen Sensoren, für welche die Umsetzung noch offen ist, gibt es noch weitere Sensoren im Gebäude, die auf jeden Fall installiert werden:

**Fahrstuhl:** Der Fahrstuhl des Gebäudes wird voraussichtlich mit zusätzlicher Hardware ausgerüstet sein. Er wird über einen Touchscreen für eine Interaktion verfügen, sowie eine Webcam für Bildmaterial aus dem Fahrstuhl. Des Weiteren wird es

möglich sein, den Fahrstuhl indirekt über eine Schnittstelle zu steuern. Dabei wird jedoch nur ein Drücken des Knopfs in der jeweiligen Etage simuliert und nicht direkt der Fahrstuhl gesteuert.

**Smart Metering:** Zusätzlich wird das Gebäude ein Smart Metering-System erhalten, mit dessen Hilfe der Stromverbrauch des gesamten Hauses gemessen und digital ausgelesen werden kann. Vermutlich wird es sich um einen EDL21-Zähler handeln. Eine detaillierte Auflistung für den Verbrauch pro Etage oder sogar pro Raum ist voraussichtlich nicht möglich.

#### 3.4.2 Mögliche virtuelle Sensoren

*Virtuelle Sensoren* stellen zusätzlich zu den bereits vorhandenen eine Möglichkeit dar, an Umgebungsinformationen zu kommen. Hiermit sind alle Arten von Informationen und Sensordaten gemeint, die sich über das Internet beziehen lassen. Die nachstehenden virtuellen Sensoren sind denkbar:

- Wetterprognose über einen Anbieter wie zum Beispiel Wetter.com (REST-Schnittstelle)
- Twitternachrichten (REST-Schnittstelle und OAuth)
- Welt- oder Lokalnachrichten (RSS-Feed)
- Speiseplan der Mensa (RSS-Feed)
- Wichtige Termine (iCalender)
- Verkehrsdaten: Öffentliche Verkehrsmittel, Staus, Parkplatzsituation usw.

Außerdem könnten weitere offene Daten verwendet werden. Für Berlin gibt es inzwischen eine ansehnliche Anzahl an Datensätzen, die frei zur Verfügung gestellt werden<sup>3</sup>.

#### 3.4.3 Analyse der Sensordaten auf Kombinationsmöglichkeiten

Die vorgestellten Sensoren geben eine Vielzahl an Informationen über das FKI-Gebäude und die nähere Umgebung. Die bloße Visualisierung dieser Daten bietet für die Personen vor der Medienfassade oder im Gebäude bereits einen Mehrwert. Besonders interessant erscheint jedoch die mögliche Kombination einiger dieser Daten, um neue Informationen und Prognosen zu erhalten.

Bei vielen der bisher erläuterten realen Sensoren steht noch nicht sicher fest, ob diese

---

<sup>3</sup>Siehe <http://daten.berlin.de/>



### 3.5 Analyse und Bewertung der Prognose der Stromerzeugung

---

wirklich zur Verfügung stehen werden. Der Fokus wird daher auf die Sensoren gerichtet, die entweder bereits vorhanden sind oder deren Installation definitiv erfolgen wird. Hierbei ist einerseits die Wetterstation zu nennen, die bereits seit zwei Jahren in Betrieb ist, andererseits die PV-Anlage, welche zum Teil bereits montiert ist, sowie das Smart Metering-System. Etwas anders sieht es bei den virtuellen Sensoren aus, deren Umsetzung nicht an einer physischen Installation sondern nur von dem Vorhandensein einer geeigneten Schnittstelle abhängig ist.

Mit den vorhandenen Sensoren sind zum Beispiel die nachstehenden Szenarien vorstellbar:

**Prognose der Stromerzeugung:** Unter Nutzung der Daten der PV-Anlage, der Wetterstation und einem virtuellen Wettersensor, der eine Wetterprognose liefert, kann eine Prognose für den erzeugten Strom durch die PV-Anlage für die nächste Zeit erstellt werden. Diese Prognose ermöglicht viele weitere interessante Schlussfolgerungen für das Gebäude (siehe Unterabschnitt 4.4.4).

**Verhältnis des erzeugten zu dem verbrauchten Strom:** Sobald der erzeugte Strom von der PV-Anlage gemessen werden kann, wäre eine Kombination mit den Daten des Smart Metering-Systems aufschlussreich. Damit könnte zum Beispiel ein Verhältnis der beiden Größen zueinander dargestellt werden.

## 3.5 Analyse und Bewertung der Prognose der Stromerzeugung

Ein wichtiger Bestandteil der Anwendung soll die Prognose des erzeugten Stroms der PV-Anlage sein. Damit lassen sich viele weitere Schlüsse für die Nutzung des Gebäudes ziehen (siehe Unterabschnitt 4.4.4). Die erzeugte Leistung ist von verschiedenen Faktoren abhängig (siehe Unterabschnitt 2.5.2), der wichtigste variable Faktor ist dabei die Bestrahlungsstärke auf die Module. Um eine Prognose für den erzeugten Strom zu einem zukünftigen Zeitpunkt zu ermöglichen, muss dementsprechend die Bestrahlungsstärke in Form der Globalstrahlung zu diesem Zeitpunkt prognostiziert werden.

### 3.5.1 Prognose der Globalstrahlung

In einem ersten Schritt soll eine Prognose der Globalstrahlung erfolgen. Als Basis für die dafür benötigten Daten können die historischen und aktuellen Werte der verschiedenen

### 3.5 Analyse und Bewertung der Prognose der Stromerzeugung

---

Sensoren verwendet werden. Besonders interessant sind dabei die historischen und aktuellen Wetterdaten der Wetterstation, die historischen, aktuellen und prognostizierten Daten eines virtuellen Wettersensors und die Daten der PV-Anlage.

Für die Prognose der Stromerzeugung wurden verschiedene Ansatzmöglichkeiten betrachtet:

#### **Prognose der Globalstrahlung mit Hilfe eines Neuronalen Netzes**

Eine Möglichkeit der Prognose ist die Nutzung eines *Neuronalen Netzes*. Besonders im Bereich der Temperaturprognose gibt es vermehrt Ansätze, ein Neuronales Netz zu erstellen und mit historischen Daten zu trainieren. Zum Beispiel haben Dr. S. Santhosh Baboo und I. Kadar Shereef für die Temperaturprognose für einen Tag bereits Ergebnisse erreicht, die mit einer numerischen Prognose vergleichbar sind (siehe [BS10]). Auch eine Prognose für die Globalstrahlung wurde bereits untersucht. M. A. Abdulazeez hat auf Basis von der Sonnenscheindauer, der maximalen Temperatur und der relativen Luftfeuchtigkeit gute Ergebnisse für die durchschnittliche Globalstrahlung in Nigeria für einen Monat erzielt (siehe [Abd11]). Ein monatlicher Durchschnittswert wäre für die Ziele dieser Arbeit noch nicht geeignet. Es wurde jedoch bereits eine Abhandlung von T. Krishnaiah, S. Srinivasa Rao, K. Madhumurthy und K. S. Reddy veröffentlicht, in der ein Neuronales Netz für die stündliche Prognose der Globalstrahlung in Indien erstellt wurde (siehe [KRMR07]). Die Ergebnisse waren nahe an den real gemessenen Werten. Es ist jedoch fraglich, inwiefern diese Modelle auf die meteorologischen Eigenheiten in Deutschland übertragen werden können. Die Arbeiten haben allerdings deutlich gezeigt, wie komplex und aufwändig die Erstellung eines solchen Neuronalen Netzes ist. Das Ergebnis wäre ungewiss und mit der Erstellung eines Neuronalen Netzes für die stündliche Prognose der Globalstrahlung in Deutschland könnte sehr schnell der Rahmen dieser Arbeit überschritten werden.

#### **Prognose der Globalstrahlung mit Hilfe von Data Mining**

Ebenso vorstellbar wäre eine Nutzung von Data Mining (siehe Unterabschnitt 2.2.1) für die Prognose der Globalstrahlung. Da historische Daten der Bestrahlungsstärke von der Wetterstation im Minutentakt aus den letzten beiden Jahren vorliegen, könnten diese als Datenbasis dienen. Es ist dabei jedoch völlig offen, ob die Daten der Wetterstation zu einer brauchbaren Prognose verwendbar sind. Der Aufwand und Nutzen ist dementsprechend schwer abzuschätzen.

### Prognose der Globalstrahlung mit Hilfe physikalischer Berechnungen

Eine weitere Möglichkeit wäre es, die Globalstrahlung mit Hilfe physikalischer Formeln zu berechnen. In Abschnitt 2.4 sind die dafür benötigten Zusammenhänge beschrieben. Vorteilhaft bei der Berechnung der Globalstrahlung ist, dass dies vergleichsweise exakte Werte als Ergebnis liefert und die Komplexität der Berechnung zwar nicht trivial jedoch überschaubar ist.

#### Ergebnis

Die Berechnung der Globalstrahlung ist die einzige der drei Methoden, bei der das Ergebnis nicht völlig offen ist. Ebenso ist sie der einzige Lösungsansatz, der eine Aufwandsabschätzung ermöglicht. Da der Aufwand im Rahmen dieser Arbeit erfüllbar erscheint und exakte Ergebnisse zu erwarten sind, wird für die Prognose der Globalstrahlung dieser Weg gewählt.

### 3.5.2 Prognose der Stromerzeugung

Sobald die Globalstrahlung korrekt berechnet worden ist, kann die Leistung der PV-Anlage bestimmt werden. Es muss allerdings beachtet werden, dass aufgrund unterschiedlicher Bewölkungsgrade die Globalstrahlung, die an den Solarzellen ankommt, nicht der berechneten entspricht. Da die Wetterstation der HTW Berlin leider nicht über einen Sensor für die Messung des Bewölkungsgrads verfügt, liegen keine historischen Daten vor. Es können jedoch historische Daten in Form eines Durchschnittswerts pro Tag vom Deutschen Wetterdienst sowie aktuelle Daten über einen virtuellen Sensor einer Wetter-Webseite bezogen werden. Eine korrekte physikalische Berechnung für diesen Zusammenhang ist äußerst komplex. Daher soll versucht werden, einen einfachen Zusammenhang über die historischen Daten zu zeigen.

Ebenso wird die Außentemperatur benötigt, da die Leistung der Solarmodule von deren Temperatur abhängt (siehe Unterabschnitt 2.5.2). Diese kann einfach über einen virtuellen Sensor einer Wetter-Webseite bezogen werden.

Sobald die Globalstrahlung in Abhängigkeit des Bewölkungsgrads und die Außentemperatur vorliegt, kann die Leistung der PV-Anlage berechnet werden.

## 3.6 Analyse der Visualisierungstechnologien einer Medienfassade

### 3.6.1 Visualisierung auf einer Medienfassade

Die Vielfalt der verschiedenen Technologien (siehe Unterabschnitt 2.1.3) zeigt, dass die Visualisierung auf einer Medienfassade stets von dem konkreten Aufbau und den verwendeten Technologien abhängt. Häufig werden mehrere Technologien aufgrund ihrer Eigenarten an einer Medienfassade kombiniert. Dies bietet besonders interessante und vielfältige Möglichkeiten der Nutzung.

Diejenige Person, die nun eine Anwendung oder Visualisierung für eine Medienfassade realisieren möchte, muss stets zuerst die Fassade genau betrachten und analysieren. Wichtig sind dabei Faktoren wie die Anzahl der Pixel, der Pixelabstand, die Realisierbarkeit von Farb- oder Videodarstellung und die Nutzungsmöglichkeit bei Tag oder Nacht. Aber auch die Position des Betrachters zur Medienfassade und die Kombinationsmöglichkeiten der Technologien sollten berücksichtigt werden.

### 3.6.2 Visualisierung auf der Medienfassade des FKI-Gebäudes

Die Medienfassade des FKI-Gebäudes verfügt über drei unterschiedliche Visualisierungstechnologien. Für die Darstellung der Anwendung soll in der ersten Version auf die Verwendung der Rückprojektion verzichtet werden. Dies hat den Hintergrund, dass die Anwendung als Informationstafel nur bei Tag, wenn viele Studenten auf dem Campus unterwegs sind, sinnvoll ist. Die Rückprojektion kann jedoch aufgrund der Nutzung der Räumlichkeiten am Tag nicht verwendet werden. Sowohl die LED-Wand als auch das Ambient-Light können hingegen tagsüber eingesetzt werden. Dabei soll der Fokus aufgrund der höheren Pixelanzahl auf der LED-Wand liegen. Diese kann zusätzlich mit einer hohen Helligkeit und Farbintensität auftrumpfen. Das Ambient-Light könnte unterstützend zum Einsatz kommen, problematisch ist jedoch, dass zum aktuellen Zeitpunkt noch keine Testmöglichkeit dafür zur Verfügung steht.

Bei der Entwicklung für die LED-Wand ist zu beachten, dass es zu erheblichen Unterschieden zwischen der Darstellung auf dem Entwicklungsgerät und der eigentlichen Fassade kommen kann. Problematisch sind dabei der unterschiedliche Pixelabstand zwischen einem Monitor und der LED-Wand sowie die Hardwareabhängigkeit des RGB-Farbraums (siehe Unterabschnitt 2.1.1). Beispielsweise kann ein in diesem Farbraum definiertes Rot auf dem Monitor des Entwicklers ganz anders wirken als auf den LED-

### ***3.6 Analyse der Visualisierungstechnologien einer Medienfassade***

---

Kacheln. Zum Testen der Visualisierung steht daher ein Testlabor zur Verfügung (siehe Unterabschnitt 6.1.1). Mit Hilfe des Testlabors kann geprüft werden, wie die Visualisierung zumindest annähernd auf der Fassade aussehen wird.

# 4 Entwurf

In diesem Kapitel wird der Entwurf für die Anwendung detailliert vorgestellt. Zunächst erfolgt eine Beschreibung der Systemarchitektur und darauf aufbauend eine Modellierung der Anwendung. Dabei wird ausführlich auf den Entwurf der einzelnen Widgets eingegangen. Zum Abschluss wird noch der komplexeste Algorithmus der Anwendung, die Prognose des Stroms, erläutert.

## 4.1 Entwurf der Systemarchitektur

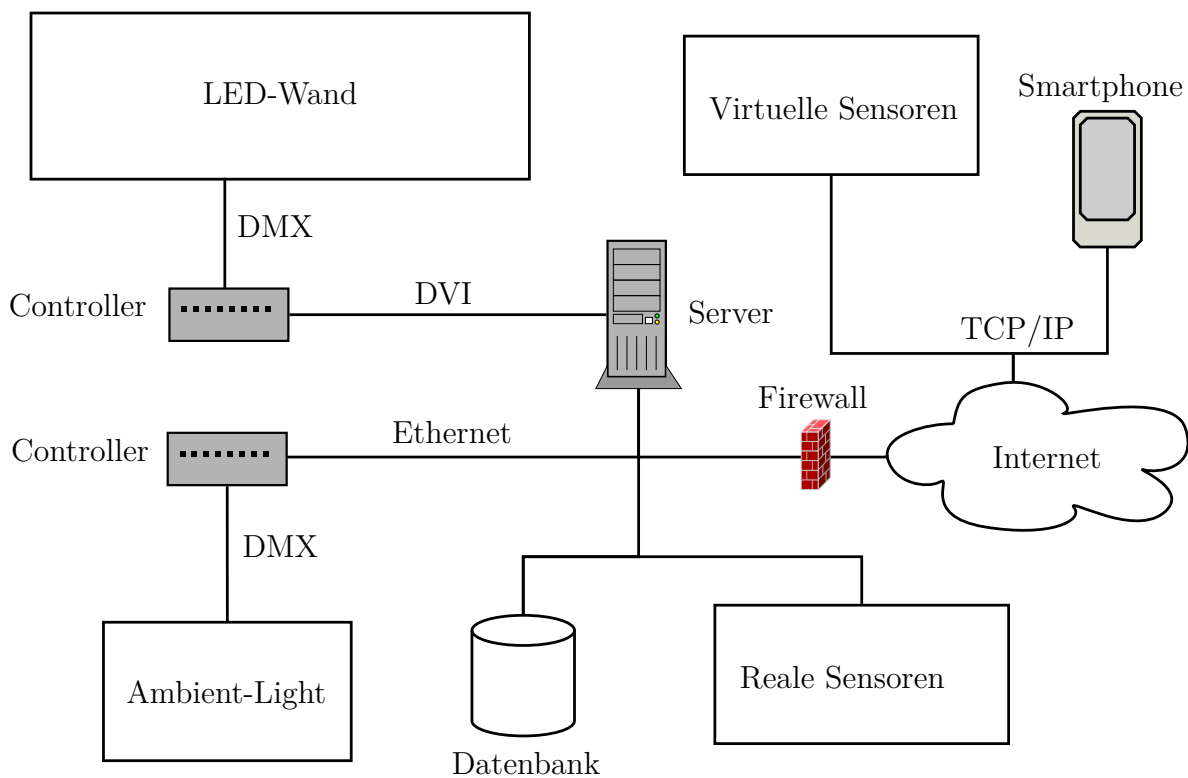


Abbildung 4.1: Systemarchitektur der Anwendung

In der Abbildung 4.1 ist die Systemarchitektur dargestellt. Diese beinhaltet einen zentralen Server, auf dem die Dashboard-Anwendung und eine Art Kommunikationsserver

läuft. Dieser Kommunikationsserver regelt den Datenaustausch mit den mobilen Geräten, die zur Interaktion mit der Anwendung verwendet werden können (siehe Unterabschnitt 4.3.4). Die Visualisierung erfolgt auf der LED-Wand über einen Controller. Dieser wird per DVI angesteuert und verteilt wiederum per DMX die Daten an die einzelnen LED-Module. Das Ambient-Light wird voraussichtlich ebenfalls über einen Controller verfügen, der die LED-Leisten des Ambient-Lights genauso per DMX steuert. Für die Datenhaltung wird eine bereits existierende *postgreSQL-Datenbank* verwendet, in welche die Daten der realen Sensoren gespeichert werden. Dies beinhaltet zurzeit die Daten der Wetterstation und wird in Zukunft auch die Daten der PV-Anlage und des Smart Metering-Systems einbeziehen. Die virtuellen Sensoren sind über das Internet durch verschiedene auf TCP/IP aufbauende Protokolle eingebunden.

## 4.2 Modellierung des Dashboards

Das Programm wird über verschiedene Komponenten verfügen, wobei ein Kernelement die Widgets sein werden. Zudem wird es den *Widget-Launcher* geben, der die Widgets verwaltet und startet beziehungsweise beendet.

Die Zusammenhänge der wichtigsten Komponenten sind in der Abbildung 4.2 dargestellt. Der Widget-Launcher wird  $n$  Widgets verwalten. Die verschiedenen Widgets verfügen über eine gemeinsame abstrakte Elternklasse, damit sie eine einheitliche Schnittstelle haben. Außerdem werden sie über einen oder mehrere Sensoren verfügen, über die sie mit Daten versorgt werden. Die Sensoren werden ebenfalls eine abstrakte Elternklasse haben, damit auch hier eine gemeinsame Schnittstelle vorhanden ist.

### 4.2.1 Entwurf der Oberfläche des Dashboards

Die vorhandene Visualisierungsfläche wird unter den Widgets aufgeteilt. Dabei können zwei Widgets nebeneinander positioniert werden, wie in Abbildung 4.3 dargestellt. Da die Oberfläche ein Format von 3 : 1 hat, ist die Aufteilung nebeneinander am sinnvollsten. Hierbei hätte jedes Widget eine Pixelanzahl von 142 x 96 Pixeln zur Verfügung, wenn noch ein Trennbalken von vier Pixeln Breite zwischen den Widgets eingeplant wird. Bei dem Entwurf der Widgets sollte aufgrund der geringen Pixelanzahl darauf geachtet werden, dass nicht zu viel Text verwendet wird. Gut lesbare Schrift sollte mindestens 10 x 10 Pixel pro Zeichen aufweisen. Da der Betrachter etwas weiter von der Fassade entfernt steht, sollte die Schriftgröße in einer angemessenen Größe gewählt wer-

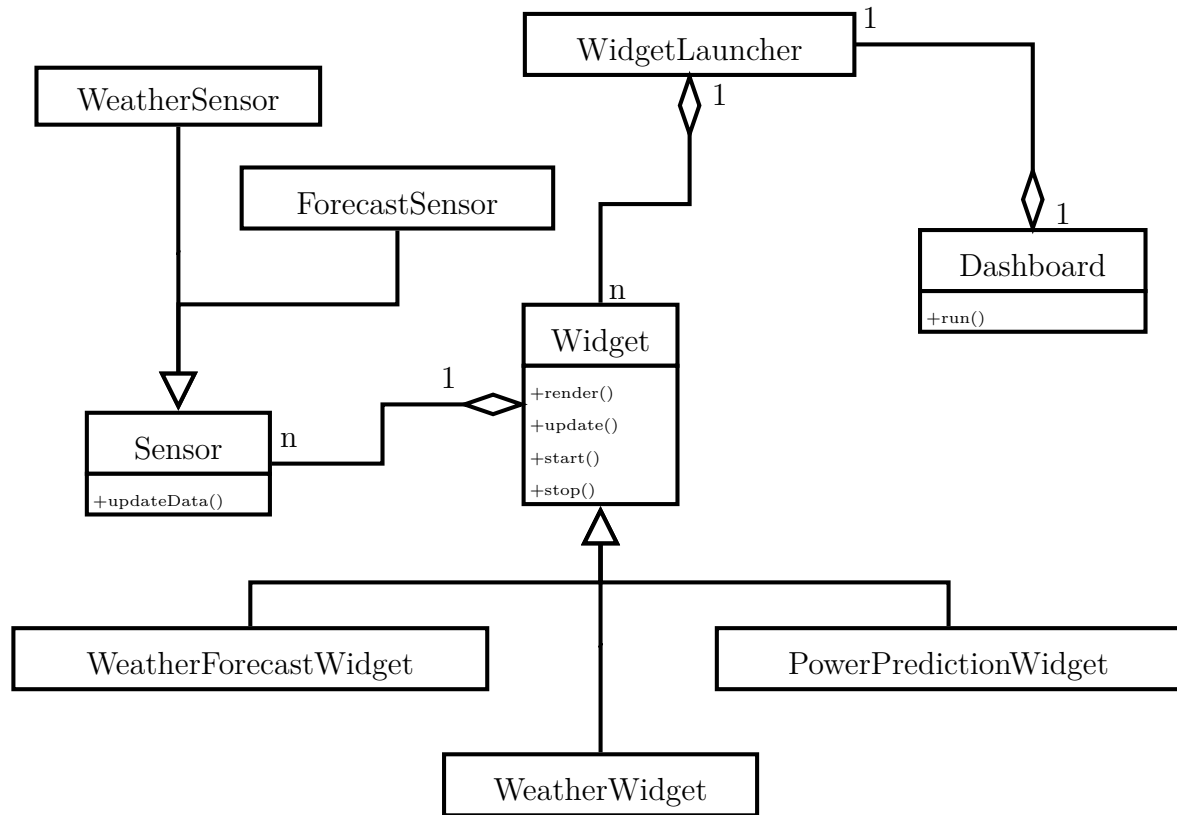


Abbildung 4.2: Auszug des Klassendiagramms der Anwendung

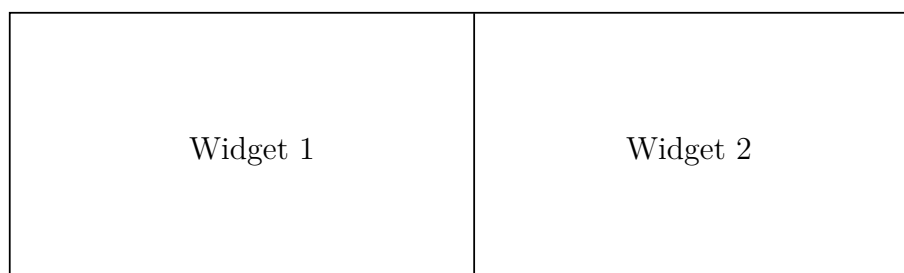


Abbildung 4.3: Aufteilung der Fläche der LED-Wand für zwei Widgets



## 4.3 Entwurf der Widgets

---

den, weshalb es empfehlenswert ist, mehr als 10 x 10 Pixel pro Zeichen zu verwenden. Mit dem benötigten Abstand zwischen den Schriftzeichen sollte ein Widget daher nicht mehr als sechs bis sieben Zeilen Text haben. Diese Annahme konnte im Testlabor (siehe Unterabschnitt 6.1.1) bestätigt werden. Dementsprechend ist es eine Herausforderung beim Entwurf der Widgets, die verfügbaren Informationen in der geringen Fläche und Pixelanzahl unterzubringen.

Insgesamt ist eine Verwendung von Symbolik empfehlenswert, da der Nutzer dadurch die Informationen leichter und aus größerer Entfernung aufnehmen kann. Jedoch muss aufgrund der geringen Pixelanzahl dabei ebenso darauf geachtet werden, dass ein Widget nicht mit zu vielen Informationen, die als Folge vielleicht nicht mehr gut zu erkennen sind, überladen wird.

Da es deutlich mehr als zwei Widgets geben soll, werden diese wechselnd angezeigt. Eine Darstellung aller Widgets gleichzeitig ist sowohl aus Platz- als auch Informationsgründen nicht sinnvoll. Den zeitlichen Wechsel der Widgets wird der Widget-Launcher (siehe Abschnitt 5.2) realisieren.

## 4.3 Entwurf der Widgets

Für die Umsetzung des Dashboards sind viele unterschiedliche Widgets vorstellbar. Im Folgenden sollen einige näher vorgestellt werden.

### 4.3.1 Widgets für die PV-Anlage

Ein Kernelement der Anwendung ist die Prognose der Stromerzeugung der PV-Anlage. Dementsprechend wird es auch eine Visualisierung für die berechneten Daten geben. Die nachstehenden Informationen könnten für den Betrachter interessant sein:

- Die Prognose für den erzeugten Strom für den aktuellen Tag, in mehrere Zeiträume aufgeteilt.
- Der Gesamtstrom, der voraussichtlich für die folgenden Tage erzeugt wird.
- Der verbrauchte Strom im Verhältnis zu dem aktuell erzeugten Strom.
- Die Sonnenposition oder die Globalstrahlung zu einem gewissen Zeitpunkt.

Aufgrund der Vielfalt der Informationen ist es sinnvoll, den Inhalt auf mehrere Widgets zu verteilen. Der Fokus soll dabei auf den Widgets für die Prognose der Stromerzeugung und der Berechnung der Sonnenposition liegen, die im Folgenden beschrieben werden.

#### Prognose der Stromerzeugung der PV-Anlage für den aktuellen Tag

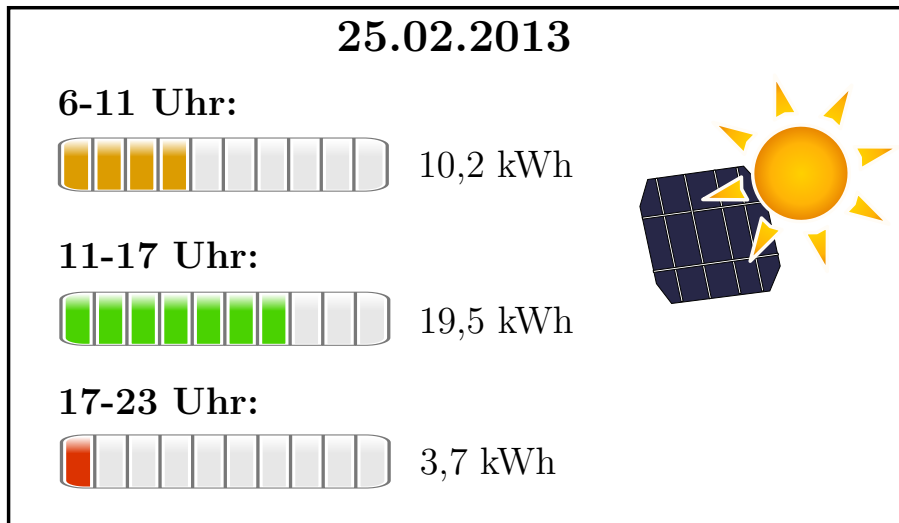


Abbildung 4.4: Entwurf des Widgets für den prognostizierten Strom der PV-Anlage

In Abbildung 4.4 ist ein Entwurf für die Darstellung des erzeugten Stroms der PV-Anlage für einen gewissen Zeitraum des Tages dargestellt. Die Zeiträume sind dabei so gewählt, dass sie mit den Prognosedaten fürs Wetter übereinstimmen, damit eine bessere Zuverlässigkeit der Aussage erreicht werden kann. Eine zusätzliche Visualisierung der erzeugten Leistung in Form eines Balkens wurde gewählt, damit der Betrachter sofort weiß, ob viel oder wenig Strom erzeugt wird. Eine reine Darstellung als Zahl würde dem unerfahrenen Betrachter keine Aussage diesbezüglich liefern. Ein zu 100 % gefüllter Balken entspricht der Leistung, die sich aus der höchsten gemessenen Strahlungsstärke der letzten zwei Jahre ergibt (siehe Unterabschnitt 4.4.3). Ein Farbwechsel von Rot über Gelb nach Grün verstärkt den visuellen Effekt des Balkens. Da die Wetterprognosedaten für drei Tage in die Zukunft verfügbar sind, wird die Anzeige der Prognose für drei Tage wechselnd dargestellt werden. Die Informationen für mehrere Tage sind sonst nicht praktikabel in der gegebenen Bildschirmgröße unterzubringen.

Für die Umsetzung des Widgets wäre ein virtueller Sensor für die Wetterprognose notwendig, um die Leistung der PV-Anlage korrekt zu bestimmen. Es werden einerseits die Umgebungstemperatur, andererseits der Wetterzustand benötigt. Es wird im Folgenden zwischen diesen Wetterzuständen unterschieden: sonnig, leicht bewölkt, bewölkt, bedeckt, leichter Regen, starker Regen, Schnee und Sturm. Mit diesen Daten kann die prognostizierte Leistung zu einem gewissen Zeitpunkt bestimmt werden. Die Berechnung muss für mehrere Zeitpunkte in der gegebenen Zeitspanne bestimmt und auf die

### 4.3 Entwurf der Widgets

---

komplette Zeitspanne aufsummiert werden. Dabei muss berücksichtigt werden, dass die PV-Anlage über drei Seiten verfügt, die einzeln berechnet werden müssen. Des Weiteren muss der maximale Wert für die PV-Anlage aus den historischen Daten bestimmt werden, damit die berechnete Leistung ins Verhältnis zu der maximalen Leistung als Balken visualisiert werden kann.

#### Berechnung der Sonnenposition zu einem gegebenen Zeitpunkt

Ein weiterer interessanter Aspekt könnte die Berechnung der Sonnenposition zu einem gegebenen Zeitpunkt sein. Hierbei wäre es jedoch notwendig, dass der Nutzer diesen Zeitpunkt auswählen kann. Dies setzt eine Interaktionsmöglichkeit voraus. Die Abbildung 4.5 zeigt einen Entwurf für dieses Widget. Es werden die wichtigsten Daten angezeigt: der aktuelle Zeitpunkt, der Sonnenhöhenwinkel und der Sonnenazimutwinkel. Das abgebildete Smartphone unten rechts in der Ecke soll dem Benutzer visualisieren, dass er mit diesem Widget interagieren kann (mehr dazu in Unterabschnitt 4.3.4). Das Eingabegerät des Nutzers benötigt für die Realisierung dieses Widgets lediglich die Möglichkeit ein Datum und eine Uhrzeit einzugeben. Dieses Widget wird im Prototyp das einzige mit einer Interaktionsmöglichkeit sein.

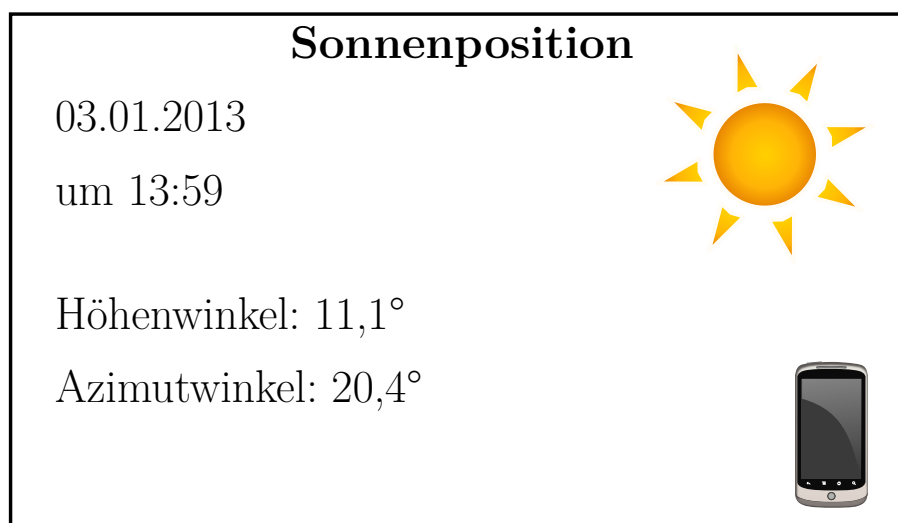


Abbildung 4.5: Entwurf des Widgets für die Sonnenposition zu einer bestimmten Zeit

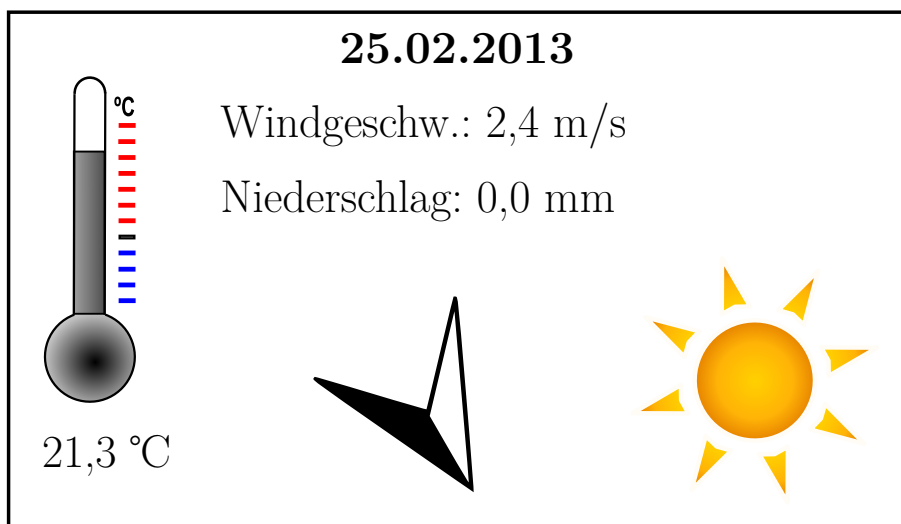
#### 4.3.2 Widgets für das Wetter

Beim Wetter sind insbesondere das aktuelle und das zukünftige Wetter interessant für den Betrachter, die im Folgenden erläutert werden.

#### Aktuelles Wetter

Das aktuelle Wetter kann über die Wetterstation auf dem Nebengebäude präzise bestimmt werden (siehe Abschnitt 3.2). Da kein direkter Zugriff auf die Datenbank der Wetterstation möglich ist, werden diese Daten im CSV-Format über eine REST-Schnittstelle bezogen und in einer lokalen PostgreSQL-Datenbank gespeichert. Die historischen Daten wurden mit Hilfe eines Python-Skripts aus mehreren CSV-Dateien in die Datenbank übertragen.

Bei der Visualisierung könnten sämtliche Sensoren der Wetterstation verwendet werden, für die meisten Betrachter ist jedoch nur ein kleiner Teil der Daten interessant. Das sind insbesondere die Temperatur, die Windgeschwindigkeit und Windrichtung, der Niederschlag und der Wetterzustand. In Abbildung 4.6 ist ein Entwurf des Widgets zu sehen, in dem diese Werte der Wetterstation dargestellt sind. Der Balken des Thermometers wird sich der realen Temperatur anpassen, wobei ein Schritt 5 °C entspricht. Die Windrichtung wird variabel mit einem Pfeil visualisiert, der sich an den Himmelsrichtungen orientiert. Der Zustand des Wetters wird von der Wetterstation nicht gemessen, sondern müsste berechnet werden. Da eine korrekte Bestimmung mit den vorhandenen Sensoren aufwändig wäre, wird auch an dieser Stelle ein weiterer Sensor genutzt. Mit Hilfe eines virtuellen Sensors für eine Wetterprognose kann auch für den aktuellen Zeitpunkt der gegenwärtige Wetterzustand ermittelt werden. Das Ergebnis wird über eine Symbolik in der unteren rechten Ecke des Widgets visualisiert (siehe Abbildung 4.7).



**Abbildung 4.6:** Entwurf des Widgets für die aktuellen Wetterdaten der Wetterstation

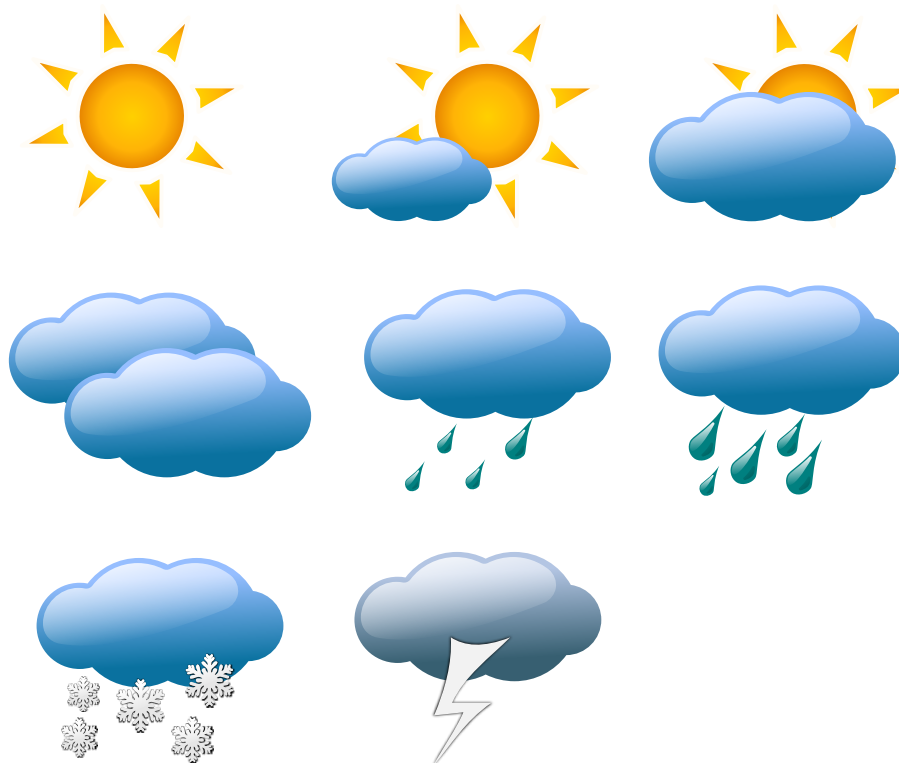


Abbildung 4.7: Die verschiedenen Symbole für den Wetterzustand

#### Wetterprognose

Neben der Darstellung des aktuellen Wetters ist natürlich auch die Visualisierung einer Wetterprognose denkbar. Dafür wäre ein virtueller Sensor in der Form notwendig, dass über diesen die lokale Wetterprognose bezogen werden kann. Eine Prognose aus den Daten der Wetterstation der Hochschule ist zwar prinzipiell ebenso vorstellbar, allerdings nur mit erheblichem Aufwand und dem dafür nötigen Fachwissen realisierbar. Zusätzlich sind für eine korrekte Wetterprognose nicht nur die lokalen sondern auch die regionalen und überregionalen Wetterdaten notwendig, die wiederum beschafft werden müssten (siehe Unterabschnitt 2.3.2).

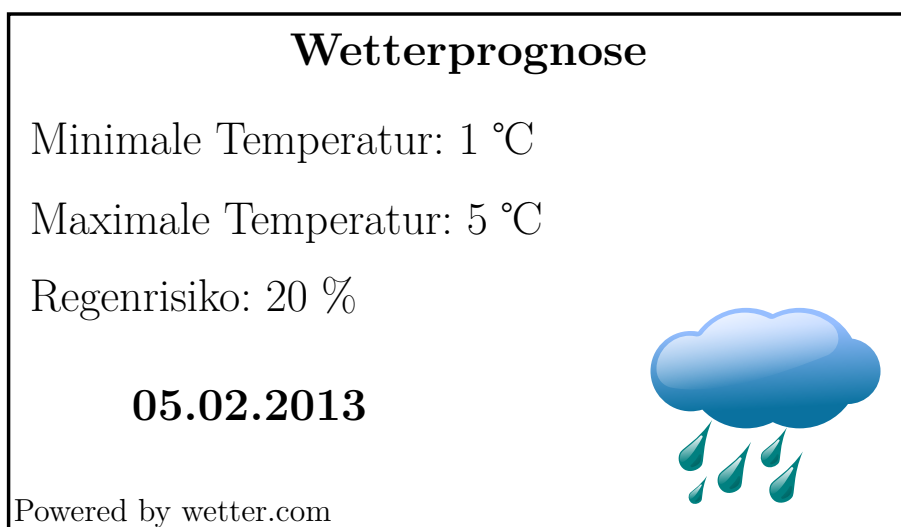
Praktikabler ist es, wenn die Wetterprognose von Experten erledigt wird und diese Daten über einen virtuellen Sensor beschafft werden. Verschiedene Anbieter stellen ihre Wetterprognosen über eine API zur Verfügung. Hier sind beispielsweise *wetter.com*, *weather.com* oder *wetter24.de* zu nennen. Die Daten werden meistens über eine REST-Schnittstelle zur Verfügung gestellt. *Wetter.com* ist von diesen jedoch der einzige Anbieter, der die Daten kostenfrei anbietet. In diesem Fall ist zwar eine Limitierung von 10.000 Zugriffen im Monat gegeben, dies reicht jedoch für die Zwecke des Widgets

### 4.3 Entwurf der Widgets

---

vollkommen aus. Des Weiteren muss ein Hinweis vorhanden sein, dass die Daten von diesem Anbieter stammen. Die Daten werden über eine REST-Schnittstelle wahlweise im *JSON*- oder *XML*-Format angeboten. Da sie im weiteren Verlauf nur maschinell ausgewertet werden sollen, ist das *JSON*-Format aufgrund des tendenziell geringeren Datenaufkommens und des schnelleren Parsens vorzuziehen.

Die Abbildung 4.8 zeigt den Entwurf des Widgets für die Wettervorhersage. Bei diesem sind die wichtigsten Daten die Temperatur, die Regenwahrscheinlichkeit und der Wetterzustand visualisiert und werden wechselnd für drei Tage angezeigt.



**Abbildung 4.8:** Entwurf des Widgets für die Wettervorhersage

#### 4.3.3 Weitere Widgets

Neben den beiden vorgestellten Widget-Gruppen sind noch weitere Widgets denkbar, die nachstehend beschrieben werden.

##### **Abfahrtszeiten der öffentlichen Verkehrsmittel**

Direkt vor dem Campus der HTW Berlin befindet sich eine Haltestelle verschiedener Tram- und Nachtbuslinien, wodurch der Campus über die öffentlichen Verkehrsmittel erreichbar ist. Die Abfahrtszeiten dieser Linien sind für die Personen auf dem Campus interessant und werden häufig individuell über das Internet abgefragt. Dementsprechend wäre auch die Visualisierung dieser Daten auf der Medienfassade vorteilhaft.

Die Berliner Verkehrsbetriebe (BVG) bieten die Abfahrtszeiten über ihre Webseite an, wobei sowohl Live- als auch Plan-Daten abgefragt werden können. Nach Rücksprache

### 4.3 Entwurf der Widgets

---

mit der BVG hat diese keine Einwände bekundet, die Daten auf der Medienfassade zu veröffentlichen und zu visualisieren. Besonders interessant wäre eine Visualisierung der Live-Daten für die verschiedenen Linien in beide Fahrtrichtungen, wie sie skizzenhaft in Abbildung 4.9 dargestellt ist. Dabei sind die wichtigsten Informationen so visualisiert, dass sie auch auf eine größere Entfernung noch gut erkennbar sind. Dies ist sinnvoll, damit die Studenten sich nicht erst in die Nähe der Medienfassade begeben müssen, welche sich von der Richtung her für die meisten Studenten gegensätzlich zur Haltestelle befindet. Daher werden die Abfahrtszeiten besonders groß dargestellt und die Nummer der Linie in einem Farbkontrast von Gelb auf Schwarz visualisiert.

Eine passende Schnittstelle für die Live-Daten wird jedoch erst im ersten oder zweiten Quartal 2013 zur Verfügung stehen. Alternativ könnten die Daten der Webseite geparkt werden, welches allerdings mit einem vergleichsweise hohen Aufwand verbunden wäre, der aufgrund der baldigen Verfügbarkeit der Schnittstelle nicht gerechtfertigt scheint.

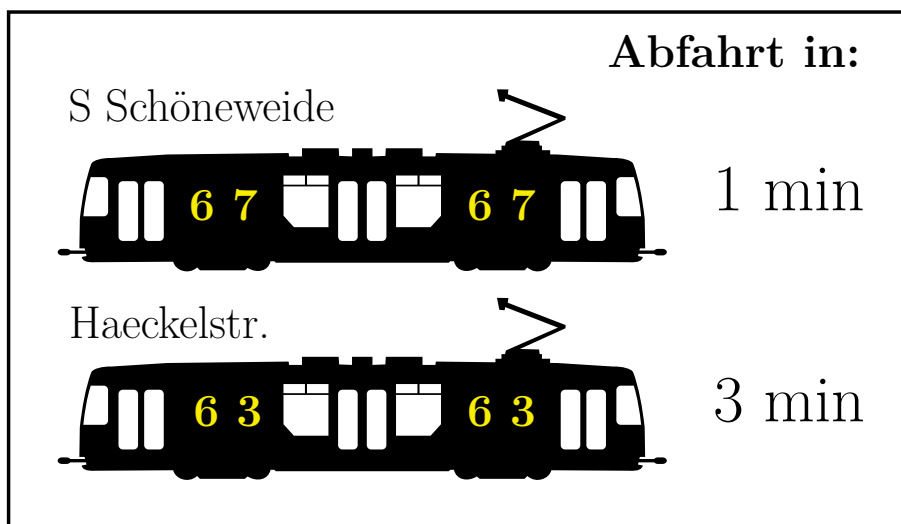


Abbildung 4.9: Skizze des Widgets für die Abfahrtszeiten der öffentlichen Verkehrsmittel

#### Twitter-Wall

Sogenannte *Twitter-Walls* sind Displays, auf denen Tweets zu einem bestimmten Thema vereint werden. Besonders auf Konferenzen sind diese beliebt. Das Konzept könnte auch gut auf ein Widget für das Dashboard übertragen werden. Wie in Abbildung 4.10 skizziert, könnten Nachrichten an einen speziellen Nutzer, in diesem Fall der Nutzer *FKI*, visualisiert werden. Für die Verwendung an der Medienfassade würde hierfür ein spezieller Twitter-Account angelegt werden.

### 4.3 Entwurf der Widgets

---

Twitter verfügt über eine REST-API, mit der die Daten zu einem Benutzer oder *Hashtag* abgefragt werden können. Ein Hashtag wird verwendet, um ein Schlüsselwort oder Thema in einer Nachricht hervorzuheben. Für die Abfrage muss allerdings erst eine Autorisierung des Benutzers erfolgen. Bei Twitter ist dies über *OAuth* möglich, welches ein offenes Protokoll für die einfache, standardisierte und sichere Autorisierung für Web-, Mobil- und Desktopanwendungen ist [OAu]. Für die Realisierung gibt es verschiedene Twitter-Bibliotheken für C++, die dem Entwickler die Arbeit erleichtern. Hier wären zum Beispiel `Twitcurl`<sup>1</sup>, `libOAuth`<sup>2</sup> oder `QTweetLib`<sup>3</sup> zu nennen.

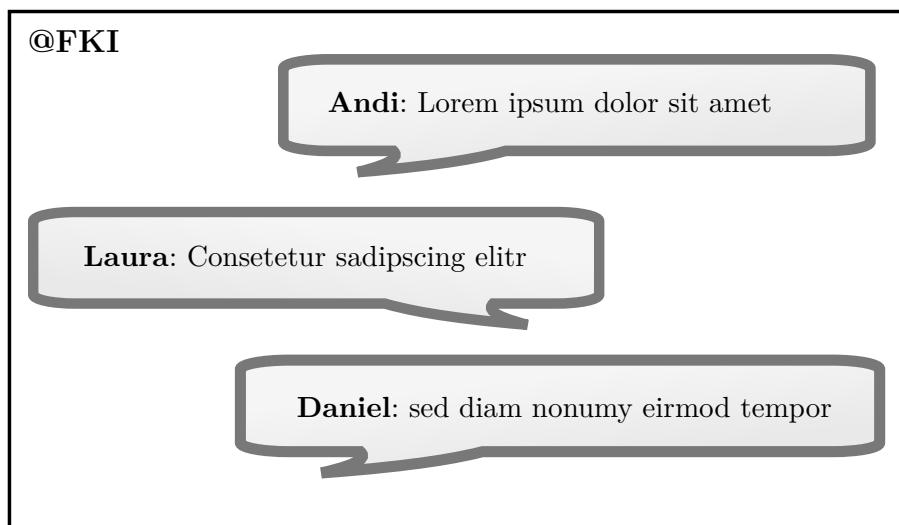


Abbildung 4.10: Skizze für das Twitter-Widget

#### Sonstige Widgets

Neben den bereits detailliert beschriebenen sind noch weitere Widgets vorstellbar. Sinnvoll könnte eine Anzeige des aktuellen Speiseplans der Mensa sein. Die hierfür benötigten Daten stehen per RSS-Feed zur Verfügung.

Außerdem wäre auch die Darstellung wichtiger Hochschultermine für die Studenten interessant. Zum Beispiel könnten Hochschulwahlen oder Anmeldetermine für die Kurse oder Prüfungen angezeigt werden. Als Schnittstelle wäre beispielsweise iCalendar verwendbar.

---

<sup>1</sup><http://code.google.com/p/twitcurl/>

<sup>2</sup><http://liboauth.sourceforge.net>

<sup>3</sup><https://github.com/minimoog/QTweetLib/>



### 4.3 Entwurf der Widgets

---

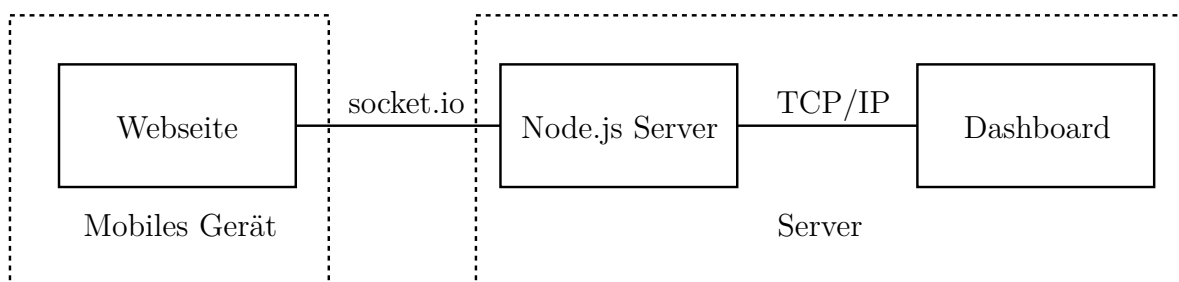
Außerdem könnten Widgets für eine Unwetterwarnung oder einen Nachrichtenticker erstellt werden. Sowohl Welt- als auch Lokalnachrichten könnten per RSS-Feed bezogen werden.

#### 4.3.4 Entwurf der Interaktion

Es gibt viele unterschiedliche Möglichkeiten, eine Interaktion mit einer Medienfassade zu realisieren. Der Benutzer kann das Eingabegerät mitbringen, zum Beispiel in Form eines Smartphones, das Eingabegerät kann an der Fassade montiert sein, beispielsweise in Form eines Terminals, oder die Eingabe kann ohne zusätzliches Eingabegerät erfolgen, wie zum Beispiel mit Hilfe von Tracking. Eine ausführliche Auflistung der verschiedenen Eingabemöglichkeiten ist in der Arbeit „Medienfassaden - Möglichkeiten von Visualisierung und Interaktion“ [BS12, Seite 29 ff.] zu finden.

Besonders das Smartphone bzw. das Tablet bieten viele Vorteile. Smartphones sind inzwischen sehr weit verbreitet, besitzen eine gute Eingabemöglichkeit, sind programmierbar und verfügen über viele weitere Sensoren. Insbesondere die präzise und schnelle Eingabemöglichkeit ist auch für die Interaktion mit dem Dashboard vorteilhaft. Hierfür wird eine spezielle Client-Anwendung entwickelt werden.

Damit eine möglichst große Anzahl an Menschen mit der Fassade interagieren kann, ist es notwendig, dass die Client-Anwendung auf vielen unterschiedlichen Plattformen läuft. Würde eine native App entwickelt werden, würde diese nur auf einem der Smartphone-Betriebssysteme laufen. Um dieses Problem zu umgehen, wird eine HTML5-Webseite erstellt werden, die für mobile Geräte optimiert ist. So wird nur ein HTML5-kompatibler Browser auf dem Gerät benötigt. Solche Browser sind inzwischen sehr weit verbreitet. Mit Hilfe von *Websockets* kann über eine Webseite eine TCP/IP-Verbindung zu einem Server aufgebaut werden, die ohne größere Verzögerung funktioniert. Leider ist die Un-



**Abbildung 4.11:** Visualisierung der Kommunikation zwischen dem mobilen Client und der Dashboard-Anwendung

terstützung von Websockets immer noch nicht in allen mobilen Browsern verfügbar<sup>4</sup>. Um diese Problematik zu beheben, wird ein Node.js-Server mit dem socket.io Plugin verwendet (siehe Unterabschnitt 5.7.8).

In der Abbildung 4.11 ist die Kommunikation zwischen dem mobilen Endgerät und der Dashboard-Anwendung visualisiert. Der Benutzer ruft mit seinem mobilen Endgerät die Webseite auf. Von dieser wird eine socket.io-Verbindung zum Node.js-Server aufgebaut. Socket.io hat dabei den großen Vorteil, dass es automatisch die bestmögliche Verbindungsart auswählt. Sollten zum Beispiel keine Websockets verfügbar sein, wird geprüft, ob *Flash-Sockets* zur Verfügung stehen. Insgesamt kennt socket.io sechs unterschiedliche Arten eines Kommunikationsaufbaus, wodurch eine sehr große Anzahl an Browsern unterstützt wird.

Der Node.js-Server nimmt die Daten vom Web-Client entgegen und sendet sie weiter an das Dashboard über eine normale Socket-Verbindung. Er dient dementsprechend als Mittelsmann, der die Daten einfach nur weiterleitet.

## 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

Das Widget für die Prognose der Stromerzeugung (siehe Unterabschnitt 4.3.1) benötigt einen komplexen Algorithmus zur Berechnung der Ergebnisse. Die Gleichungen aus dem Unterabschnitt 2.4.2 und dem Unterabschnitt 2.4.3 sind die Grundlage, damit eine Prognose für den erzeugten Solarstrom erstellt werden kann.

### 4.4.1 Berechnung der Globalstrahlung auf die PV-Anlage zu einem bestimmten Zeitpunkt

Zuerst muss die Globalstrahlung zu dem gegebenen Zeitpunkt bestimmt werden. Folgende Werte müssen hierfür vorhanden sein:

1. Das Datum und die Uhrzeit sowie daraus berechnet der aktuelle Tag des Jahres ( $J$ ) und der Monat ( $M$ ).
2. Der Längen- ( $\varphi_{Lokal}$ ) und Breitengrad ( $b$ ) des Ortes sowie der dazugehörigen Längengrad des Standardmeridians der Zeitzone ( $\varphi_{St}$ ).
3. Der Neigungswinkel ( $\gamma_E$ ) und der Azimutwinkel ( $a$ ) der Solarmodule.

---

<sup>4</sup>Der aktuelle Stand kann unter <http://caniuse.com/websockets> abgerufen werden.

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

Sobald alle diese Werte vorliegen, sind die nachstehenden Schritte notwendig, um die Globalstrahlung zu bestimmen:

1. Im ersten Schritt muss die Sonnenposition ermittelt werden:
  - a) Dabei wird zuerst die wahre Sonnenzeit ( $\sigma$ ) ermittelt.
  - b) Danach kann der Stundenwinkel ( $\tau$ ) und die Deklination ( $\delta$ ) bestimmt werden.
  - c) Mit Hilfe dieser beiden Werte kann der Sonnenhöhenwinkel ( $\gamma_S$ ) errechnet werden.
  - d) Darauf aufbauend wird der Sonnenazimutwinkel ( $a_S$ ) berechnet.
  - e) Zuletzt wird noch der Einfallswinkel ( $\psi$ ) ermittelt.
2. Mit Hilfe der Sonnenposition kann als nächstes die direkte Strahlung ausgerechnet werden:
  - a) Zuerst wird die direkte Strahlung ( $E_D$ ) berechnet.
  - b) Danach kann die direkte Strahlung auf eine horizontale Fläche bestimmt werden ( $E_{D,h}$ ).
  - c) Mit Hilfe des Ergebnisses kann die direkte Strahlung auf eine geneigte Fläche ( $E_{D,g}$ ), wie zum Beispiel dem Solarmodul, ermittelt werden.
3. Berechnung der diffusen Strahlung:
  - a) Als erstes wird die diffuse Strahlung auf eine horizontale Fläche bestimmt ( $E_{H,h}$ ).
  - b) Mit dem Ergebnis wird die diffuse Strahlung auf eine beliebig geneigte Fläche errechnet ( $E_{H,g}$ ).
4. Berechnung der reflektierten Strahlung:
  - a) Zunächst muss eine Abschätzung der Bodenreflexion ( $\rho_B$ ) erfolgen.
  - b) Danach kann mit der diffusen und direkten Strahlung auf eine Horizontale die reflektierte Strahlung ( $G_R$ ) berechnet werden. Hier wird nicht zwischen einer horizontalen und geneigten Fläche unterschieden, da ein isotroper Ansatz zur Berechnung der reflektierten Strahlung gewählt wurde.
5. Nachdem alle Teilstrahlungen bekannt sind, kann die Globalstrahlung ( $E_{G,g}$ ) auf die geneigten Solarmodule bestimmt werden mit:  $E_{G,g} = E_{D,g} + E_{H,g} + E_R$
6. Zum Abschluss wird noch der Einfluss des Wetters verrechnet.

Diese Abfolge von Berechnungen muss für jede der drei Seiten der PV-Anlage durchlaufen werden, da der Azimutwinkel der Solarmodule für jede Seite unterschiedlich ist.

##### Abschätzung der Bodenreflexion

Die Bodenreflexion lässt sich aufgrund der vielfältigen Bodenarten in der Umgebung des Gebäudes nur abschätzen. Das FKI-Gebäude liegt direkt an der Spree und ist ansonsten von Wohn- bzw. Industriegebiet umgeben. Für den Fluss wird der Bodenreflexionswert  $\rho_{Wasser} = 0,33$  angenommen und für das Wohngebiet  $\rho_{Wohn} = 0,16$  (siehe Tabelle 2.2). Dabei wird von einer gleichmäßigen Verteilung von 50 % Wasser und 50 % Wohngebiet ausgegangen. Eine exaktere Bestimmung der Umgebung wäre sehr aufwändig und ist aufgrund des geringen Anteils der reflektierten Strahlung im Vergleich zu den anderen Strahlungsteilen vernachlässigbar. Die Bodenreflexion ist für alle drei Seiten des Gebäudes mit den Solarzellen in diesem Fall identisch, da die Berechnung auf Basis eines isotropen Ansatzes erfolgt.

Die Basis ist zunächst folgende Formel:

$$\rho_B = \bar{\rho}_{Wasser} \cdot 0,5 + \bar{\rho}_{Wohn} \cdot 0,5 \quad (4.1)$$

mit:  $\bar{\rho}_{Wohn}$  = Bodenreflexion vom Wohngebiet mit Monateinfluss

$\bar{\rho}_{Wasser}$  = Bodenreflexion vom Wasser mit Monateinfluss

Je nach Jahreszeit kann die Bodenreflexion sich jedoch ändern, denn  $\rho_{Wasser}$  und  $\rho_{Wohn}$  gelten nur zu dem Zeitpunkt, an dem kein Schnee liegt, denn bei Schnee und Eis verbessern sich die Bodenreflexionswerte. Nach dem Klimadiagramm für Berlin [Ite] sind die Monate Dezember, Januar und Februar die kalten Monate, wobei der Januar im Vergleich der kälteste ist. Bei Schnee und Eis erhöht sich die Bodenreflexion für die Wohnumgebung leicht und fürs Wasser deutlich ( $\rho_{Eis} = 0,68$ ). Im Durchschnitt liegen in Berlin an 34 Tagen Schnee (Stand 2005) [Kle05]. Bei einer geschätzten Verteilung der Schneetage zu 30% Dezember, 40% Januar und 30% Februar ergibt sich die in Tabelle 4.1 dargestellte Verteilung der Schneetage für die entsprechenden Monate.

**Tabelle 4.1:** Aufteilung der Schneetage für Berlin für die Monate Dezember, Januar und Februar

Monat	Tage	kein Schnee	Schnee
Dezember	31	20,80 Tage = 67,097 %	10,20 Tage = 32,903 %
Januar	31	17,40 Tage = 56,129 %	13,60 Tage = 43,871 %
Februar	28,25	18,05 Tage = 63,894 %	10,20 Tage = 36,106 %

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

Dementsprechend wird  $\bar{\rho}_{Wasser}$  wie folgt eingeführt:

$$\text{für } M = 12 : \bar{\rho}_{Wasser} = 0,6710 \cdot \rho_{Wasser} + 0,3290 \cdot \rho_{Eis} \quad (4.2)$$

$$\text{für } M = 1 : \bar{\rho}_{Wasser} = 0,5613 \cdot \rho_{Wasser} + 0,4387 \cdot \rho_{Eis} \quad (4.3)$$

$$\text{für } M = 2 : \bar{\rho}_{Wasser} = 0,6389 \cdot \rho_{Wasser} + 0,3610 \cdot \rho_{Eis} \quad (4.4)$$

$$\forall M \in [3, 4, \dots, 11] : \bar{\rho}_{Wasser} = \rho_{Wasser} \quad (4.5)$$

Da der Unterschied von  $\bar{\rho}_{Wohn}$  zu  $\rho_{Wohn}$  marginal ist, wird die nachstehende vereinfachte Definition für  $\bar{\rho}_{Wohn}$  verwendet:

$$\text{für } M = 12 : \bar{\rho}_{Wohn} = \rho_{Wohn} + 0,03 \quad (4.6)$$

$$\text{für } M = 1 : \bar{\rho}_{Wohn} = \rho_{Wohn} + 0,04 \quad (4.7)$$

$$\text{für } M = 2 : \bar{\rho}_{Wohn} = \rho_{Wohn} + 0,03 \quad (4.8)$$

$$\forall M \in [3, 4, \dots, 11] : \bar{\rho}_{Wohn} = \rho_{Wohn} \quad (4.9)$$

mit:  $\rho_{Eis}$  = Bodenreflexion von Eis/schneebedecktem Wasser = 0,68

$\bar{\rho}_{Wohn}$  = Bodenreflexion vom Wohngebiet mit Monateinfluss

$\bar{\rho}_{Wasser}$  = Bodenreflexion vom Wasser mit Monateinfluss

$\rho_{Wohn}$  = Bodenreflexion vom Wohngebiet = 0,16

$\rho_{Wasser}$  = Bodenreflexion vom Wasser = 0,33

$M$  = Monat des Jahres

Die Verteilung der Schneetage auf die Monate sowie die Anzahl der Schneetage variiert jedoch von Jahr zu Jahr. Daher kann diese Berechnung nur als Schätzung betrachtet werden, in der Einflüsse wie zum Beispiel der seltenere Schnee im März nicht berücksichtigt werden.

#### **Einfluss des Wetters**

Das Wetter ist ein wichtiger Faktor bei der Berechnung der Leistung einer PV-Anlage, denn bei stark bewölktem oder nebligem Wetter sinkt die Globalstrahlung auf die Solarmodule stark ab. Der Grad der Bewölkung wird in Achtel gemessen, wobei die Wetterdienste für Prognosen normalerweise sogar nur zwischen fünf Zuständen unterscheiden (siehe Unterabschnitt 2.3.1). Durch diese sehr grobe Einteilung kann der Einfluss der Bewölkung auf die Globalstrahlung nur mit einem vergleichsweise großen Fehler bestimmt werden. Dies ist besonders wegen der großen Gewichtung der Bewölkung unvorteilhaft.

## 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

Trotzdem soll versucht werden, eine möglichst präzise Relation zwischen den beiden Werten zu finden. Eine Berechnung wie bei der Sonnenposition oder der Globalstrahlung ist nicht ohne weiteres möglich, da die Zusammenhänge sehr viel komplexer sind und damit den Rahmen dieser Arbeit überschreiten würden. Daher wird im Folgenden versucht eine Formel zu erstellen, die auf bereits vorhandenen Messdaten basiert. Der gemeinsame Nenner der Daten ist der Durchschnittswert am Tag, dementsprechend soll das Ziel sein, eine Relation zwischen dem durchschnittlichen Bewölkungsgrad und der durchschnittlichen Bestrahlungsstärke pro Tag herzustellen. Die folgenden Schritte wurden hierfür getätigt (siehe hierzu auch Abschnitt 2.2):

**Daten sammeln** Als erstes müssen möglichst präzise und umfangreiche Daten besorgt werden. Für den Bewölkungsgrad konnte die durchschnittliche Bewölkung der letzten 70 Jahre für Berlin pro Tag bezogen werden, welche kostenfrei vom Deutschen Wetterdienst<sup>5</sup> zur Verfügung gestellt wird. Die Datensätze sind so aufgebaut, dass in einer CSV-Datei jede Zeile einem Datensatz mit den durchschnittlichen Wetterdaten für einen bestimmten Tag entspricht. Als zweiter Wert wird die Bestrahlungsstärke benötigt. Diese liegt vom Sensor der Wetterstation im Minutentakt für die letzten zwei Jahre vor.

**Daten filtern** Der zweite Schritt ist die Filterung der Daten. Es müssen Datensätze mit Messfehlern und fehlenden Messwerten gefiltert werden. Bei der Bestrahlungsstärke muss zum Beispiel jeder Tag herausgefiltert werden, für den nur eine unzureichende Anzahl an korrekten Messwerten vorhanden ist, damit ein plausibler Durchschnittswert berechnet werden kann. Als Richtwert wurde gewählt, dass mindestens 50 % der Messwerte für einen Tag vorhanden sind. Hierfür wurde ein Python-Skript entwickelt, welches die CSV-Dateien einliest, filtert und erneut herausschreibt.

**Daten aufbereiten** Da die Daten aus verschiedenen Quellen stammen, müssen sie aufbereitet werden. Beispielsweise steht das Datum in verschiedenen Formaten in den Datensätzen. Für eine korrekte Zuordnung der Datensätze muss dieses vereinheitlicht werden. Des Weiteren muss auf die gleiche Datenbasis umgerechnet werden. Die Bestrahlungsstärke liegt im Minutentakt vor, weshalb der Tagesdurchschnitt berechnet wird, wie er für den Bewölkungsgrad verfügbar ist. Das Ergebnis ist eine CSV-Datei

---

<sup>5</sup><http://www.dwd.de>

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

für die durchschnittliche Bewölkung und eine Datei für die durchschnittliche Bestrahlungsstärke pro Tag für die letzten zwei Jahre.

**Daten zuweisen** Der nächste Schritt ist es, die Daten der beiden Dateien aufeinander abzubilden, sodass für den jeweiligen Tag eine Aussage zu beiden Werten getroffen werden kann. Der Wertebereich wird aus den gemessenen Daten abgelesen und ist in Tabelle 4.2 dargestellt. Auf diesen Daten aufbauend werden die Daten anhand des Bewölkungsgrads in Achtel aufgeteilt. Für jedes Achtel wird schließlich der Mittelwert berechnet, der in Tabelle 4.3 dargestellt ist. Dieser ergibt im Verhältnis zum maximal gemessenen Mittelwert den Bewölkungseinflussfaktor  $b_E$ .

**Tabelle 4.2:** Gemessene Wertebereiche für die Bestrahlungsstärke und den Bewölkungsgrad

	Minimal:	Maximal:
Bewölkungsgrad $\emptyset$	0,0	8,0
Bestrahlungsstärke	0,0	1 386,0
Bestrahlungsstärke $\emptyset$ (nur tagsüber)	19,270	534,261

**Tabelle 4.3:** Der Bewölkungseinflussfaktor  $b_E$  für den Standort des FKI-Gebäudes

Bewölkungsgrad $b$	Bestrahlungsstärke $\emptyset$	Bewölkungseinflussfaktor $b_E$
$b \geq 0 \wedge b < 1$	369,520	69,165 %
$b \geq 1 \wedge b < 2$	381,465	71,401 %
$b \geq 2 \wedge b < 3$	365,361	68,386 %
$b \geq 3 \wedge b < 4$	319,829	59,864 %
$b \geq 4 \wedge b < 5$	322,244	60,316 %
$b \geq 5 \wedge b < 6$	249,004	46,607 %
$b \geq 6 \wedge b < 7$	196,028	36,692 %
$b \geq 7 \wedge b \leq 8$	95,816	17,934 %

**Ergebnis** Mit Hilfe des Bewölkungseinflussfaktors  $b_E$  lässt sich eine Schätzung für die Globalstrahlung in Abhängigkeit der Bewölkung treffen. Die Tabelle 4.4 zeigt die Umrechnung auf die meteorologischen Angaben zum Bewölkungsgrad. Dabei wird für jeden

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

Teilbereich eine Gewichtung vorgenommen. Beispielsweise wird der Bewölkungseinflussfaktor für den Bewölkungsgrad 0,5 bis 3,5 ( $b_{0,5-3,5}$ ) folgendermaßen bestimmt:

$$\begin{aligned} b_{0,5-3,5} &= \frac{0,5 \cdot b_{0-1} + 1 \cdot b_{1-2} + 1 \cdot b_{2-3} + 0,5 \cdot b_{3-4}}{3} \\ &= \frac{0,5 \cdot 69,165 \% + 1 \cdot 71,401 \% + 1 \cdot 68,386 \% + 0,5 \cdot 59,864 \%}{3} \\ &= \underline{\underline{68,101 \%}} \end{aligned}$$

Weitere Wetterzustände, wie zum Beispiel Regen oder Nebel, werden genauso wie der Zustand Bedeckt behandelt. Es ergibt sich die Berechnung der Globalstrahlung in Abhängigkeit der Bewölkung  $E_{G,b}$ :

$$E_{G,b} = E_{G,g} \cdot b_E \quad (4.10)$$

mit:  $E_{G,g}$  = Gesamtstrahlung auf eine geneigte Fläche

$b_E$  = Bewölkungseinflussfaktor

Wird zum Beispiel eine Bestrahlungsstärke  $E_{G,g} = 850,983 \frac{\text{W}}{\text{m}^2}$  berechnet und es liegt wolkiges Wetter vor ( $b_E = 51,734 \%$ ), dann wird die Bestrahlungsstärke mit Wettereinfluss  $E_{G,b}$  wie folgt bestimmt:

$$\begin{aligned} E_{G,b} &= E_{G,g} \cdot b_E \\ &= 850,983 \frac{\text{W}}{\text{m}^2} \cdot 51,734 \% \\ &= \underline{\underline{440,248 \frac{\text{W}}{\text{m}^2}}} \end{aligned}$$

**Tabelle 4.4:** Bewölkungseinflussfaktor  $b_E$  für die meteorologischen Angaben des Bewölkungsgrades für den Standort des FKI-Gebäudes

Bezeichnung	Bewölkungsgrad $b$	Bewölkungseinflussfaktor $b_E$
sonnig	0,0 – 0,5	70,283 %
heiter	0,5 – 3,5	68,101 %
wolkig	3,5 – 6,5	51,734 %
stark bewölkt	6,5 – 7,5	27,313 %
bedeckt	7,5 – 8,0	17,934 %



### 4.4.2 Berechnung der Leistung der PV-Anlage zu einem bestimmten Zeitpunkt

Für die Berechnung der Leistung der PV-Anlage zu einem bestimmten Zeitpunkt müssen die folgenden Werte bekannt sein:

- Die aktuelle Lufttemperatur ( $\vartheta_U$ ) und die Proportionalitätskonstante der Solarmodule ( $c$ )
- Der Temperaturkoeffizient ( $\alpha$ ) der Solarmodule; dieser Wert kann gegebenenfalls über Tabellen geschätzt werden
- Die Leistung ( $P_{STC}$ ) unter Standardtestbedingungen sowie die Fläche ( $A$ ) einer Solarzelle
- Die Globalstrahlung auf die Solarmodule mit Berücksichtigung des Wetters ( $E_{G,b}$ ) zu dem aktuellen Zeitpunkt

Wenn diese Werte alle vorliegen, kann über die nachstehenden Schritte die Leistung einer Solarzelle bestimmt werden:

1. Zuerst wird der Wirkungsgrad unter Standardtestbedingungen ( $\eta_{STC}$ ) berechnet.
2. Mit dem Ergebnis kann der reale Wirkungsgrad  $\eta$  bestimmt werden.
3. Als nächster Schritt wird aus der aktuellen Lufttemperatur ( $\vartheta_U$ ) und der Proportionalitätskonstante ( $c$ ) die Modultemperatur ( $\vartheta_M$ ) errechnet.

Auf diesen Ergebnissen aufbauend kann folglich die Leistung für eine Solarzelle mit Hilfe der Gleichung 2.17 berechnet werden. Das Ergebnis ist die Leistung für lediglich eine Solarzelle. Dieses Ergebnis muss daher noch mit der Anzahl der Solarmodule auf der Fassadenseite multipliziert werden. Abschließend wird der Vorgang für die beiden weiteren Seiten der Fassade mit PV-Anlage wiederholt.

### 4.4.3 Maximale Leistung der PV-Anlage

Die Analyse der historischen Daten der Wetterstation ergibt, dass der 29.06.2011 der Tag war, an dem durchschnittlich die höchste Bestrahlungsstärke erreicht worden ist. Aus den Daten für diesen Tag lässt sich die durchschnittliche maximale Leistung für die drei Zeitspannen errechnen (siehe Tabelle 4.5). Beispielhaft wird im Folgenden die Berechnung für die Zeitspanne von 6 bis 11 Uhr mit 76 Modulen gezeigt:

mit:  $P_{STC} = 340 \text{ W}$

$$\eta_{STC} = 8,026 \%$$

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

$$\begin{aligned}
 E_{G,b} &= 633,230 \frac{\text{W}}{\text{m}^2} \\
 \alpha &= -0,4 \% \\
 A &= 4,236 \text{ m}^2 \\
 \vartheta_U &= 23,617 \text{ }^\circ\text{C} \\
 c &= 35 \text{ }^\circ\text{C} \\
 f &= 0,9
 \end{aligned}$$

$$\begin{aligned}
 \vartheta_M &= \vartheta_U + c \cdot \frac{E_{G,b}}{1000 \frac{\text{W}}{\text{m}^2}} \\
 &= 23,617 \text{ }^\circ\text{C} + 35 \text{ }^\circ\text{C} \cdot \frac{633,23 \frac{\text{W}}{\text{m}^2}}{1000 \frac{\text{W}}{\text{m}^2}} \\
 &= \underline{\underline{45,780 \text{ }^\circ\text{C}}}
 \end{aligned}$$

$$P = \eta_{STC} \cdot \left( 1 + 0,0825 \cdot \ln \frac{E_{G,b}}{1000 \frac{\text{W}}{\text{m}^2}} \right) \cdot A \cdot E_{G,b} \cdot (1 + \alpha \cdot (\vartheta_M - 25)) \cdot f$$

$$P = 8,026 \% \cdot \left( 1 + 0,0825 \cdot \ln \frac{633,23 \frac{\text{W}}{\text{m}^2}}{1000 \frac{\text{W}}{\text{m}^2}} \right) \cdot 4,236 \text{ m}^2 \cdot$$

$$633,23 \frac{\text{W}}{\text{m}^2} \cdot (1 + (-0,004) \cdot (45,780 - 25)) \cdot 0,9$$

$$P = \underline{\underline{170,956 \text{ W}}}$$

$$\Rightarrow 170,956 \text{ W} \cdot 5 \text{ h} = 854,779 \text{ Wh pro Modul}$$

$$\Rightarrow 854,779 \text{ Wh} \cdot 76 = \underline{\underline{64,963 \text{ kWh}}} \text{ für die komplette Anlage}$$

**Tabelle 4.5:** Maximale Leistung in verschiedenen Zeitspannen eines Tages für die PV-Anlage des FKI-Gebäudes

Zeitspanne	$\vartheta_U$ in $^\circ\text{C}$	$\vartheta_M$ in $^\circ\text{C}$	$E_{G,b}$ in $\frac{\text{W}}{\text{m}^2}$	P in Wh
06:00 - 11:00	23,617	45,780	633,230	64 963,204
11:00 - 17:00	28,977	56,138	776,056	92 816,735
17:00 - 23:00	26,070	29,643	102,094	11 349,181

#### 4.4.4 Möglichkeiten durch die Prognose der Stromerzeugung

Die erstellte Prognose für die Stromerzeugung des Gebäudes kann für viele unterschiedliche Szenarien verwendet werden:

#### 4.4 Algorithmus für die Prognose des erzeugten Solarstroms

---

- Der Fahrstuhl wird über zusätzliche Hardware wie zum Beispiel ein ansteuerbares Touchdisplay verfügen. Wenn gerade wenig Strom produziert wird, könnte ein Hinweis erscheinen, dass die Person möglichst lieber auf den Fahrstuhl verzichten und die Treppe nutzen sollte.
- Wenn gerade mehr Strom erzeugt als verbraucht wird, es allerdings absehbar ist, dass es in der Folgezeit wieder deutlich weniger wird, könnte eine Information an die Mitarbeiter versendet werden, dass diese möglichst zeitnah alle Geräte mit Akkus aufladen. Die Information könnte beispielsweise per E-Mail versendet werden.
- Ebenso könnte die Temperatur im Serverraum höher geregelt werden, damit die Klimaanlage weniger Arbeit zu verrichten hat, sobald weniger Strom erzeugt wird.
- Bei wenig produziertem Strom kann die Helligkeit der Medienfassade herunterreguliert oder die Medienfassade sogar komplett abgeschaltet werden.

Aus den Beispielen kann geschlussfolgert werden, dass es drei Zustände gibt, aus denen weitere Anwendungsszenarien erstellt werden können:

- Es wird mehr Strom erzeugt als verbraucht. Der Zustand in der Zukunft ist nicht relevant.
- Es wird mehr Strom erzeugt als verbraucht. Für die Zukunft wird erwartet, dass weniger Strom erzeugt als verbraucht wird.
- Es wird weniger Strom erzeugt als verbraucht.

Beim dritten Zustand gibt es keine Aufteilung, ob der Zustand in der Zukunft relevant ist oder nicht. Bei den ersten beiden Fällen, wenn mehr Strom erzeugt wird als verbraucht, kann dieser für weitere Anwendungen in irgendeiner Form gespeichert werden. Wird hingegen mehr verbraucht als erzeugt, gibt es keinen speicherbaren Überschuss.

# 5 Proof of Concept

In diesem Kapitel werden die wichtigsten Implementierungsdetails des Entwurfs, der im letzten Kapitel erarbeitet wurde, näher erläutert. Zuerst werden kurz die Widget-Klasse sowie einige Probleme und deren Lösungen bei der Erstellung der Kernkomponente, dem Widget-Launcher, beschrieben. Danach werden ausgewählte Probleme bei der Umsetzung der Widgets dargestellt und auf verschiedene weitere Aspekte, wie die Schnittstellen oder die Konfiguration der Anwendung, eingegangen. Abschließend folgen eine kurze Beschreibung der verwendeten Bibliotheken und eine Darstellung des Fehlerbehandlungskonzepts der Anwendung.

## 5.1 Widget-Klasse

Die Widget-Klasse ist eine abstrakte Elternklasse, von der jedes implementierte Widget (siehe Abschnitt 4.3) abgeleitet wird. Die wichtigsten Funktionen der abstrakten Widget-Klasse:

- start()** Diese Methode wird aufgerufen, sobald das Widget angezeigt werden soll. In der Methode werden zum Beispiel die Sensoren initialisiert und die Grafiken geladen.
- stop()** Die **stop()**-Methode wird aufgerufen, wenn die Laufzeit des Widgets vorüber ist. In der Funktion können beispielsweise Attribute zurückgesetzt werden.
- render()** Für jedes Frame wird die **render()**-Methode aufgerufen. In ihr werden unter anderem die Schriften und Grafiken gerendert.
- update()** Die **update()**-Methode wird bei einem Screen-Wechsel aufgerufen. Mit dem Aufruf wird dem Widget mitgeteilt, dass es einen anderen Inhalt darzustellen hat (siehe Unterabschnitt 5.2.2).

## 5.2 Widget-Launcher

Der Widget-Launcher ist ein elementares Element der Anwendung. Über ihn werden die Widgets gestartet, gerendert und gestoppt. Ebenso wird der korrekte Wechsel der Widgets nach deren Laufzeit gesteuert.

Sämtliche Methoden eines Widgets werden zentral vom Widget-Launcher aufgerufen. Der genaue Ablauf ist in Abbildung 5.1 dargestellt. Nach der Erstellung des Widgets wird dieses in einer Warteschlange abgelegt. Diese Warteschlange wird über die Zeit abgearbeitet (siehe Unterabschnitt 5.2.1). Sobald ein Widget an der Reihe ist, wird die `start()`-Methode aufgerufen. Hiermit beginnt die Laufzeit des Widgets. In seiner Laufzeit wird für jedes Frame die Methode `render()` und falls nötig nach einer gewissen Zeit die Methode `update()` für einen Screen-Wechsel (siehe Unterabschnitt 5.2.2) aufgerufen. Sobald die Laufzeit des Widgets vorüber ist, wird die Methode `stop()` aufgerufen. Die Widgets bleiben für die Programmlaufzeit erhalten und werden erst zum Programmende zerstört.

### 5.2.1 Warteschlange

Zum Start der Anwendung werden zunächst alle Widgets erzeugt und in einer Warteschlange abgelegt. Deren Abarbeitung erfolgt zur Laufzeit nach einem Ring-Prinzip. Wie in Abbildung 5.2 dargestellt, werden zwei Widgets gleichzeitig visualisiert. Beim Start des Programms sind dies die ersten beiden Widgets in der Warteschlange, wobei das erste Widget auf der rechten Position des Dashboards angezeigt wird und das zweite auf der linken. Sobald die Laufzeit der Widgets vorüber ist, werden diese zurück ans Ende der Warteschlange gesetzt. Die freigewordene Position wird durch ein neues Widget aus der Warteschlange ersetzt. Somit wird eine geordnete und gleichmäßige Laufzeit für jedes Widget garantiert, egal wie viele Widgets für das Dashboard verfügbar sind. Für einen reibungslosen Betrieb sind jedoch mindestens drei Stück erforderlich. Eine größere Anzahl ist allerdings empfehlenswert, damit die Anwendung auch beim Auftreten eines Fehlers weiterlaufen kann (siehe Abschnitt 5.8). Jedes Widget verfügt über eine individuelle Laufzeit. Dies bietet die Möglichkeit, Widgets mit mehreren Screens (siehe Unterabschnitt 5.2.2) länger als solche mit nur einem laufen zu lassen.

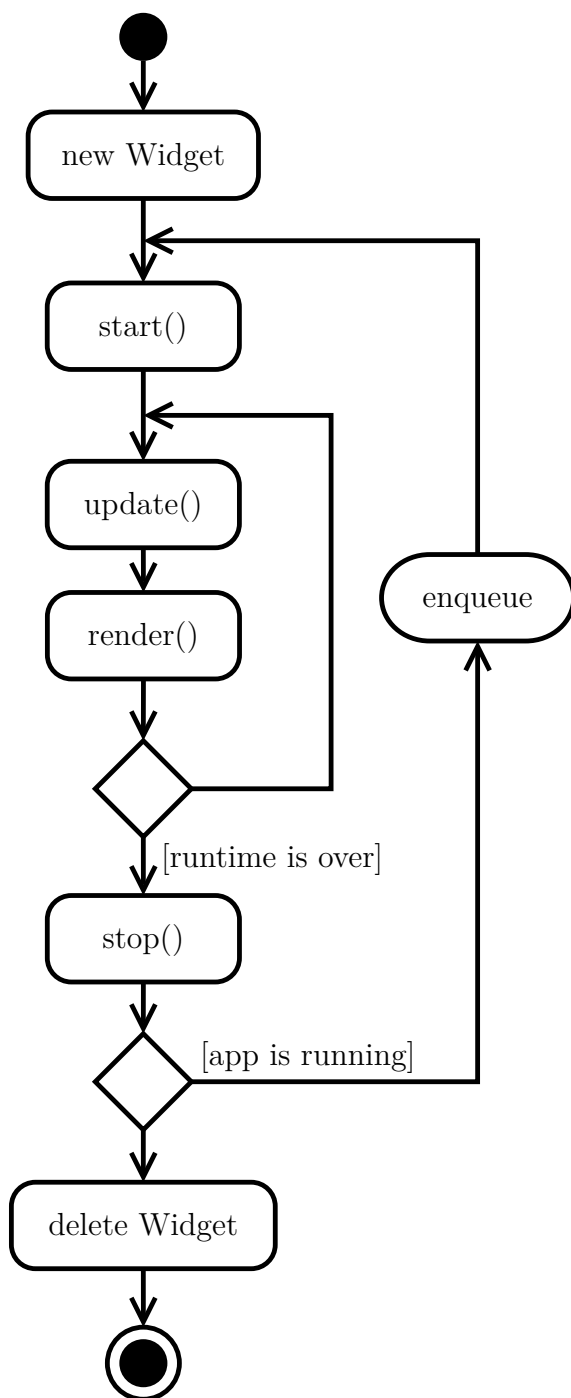


Abbildung 5.1: Lebenszyklus eines Widgets

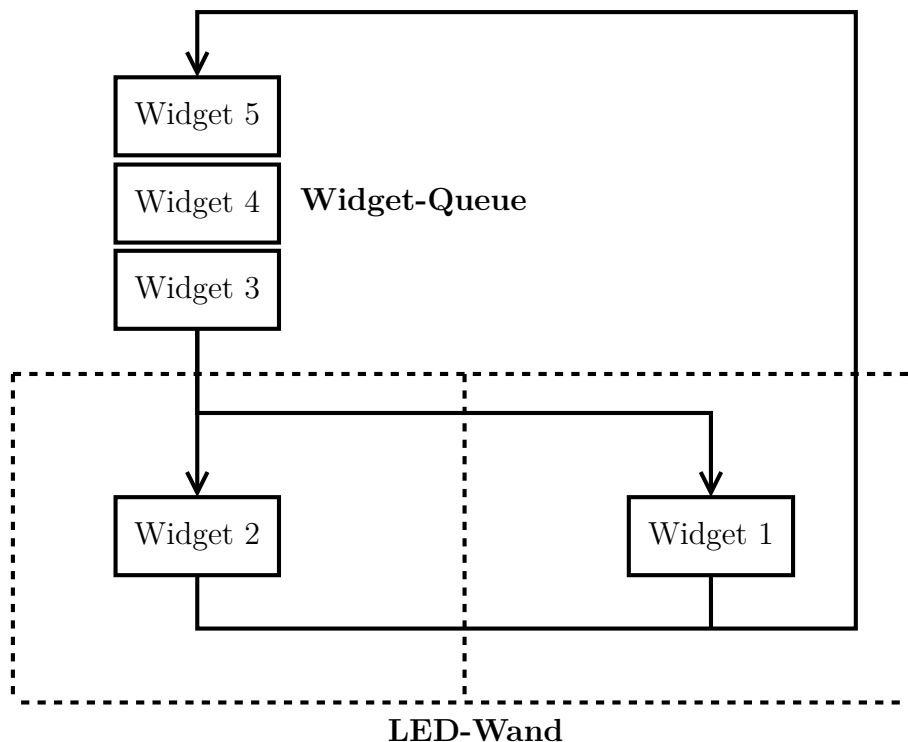


Abbildung 5.2: Widget-Warteschlange beim Widget-Launcher

### 5.2.2 Screens

Da die Anzahl der Pixel für ein Widget sehr gering ist, wurde ein Screen-Konzept implementiert. Dies bedeutet, dass jedes Widget über mehrere Oberflächen (Screens) verfügen kann, zwischen denen gewechselt wird. Somit können pro Widget deutlich mehr Informationen angezeigt werden. Als Beispiel wäre hier die Wettervorhersage für drei Tage zu nennen, wobei jeder Screen genau einem Tag entsprechen würde. Die Umsetzung ist wie folgt realisiert: Jedes Widget verfügt über eine Laufzeit. Diese wird durch die Anzahl der Screens geteilt, womit die Laufzeit pro Screen definiert wird. Der Widget-Launcher überwacht die Laufzeit des Widgets und teilt diesem mit Hilfe der Methode `update()` mit, dass der Screen gewechselt werden soll.

## 5.3 Implementierte Widgets

Um auch den korrekten Wechsel der Widgets zu zeigen, mussten mindestens drei Widgets implementiert werden. Es wurden für den Prototyp die Prognose des erzeugten Stroms, das aktuelle Wetter und die Wettervorhersage für die nächsten drei Tage implementiert. Zusätzlich wurde noch das Widget für die Berechnung der Sonnenposition

## 5.3 Implementierte Widgets

---

beispielhaft für die Interaktion umgesetzt. Die Wahl fiel auf diese Widgets, da einige Sensoren erst sehr spät oder gar nicht während der Bearbeitungszeit dieser Arbeit zur Verfügung standen. Dazu zählen zum Beispiel das Smart-Metering-System oder die Logger der PV-Anlage. Die Wetterstation und virtuellen Sensoren waren hingegen von Beginn an verfügbar.

Im Gegensatz zum Entwurf wurde eine wichtige visuelle Änderung an allen Widgets vorgenommen: Der Hintergrund ist nicht mehr weiß sondern schwarz. Dies hat den großen Vorteil, dass bei Schwarz die LEDs nicht leuchten und dementsprechend auch keinen Strom benötigen. Bei Weiß hingegen ist der Strombedarf der LEDs am höchsten. Es wäre kontraproduktiv, wenn die Dashboard-Anwendung, welche das Ziel hat, möglichst häufig auf der Medienfassade zu laufen und Informationen zum Strombedarf und zur Stromerzeugung zu liefern, nicht so gestaltet wäre, dass sie selbst wenig Strom verbraucht. Für die gute Lesbarkeit wurde als Kontrast zum Hintergrund weiße Schrift verwendet. In Abbildung 5.3 und Abbildung 5.4 sind Screenshots der fertigen Widgets abgebildet.

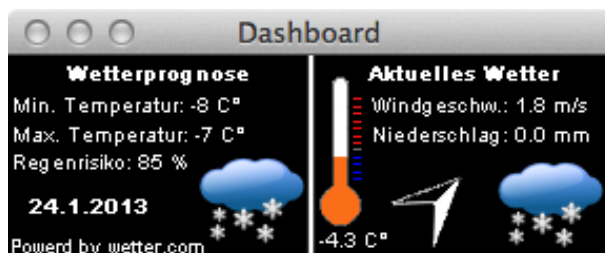


Abbildung 5.3: Screenshot der Widgets „Wetterprognose“ und „Aktuelles Wetter“



Abbildung 5.4: Screenshot der Widgets „Sonnenposition“ und „Prognose des erzeugten Stroms“

### 5.3.1 Widget für die Prognose des Stroms

Das Klassendiagramm in Abbildung 5.5 zeigt die Beziehung zwischen den Klassen, die für die Berechnung des Stroms notwendig sind. Die Klasse `PowerPredictionWidget`



## 5.3 Implementierte Widgets

hat für jeden Zeitraum, der berechnet werden soll, ein Objekt der `PowerPrediction`-Klasse. Für die Umsetzung des Widgets wurden drei Zeiträume definiert: von 6-11 Uhr, 11-17 Uhr und 17-23 Uhr. Jede `PowerPrediction` hat wiederum ein Element der Klasse `SolarPosition`. Diese Klasse ist für die Berechnung der Sonnenposition verantwortlich. Des Weiteren hat die Klasse `PowerPrediction` so viele Elemente der Klasse `SolarRadiation`, wie es Seiten der PV-Anlage gibt. Die Sonnenposition ist für jede Seite identisch, die Globalstrahlung jedoch nicht. Außerdem verfügt die `PowerPrediction` noch für jedes Solarmodul über ein Element der Klasse `ModulePowerPrediction`, welches zur Berechnung der Leistung einer Solarzelle dient. Zusätzlich werden die Daten des virtuellen Wettersensors mit der Klasse `ForecastSensor` ermittelt. Schlussendlich kann in Kombination mit diesen Daten über die Methode `getPower()` der Klasse `PowerPrediction` für den definierten Zeitraum die Stromprognose abgerufen werden.

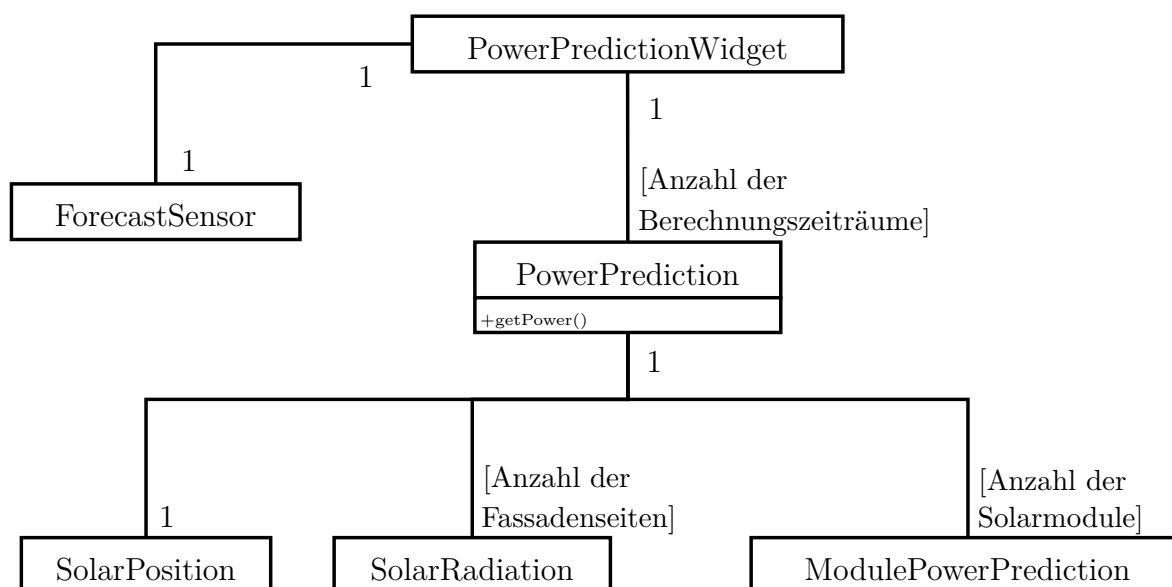


Abbildung 5.5: Klassendiagramm für die Berechnung der Stromprognose

### 5.3.2 Interaktion mit dem Widget für die Berechnung der Sonnenposition

Für das Widget für die Berechnung der Sonnenposition (siehe Unterabschnitt 4.3.1) wird der Benutzer die Möglichkeit der Interaktion haben (siehe Unterabschnitt 4.3.4). Er wird mit seinem mobilen Gerät eine Webseite aufrufen können, auf der er die Eingabemöglichkeit für das Datum und die Uhrzeit hat (siehe Abbildung 5.6). Weiterhin

## 5.4 Konfiguration der Anwendung

---

kann der Nutzer über den „Jetzt-Button“ jederzeit auf den aktuellen Zeitpunkt wechseln. Sobald der Nutzer einen der Datums- oder Zeit-Parameter ändert, wird das neue Datum an den Server gesendet. Wird das Widget gerade dargestellt, werden die Änderungen sofort umgesetzt. Sollten jedoch zwei andere Widgets auf dem Dashboard laufen, dann wird die Änderung zwischengespeichert und bei der nächsten Anzeige des Widgets ausgeführt. Zur Realisierung wurde hierfür auf Seite des Servers eine Warteschlange implementiert. Damit sich die Nutzer nicht gegenseitig in die Quere kommen, kann immer nur ein einzelner Nutzer die Steuerung übernehmen; eine parallele Steuerung ist nicht möglich.



Abbildung 5.6: Screenshot der Eingabemöglichkeit über das Smartphone

## 5.4 Konfiguration der Anwendung

Die Konfiguration der Anwendung funktioniert über eine Konfigurationsdatei. Sämtliche in der Anwendung verwendeten Parameter wurden konsequent in diese Datei ausgelagert. Dabei werden die Bereiche „Widgets“, „Ort und Umgebung“, „PV-Anlage“,

„Visualisierungsgerät“ und „Datenbank“ abgedeckt. Beispielsweise kann der Benutzername, Hostname und Port der Datenbank definiert werden. Zum Lesen der Konfigurationsdatei wurde die Bibliothek `SDL_Config` verwendet (siehe Unterabschnitt 5.7.1), welche eine schnelle und einfache Implementierung zur Verfügung stellt.

## 5.5 Schnittstellen

Als Basis für die Berechnung der prognostizierten Stromerzeugung der PV-Anlage wurden mehrere Berechnungen aus dem Bereich der Astronomie und Photovoltaik implementiert. Insbesondere die Bestimmung der Sonnenposition an einem gegebenem Ort zu einer gegebenen Zeit und die Berechnung der Leistung einer PV-Anlage könnten auch für andere Entwickler von Interesse sein. Dementsprechend sollen diese Berechnungen auch in geeigneter Form zur Verfügung gestellt werden.

Bei der Entwicklung wurden alle Berechnungen in drei Header-Dateien ausgelagert:

**SolarRadiationUtil.hpp:** Sammlung von Funktionen zur Berechnung der Globalstrahlung und ihrer Bestandteile

**SolarPositionUtil.hpp:** Sämtliche Funktionen zur Berechnung der Sonnenposition

**PhotovoltaicsUtil.hpp:** Funktionen für die Berechnung der Leistung der PV-Anlage und der dafür benötigten Parameter

Über das Makefile der Anwendung lässt sich sowohl eine statische als auch dynamische Bibliothek kompilieren, die in anderen Anwendungen genutzt werden kann. In der Dashboard-Anwendung wurden Klassen implementiert, die ebenso auf diesen Funktionen aufbauen. Alle Berechnungen erfolgen für eine hohe Genauigkeit mit dem Datentyp „double“.

Vorstellbar wäre auch die Bereitstellung der Funktionalitäten über *Remote Procedure Calls*. Da die Dashboard-Anwendung allerdings nur eine von vielen Anwendungen für die Medienfassade sein wird und dementsprechend eine nicht vorhersehbare Laufzeit hat, müsste hierfür eine separate Server-Anwendung geschrieben werden, die diese Schnittstelle zum Beispiel in Form von *SOAP* zur Verfügung stellt.

## 5.6 Komponenten

Bei der Umsetzung der Anwendung wurde darauf geachtet, die Funktionalitäten soweit wie möglich in Komponenten zusammenzufassen. Diese sind so gestaltet, dass sie von

## 5.7 Verwendete Bibliotheken

---

den verschiedenen Widgets genutzt werden können. Es gibt die folgenden Komponenten:

**Realer Wettersensor:** Diese Komponente realisiert die Ansteuerung der Wetterstation.

**Virtueller Wettersensor:** Der virtuelle Wettersensor holt die Wetterprognose vom Anbieter und bereitet die Daten für die weitere Nutzung auf.

**Sonnenposition:** Mit Hilfe dieser Komponente kann die Sonnenposition zu einem gegebenem Zeitpunkt berechnet werden.

**Globalstrahlung:** Dies ist die Komponente für die Berechnung der Globalstrahlung.

**Photovoltaik:** Mit dieser Komponente kann der erzeugte Strom einer PV-Anlage ermittelt werden.

**Interaktions-Server:** Dies ist der Server für die Interaktion, der die Verbindung zum Node.js-Server regelt.

Tabelle 5.1 zeigt eine Übersicht darüber, welche Widgets welche der Komponenten nutzen.

**Tabelle 5.1:** Übersicht über die verwendeten Komponenten pro Widget

	Strom- prognose	Aktuelles Wetter	Wetter- vorhersage	Sonnen- position
Realer Wettersensor	X	X	-	-
Virtueller Wettersensor	X	X	X	-
Sonnenposition	X	-	-	X
Globalstrahlung	X	-	-	-
Photovoltaik	X	-	-	-
Interaktions-Server	-	-	-	X

## 5.7 Verwendete Bibliotheken

Zur weiteren Unterstützung bei der Umsetzung wurden Bibliotheken eingesetzt. Im Folgenden werden diese kurz vorgestellt und erläutert, warum diese Bibliothek ausgewählt wurde.

### 5.7.1 Simple DirectMedia Layer

*Simple DirectMedia Layer* (SDL)<sup>1</sup> ist eine plattformunabhängige Multimedia-Bibliothek für C++, welche entwickelt wurde, um einen einfachen Zugriff auf das Soundsystem,

---

<sup>1</sup><http://www.libsdl.org>

die Maus, die Tastatur, den Joystick, die 3D Hardware via OpenGL und den 2D Video-Framebuffer zu ermöglichen. Im Gegensatz zu *DirectX* werden verschiedene Plattformen unterstützt: einerseits alle gängigen, wie zum Beispiel Windows, Mac OS X und Linux, andererseits auch weniger verbreitete Betriebssysteme wie Solaris, FreeBSD oder OpenBSD. Die Bibliothek ist unter der GNU LGPL Version 2 lizenziert, wodurch eine kostenlose Nutzung, sogar in kommerziellen Produkten, ermöglicht wird. [SDL, Sch12] Diese Bibliothek ist die Basis für die Anwendung, da mit ihrer Hilfe die komplette Visualisierung gelöst ist. Besonders im Bereich der 2D-Anwendungen ist ihre Nutzung verbreitet. Aufgrund der Plattformunabhängigkeit ist die Anwendung kompatibel, egal ob für die Medienfassade schlussendlich Server mit Mac OS X oder Linux zum Einsatz kommen. Für die Realisierung der Anwendung wurden weitere Bibliotheken als Erweiterung für die SDL genutzt:

**SDL\_Config V. 1.2:** *SDL\_Config*<sup>2</sup> ist eine Erweiterung für das Parsen und Schreiben von Konfigurationsdateien. Ziel der Bibliothek ist es, eine einfache, schnelle und plattformunabhängige Bibliothek für Konfigurationsdateien zur Verfügung zu stellen. [Rut]

**SDL\_ttf V. 2.0.11:** *SDL\_ttf*<sup>3</sup> ist eine auf *FreeType* 2<sup>4</sup> basierende Bibliothek, mit der auf einfache Art TrueType-Schriften gerendert werden können. Dabei werden gebräuchliche Schriftoperationen wie die Änderung des Schriftgrads oder des Schriftstils unterstützt. Des Weiteren ist es möglich, die Schrift zu glätten oder eine Transparenz anzuwenden. [Lan]

**SDL\_image V. 1.2:** Die Bibliothek *SDL\_image*<sup>5</sup> ermöglicht das Lesen weit verbreiteter Bildformate wie zum Beispiel PNG, JPEG, TIFF oder GIF. [LE]

**SDL\_net V. 1.2.8:** *SDL\_net*<sup>6</sup> ist eine plattformunabhängige Netzwerkbibliothek. Mit ihrer Unterstützung werden die Netzwerkfunktionalitäten verschiedener Betriebssysteme abstrahiert. Insbesondere der Aufbau von Verbindungen sowie der Datentransfer werden vereinfacht. [SLM]

---

<sup>2</sup>[http://koshmaar.linuxpl.info/sdl\\_config/](http://koshmaar.linuxpl.info/sdl_config/)

<sup>3</sup>[http://www.libsdl.org/projects/SDL\\_ttf/](http://www.libsdl.org/projects/SDL_ttf/)

<sup>4</sup><http://www.freetype.org/freetype2/>

<sup>5</sup>[http://www.libsdl.org/projects/SDL\\_image/](http://www.libsdl.org/projects/SDL_image/)

<sup>6</sup>[http://www.libsdl.org/projects/SDL\\_net/](http://www.libsdl.org/projects/SDL_net/)

### 5.7.2 Boost

*Boost*<sup>7</sup> ist eine Sammlung von portablen C++ Bibliotheken, die ein großes Spektrum an Anwendungsmöglichkeiten abdecken. Es ist eine der meistgenutzten C++ Bibliotheken, deren Inhalte teilweise sogar für die Übernahme in den C++ Standard vorgeschlagen wurden. Boost enthält verschiedene Komponenten, unter anderem für Container, Spracherweiterungen, Zeitberechnungen, mathematische Funktionen, Tests und vieles mehr.

### 5.7.3 libcurl

Die Bibliothek *libcurl*<sup>8</sup> ist eine Multiprotokoll-Bibliothek für C. Sie ist weit verbreitet, kostenlos und unterstützt alle gängigen Protokolle, wie zum Beispiel FTP, HTTP, POP3, LDAP, IMAP usw. Zusätzlich ist die Bibliothek portabel, schnell und gut dokumentiert. [Ste]

In der Anwendung wird libcurl für die Nutzung der REST-API von [wetter.com](http://wetter.com) verwendet.

### 5.7.4 JSON Spirit

*JSON Spirit*<sup>9</sup> ist eine quelloffene und kostenlose JSON-Implementierung für C++. Ihre Vorteile sind, dass sie über Unicode-Unterstützung verfügt und durch die Nutzung der Standard C++ Bibliothek sehr gut in C++ integriert ist. Die JSON-Objekte werden in einem Standard-Vektor oder einer Standard-Map gehalten und es gibt Lese- und Schreib-Funktionen für die Standard-Strings und -Streams. Dadurch ist eine gute Interoperabilität gegeben. Die Bibliothek hat jedoch den Nachteil, dass sie von Boost (siehe Unterabschnitt 5.7.2) abhängig ist. [Wil11]

JSON Spirit wurde eingesetzt, um die auf JSON basierenden Wettervorhersagen von [wetter.com](http://wetter.com) auszuwerten.

---

<sup>7</sup><http://www.boost.org>

<sup>8</sup><http://curl.haxx.se>

<sup>9</sup><http://www.codeproject.com/Articles/20027/JSON-Spirit-A-C-JSON-Parser-Generator-Implemented>

### 5.7.5 SOCI

Für die Ansteuerung der Datenbank wird die Bibliothek *SOCI*<sup>10</sup> verwendet. SOCI ist eine Datenbankzugriffsbibliothek, die eine sehr gute Integration in C++ bietet. Ziel ist es, dem Entwickler einen natürlichen und intuitiven Weg für den Datenbankzugriff bereitzustellen. Zudem bietet die Bibliothek eine Datenbankabstraktionsschicht für die Nutzung unterschiedlicher Datenbankmanagementsysteme. In der aktuellen Version 3.1.0 werden MySQL, Oracle, PostgreSQL, SQLite3 und ODBC als generisches Backend unterstützt. [Sob]

Zurzeit wird in der Systemarchitektur der Anwendung eine postgresQL-Datenbank verwendet, weswegen auch die Nutzung der offiziellen postgresQL-Bibliothek *libpqxx*<sup>11</sup> möglich gewesen wäre. Allerdings kann mit Hilfe der Verwendung einer Datenbankabstraktionsschicht die eingesetzte Datenbank später leichter ausgetauscht werden. Die dafür nötigen Anpassungen wären vom Aufwand her sehr gering. Im Gegensatz zu anderen Bibliotheken für C++ mit einer Datenbankabstraktionsschicht ist SOCI quelloffen und kostenlos.

### 5.7.6 MD5

Für die Nutzung der REST-API von *wetter.com* wird eine Funktion zum Erstellen eines MD5-Hashwerts über einen gegebenen String benötigt. Hierfür wurde die C++ Implementierung für *MD5* von Frank Thilo verwendet. Bei einer REST-Anfrage für die Wettervorhersage wird der MD5-Hashwert über den Projektnamen, den API-Key und den Ort gebildet. Dieser wird für die Authentifizierung bei der Anfrage verwendet.

### 5.7.7 jQuery mobile

*jQuery* ist eine sehr weit verbreitete JavaScript-Bibliothek. Auf dieser aufbauend ist *jQuery mobile*<sup>12</sup> die für mobile Geräte optimierte Version von jQuery. Die Bibliothek hat den großen Vorteil, dass sie in Kombination mit HTML5 eine Unterstützung für sehr viele mobile Plattformen bietet. Weiterhin benötigt sie wenig Ressourcen, unterstützt Barrierefreiheit und verfügt über eine sehr große Nutzerzahl. Es ist aufgrund der vielen bereitgestellten Funktionalitäten einfach für den Entwickler, Webseiten mit komplexen interaktiven Inhalten zu gestalten. jQuery mobile ist dabei so aufgebaut,

---

<sup>10</sup><http://soci.sourceforge.net>

<sup>11</sup><http://pqxx.org/development/libpqxx/>

<sup>12</sup><http://jquerymobile.com>

dass die erstellten Seiten ähnlich wie native Apps auf den Nutzer wirken. [Böh]

Die Webseite für die mobilen Geräte für die Interaktion mit der Anwendung wurde auf Basis von jQuery mobile erstellt (siehe Unterabschnitt 5.3.2).

### 5.7.8 Node.js und socket.io

*Node.js*<sup>13</sup> ist ein serverseitiges JavaScript-Framework und eine Laufzeitumgebung. Der benötigte Webserver wird ebenso gleich mitgeliefert. Mit Node.js wird dem Entwickler ermöglicht, JavaScript auf Seite des Servers auszuführen. Es gibt eine große Anzahl an Plugins, die dem Entwickler bereits fertige Funktionalitäten zur Verfügung stellen. Socket.io<sup>14</sup> ist solch ein Plugin, dessen Ziel es ist, eine Verbindung zwischen dem Web-Client und dem Server ohne größere Verzögerung unabhängig vom verwendeten Browser zur Verfügung zu stellen. [Kie11, Rau12]

## 5.8 Fehlerbehandlung

Im gesamten Programm wird die Fehlerbehandlung über die strukturierte Ausnahmebehandlung realisiert. Dabei werden ausschließlich die Ausnahmen der Standard-Bibliothek für eine bessere Interoperabilität verwendet. Der reibungslose Programmablauf wird besonders durch eine Ausnahmebehandlung im Widget-Launcher gewährleistet. Hierbei müssen im Lebenszyklus eines Widgets verschiedene Phasen unterschieden werden (siehe Abbildung 5.1), in denen Probleme auftreten können.

Zuerst erfolgt die Erstellung der Widgets. Sollte in dieser Phase bereits eine Ausnahme ausgelöst werden, so hat dies zur Konsequenz, dass das entsprechende Widget nicht in die Widget-Warteschlange aufgenommen wird. Das Programm läuft korrekt weiter und stellt nur die übrigen Widgets dar.

Als nächstes erfolgt der Start eines Widgets. Wird beim Start eine Ausnahme ausgelöst, wird dieser automatisch abgebrochen und der Start des nächsten Widgets aus der Warteschlange initiiert. Das fehlerhafte Widget wird trotzdem von neuem an das Ende der Widget-Warteschlange angefügt. Dies hat den Hintergrund, dass bis zum nächsten Startversuch des Widgets das Problem nicht mehr vorhanden sein könnte. Zum Beispiel könnte eine zunächst nicht erreichbare Datenbank wieder verfügbar sein. Daher sollte darauf geachtet werden, dass bestimmte Routinen beim Start und nicht bei der Erstellung des Widgets ablaufen. Alle Methoden, die schwerwiegende Probleme verursachen

---

<sup>13</sup><http://nodejs.org>

<sup>14</sup><http://socket.io>



## 5.8 Fehlerbehandlung

---

können, sollten beim Erzeugen des Widgets aufgerufen werden, die anderen Methoden beim Start. Ein schwerwiegendes Problem wäre zum Beispiel, dass die Schrift nicht geladen werden kann.

Die meisten Fehler können sich zwar beim Starten des Widgets ereignen, jedoch können auch nach dem erfolgreichen Start Ausnahmen auftreten. Der Widget-Launcher ruft lediglich zwei Funktionen eines Widgets auf: `update()` und `render()`. Diese werden ebenso wie der Start des Widgets behandelt. Beim Auftreten einer Ausnahme wird das Widget gestoppt und wieder zurück an das Ende der Warteschlange gesetzt.

Durch diese Behandlung der Widgets wird die Anwendung robuster, da Probleme in einem Widget nur dazu führen, dass ein anderes Widget dargestellt wird. Im Idealfall ist dies für den Betrachter nicht wahrnehmbar.

# 6 Tests

Für jede Anwendung ist es essenziell, dass in irgendeiner Art und Weise Tests erstellt und ausgewertet werden. In der Regel ist dabei eine Kombination von manuellen und automatischen Tests sinnvoll. Daher soll auch in der vorliegenden Anwendung auf diese Kombination gesetzt werden. Im Folgenden wird zuerst im Abschnitt 6.1 die Testumgebung und die schwierigen Umstände beim Testen einer Anwendung für eine Medienfassade beschrieben. Darauf aufbauend werden in Abschnitt 6.2 die verschiedenen durchgeführten Testarten erläutert, im Besonderen der Test für den Algorithmus für die Berechnung der Globalstrahlung.

## 6.1 Testumgebung

Die Anwendung wurde auf einem Mac Mini der 5. Generation und einem Mac Book getestet, jeweils unter Mac OS X 10.7. Elementar fürs Testen war dabei das Testlabor, welches im Nachstehenden näher beschrieben wird.

### 6.1.1 Testlabor

Da die Medienfassade im Zeitraum der Erstellung dieser Arbeit noch nicht fertig am Gebäude angebracht war, musste für das Testen der Anwendung auf ein Testlabor zurückgegriffen werden. In diesem steht sowohl eine Rückprojektion als auch zwei Reihen mit jeweils neun LED-Kacheln inklusive Controller zur Verfügung, die eine Pixelanzahl von  $72 \times 16$  ermöglichen (siehe Abbildung 6.1). Ein Komponente zum Testen des Ambient-Lights steht leider nicht zur Verfügung. Mit Hilfe dieser Hardware konnte die Anwendung auch ohne die fertige Medienfassade für diese entwickelt und getestet werden.

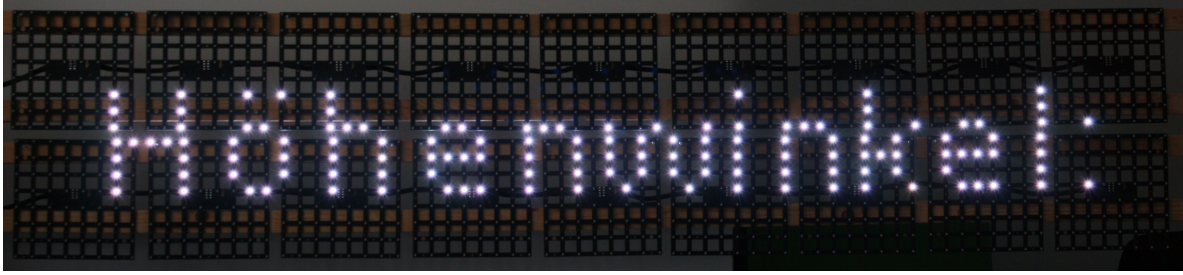
Besonders wichtig war das Testlabor für die Gestaltung der Oberfläche. Damit konnten verschiedene Kontraste getestet werden, denn aufgrund der Beschaffenheit des RGB-Farbraums (siehe Unterabschnitt 2.1.1) sehen die Farben auf dem Monitor anders als



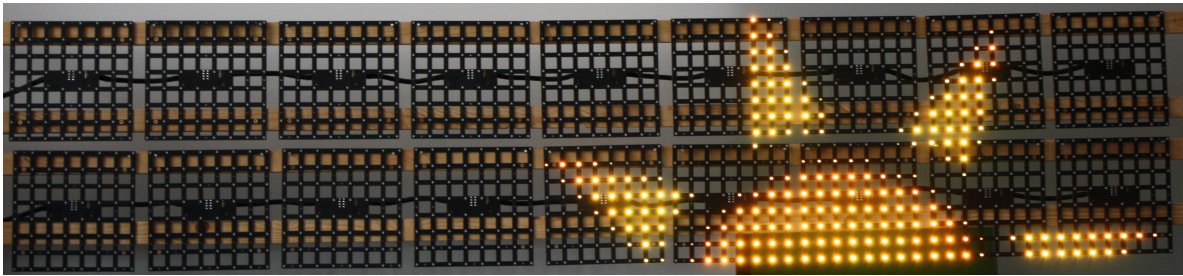
**Abbildung 6.1:** Aufbau des Testlabors mit Projektionsleinwand und LED-Kacheln

auf den LED-Kacheln aus. Zusätzlich stand auch eine Probe-Kachel der Glasfassade zur Verfügung, mit deren Hilfe auch die Farbwirkung hinter der Glasfassade getestet werden konnte.

Insgesamt konnte dank des Testlabors ein erster guter Eindruck geschaffen werden, wie die Anwendung auf der fertigen Fassade wirken wird. Allerdings lag die maximale Entfernung zu den LED-Kacheln räumlich bedingt bei nur 4,5 m; bei dem fertigen Gebäude wird der Betrachter voraussichtlich weiter entfernt stehen. Dies hatte im Testlabor zur Folge, dass durch den hohen Pixelabstand bei den LED-Kacheln die Details optisch nicht wie auf dem Monitor zum Vorschein kamen, auf dem die Pixeldichte sehr viel höher ist. Diese Problematik wurde dadurch gelöst, dass im Testlabor Fotos von der laufenden Anwendung gemacht wurden. Durch eine nachträgliche Skalierung auf dem Monitor wirkten die Pixel-Abstände zwischen den LEDs wieder kleiner - genauso, als ob der Betrachter weiter entfernt stehen würde. So war es möglich, einen ungefähren Eindruck zu bekommen, wie die Anwendung auf der fertigen Medienfassade aussehen wird. Abbildung 6.2 und Abbildung 6.3 zeigen Fotos der Anwendung im Testlabor. [Sch12] Ein weiterer wichtiger Punkt war das Testen der Schriftgröße. Einerseits sollte der begrenzte Platz gut genutzt werden und somit möglichst viel Text untergebracht werden, andererseits war es wichtig, dass dieser Text für den Betrachter gut lesbar bleibt. Verschiedene Tests haben gezeigt, dass für eine gute Lesbarkeit maximal 7 Zeilen mit höchsten 25 Zeichen je Zeile verwendet werden sollten (siehe Abbildung 6.2).



**Abbildung 6.2:** Darstellung der Schrift (Foto der Anwendung auf den LED-Kacheln im Testlabor)



**Abbildung 6.3:** Darstellung der Grafik (Foto der Anwendung auf den LED-Kacheln im Testlabor)

## 6.2 Durchgeführte Tests

Im Laufe der Entwicklung wurden sowohl automatisierte Modultests als auch manuelle Integrations- und Systemtests durchgeführt.

### 6.2.1 Modultests

Es gibt unzählige Frameworks für das Erstellen von Modultests in C++. Für diese Anwendung wurde Boost verwendet (siehe Unterabschnitt 5.7.2), welches neben vielen anderen Funktionalitäten auch über eine Komponente für das Testen verfügt. Die Entscheidung fiel auf Boost, da die Modultests leicht zu implementieren sind, der Funktionsumfang den Anforderungen genügt und keine zusätzliche Bibliothek genutzt werden muss, da Boost bereits fester Bestandteil der Anwendung ist.

Der Fokus der Modultests liegt auf den Komponenten, in denen Berechnungen stattfinden, da sich diese besonders zuverlässig mit Modultests abdecken lassen. Damit ein möglichst großes Spektrum an Tests stattfindet, wurden drei unterschiedliche Testszenarien mit verschiedenen Testdaten entwickelt:

1. Nachmittags im Dezember bei Schnee; Südwest-Seite der Fassade
2. Abends im Sommer bei freiem Himmel; Südost-Seite der Fassade
3. Vormittags im Frühling bei leicht bedecktem Himmel; Nordwest-Seite der Fassade

In Boost werden einzelne Testfälle definiert, die sich als Testsammlung gruppieren lassen. Für jede Komponente wurde dementsprechend eine Testgruppe (Test Suite) mit verschiedenen Testfällen definiert. Sobald die Testfälle fertig erstellt sind, können diese über sogenannte *Fixtures* mit unterschiedlichen Testdaten versorgt werden. Somit muss immer nur ein Testfall für beliebig viele verschiedene Testdaten definiert werden und Boost sorgt dafür, dass automatisch für jeden Satz an Testdaten der Testfall ausgeführt wird. Mehrere Testgruppen können wiederum zu einem Testmodul zusammengefasst werden. [OS09]

### 6.2.2 Integrations- und Systemtests

Bei der Fertigstellung eines Features wurde jeweils ein manueller Integrationstest für die Komponente vorgenommen. Zudem gab es beim Erreichen von wichtigen Teilzielen stets einen manuellen Systemtest. Außerdem fand zum Abschluss noch ein kompletter Systemtest statt, in dem geprüft wurde, ob alle Anforderungen erfüllt wurden.

### 6.2.3 Tests für den Algorithmus für die Berechnung der Globalstrahlung

Eine wesentliche Komponente der Anwendung ist die Prognose des erzeugten Stroms der PV-Anlage. Hierfür ist die Bestimmung der Globalstrahlung essenziell. Da die Wetterstation auch über einen Sensor für die Bestrahlungsstärke verfügt (siehe Abschnitt 3.2), kann die Genauigkeit der Berechnung geprüft werden. Dabei muss jedoch berücksichtigt werden, dass der Sensor horizontal zum Himmel ausgerichtet ist und dementsprechend nicht exakt dem Aufbau der Solarmodule entspricht.

Zum Testen wurden zwei Tage mit unterschiedlichen Wetterverhältnissen exemplarisch ausgewählt:

**04.12.2012** An diesem Tag gab es ausschließlich Schnee und Regen, der Himmel war dementsprechend sehr stark bedeckt.

**07.12.2012** Den kompletten Tag über herrschte nur eine leichte Bewölkung.

Zusätzlich wurde ein Testprogramm entwickelt, welches für einen gegebenen Tag und die jeweiligen Wetterverhältnisse für jede Minute des Tages die Globalstrahlung be-

rechnet. Dabei werden die in dieser Arbeit entwickelten Berechnungen inklusive des Einflusses des Wetters (siehe Unterabschnitt 4.4.1) verwendet. Mit Hilfe dieses Testprogramms und des Sensors für die Bestrahlungsstärke der Wetterstation konnte eine minutengenaue Gegenüberstellung des real gemessenen und des berechneten Werts für die Globalstrahlung erzeugt werden.

Die Abbildung 6.4 zeigt die gemessene und die berechnete Globalstrahlung für den 04.12.2012 im Vergleich. Auf der Y-Achse ist die Bestrahlungsstärke in  $\frac{\text{W}}{\text{m}^2}$  eingetragen, auf der X-Achse die Uhrzeit. Dabei ist zu sehen, dass für diesen Tag die Berechnung nahe den gemessenen Werten liegt. Ohne Berücksichtigung der Formel für den Wettereinfluss würden sich die berechneten Werte für die Globalstrahlung zur Mittagszeit im Bereich von  $600 \frac{\text{W}}{\text{m}^2}$  befinden. Allerdings wird auch sofort offensichtlich, dass gerade im Bereich des Sonnenauf- und Sonnenuntergangs die Abweichung größer wird. Eine Möglichkeit wäre es, dass die Werte nahe zum Sonnenauf- und Sonnenuntergang bei der Berechnung leicht abgeschwächt werden, damit eine eher parabelförmige Kurve entsteht. Wenn der Sonnenhöhenwinkel  $\gamma_S$  knapp über  $0^\circ$  ist, liegt ein Sonnenaufgang vor. Ist er hingegen bei etwas unter  $180^\circ$ , handelt es sich um einen Sonnenuntergang. Des Weiteren ist gut erkennbar, dass aufgrund der sich ändernden Wetterverhältnisse die realen Werte sehr viel stärkeren Schwankungen unterworfen sind. Um die Berechnungen zu verbessern, müsste der Bewölkungsgrad exakter und häufiger bestimmt werden. Dies kann mit einem Bewölkungssensor erfolgen, der zum Beispiel bei der Wetterstation montiert werden könnte.

Die Abbildung 6.5 zeigt die berechneten und gemessenen Werte für den 07.12.2012. Dieser Tag wies im Gegensatz zum 04.12.2012 eine deutlich geringere Bedeckung des Himmels und geringere Schwankungen bei den realen Werten auf. Die Abweichungen sind jedoch teilweise schon im Bereich von bis zu  $200 \frac{\text{W}}{\text{m}^2}$ . Das Diagramm zeigt ebenfalls, dass eine Berücksichtigung des Sonnenauf- und Sonnenuntergangs eine recht deutliche Verbesserung ermöglichen könnte.

Unter den gegebenen Umständen können die berechneten Werte für diese beiden Tage durchaus als akzeptabel betrachtet werden. Für eine genauere Analyse müssten jedoch mehr Tage mit verschiedenen Wetterbedingungen und zu unterschiedlichen Jahreszeiten betrachtet werden. Zusätzlich könnte für eine genauere Betrachtung die Fläche unter den Kurven berechnet werden, wodurch eine Vergleichszahl zur Verfügung stehen würde.

Wie erwartet kann das Ergebnis nur als eine ungefähre Prognose für die Globalstrahlung dienen. Insbesondere aufgrund des hohen Einflusses der Wetterbedingungen und

## 6.2 Durchgeführte Tests

des schwierigen Messens dieser, ist der Fehler teilweise recht hoch. Hier besteht durchaus noch Verbesserungspotenzial, zum Beispiel mit einer oder mehreren der hier vorgestellten Möglichkeiten.

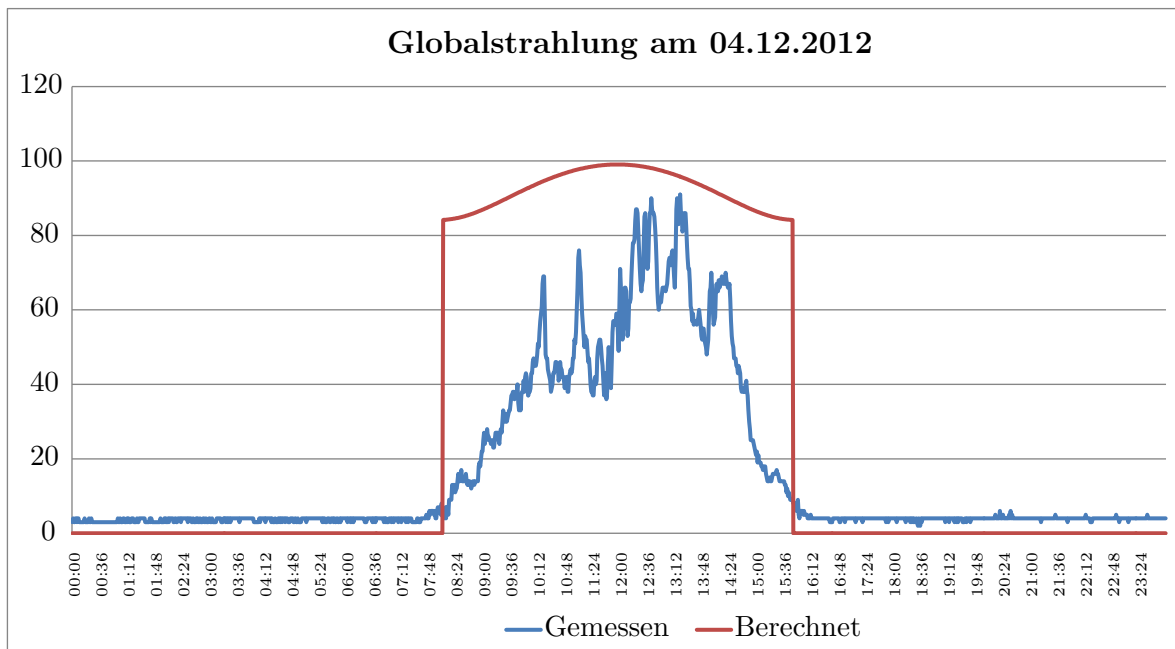


Abbildung 6.4: Gemessene und berechnete Globalstrahlung für den 04.12.2012

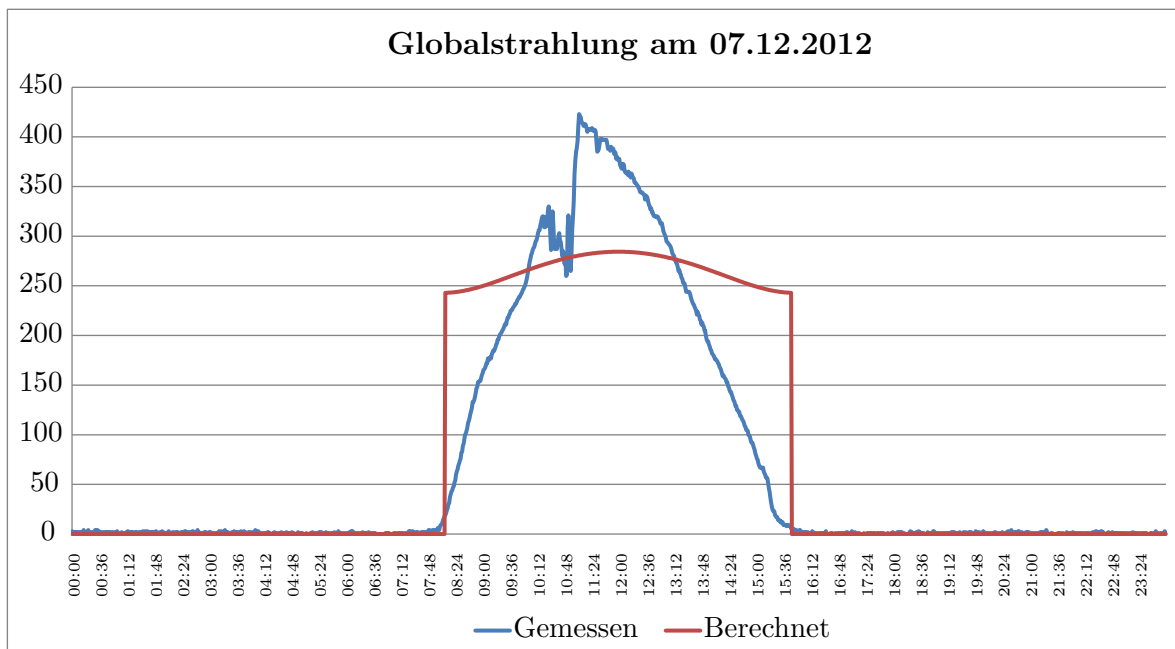


Abbildung 6.5: Gemessene und berechnete Globalstrahlung für den 07.12.2012

# 7 Ergebnis

In diesem Kapitel sollen die wichtigsten Aspekte der Arbeit noch einmal zusammengefasst sowie ein Fazit aus ihnen gezogen werden. Abschließend folgt ein kurzer Ausblick auf weitere Entwicklungsmöglichkeiten.

## 7.1 Fazit

Das primäre Ziel der Arbeit wurde erreicht: Die Daten aus verschiedenen Quellen, wie zum Beispiel der Wetterstation und unterschiedlicher virtueller Sensoren, wurden ausgewertet, kombiniert und auf einer Medienfassade visualisiert. Dabei fand zunächst eine ausführliche Analyse des FKI-Gebäudes, der Medienfassade und der möglichen realen und virtuellen Sensoren für das Gebäude statt. In diesem Zusammenhang wurde gezeigt, welche der Sensoren einzeln oder in Kombination mit anderen sinnvoll nutzbar für eine Visualisierung sind. Dabei hat sich besonders eine Verknüpfung als äußerst interessant herausgestellt: die Daten der Wetterstation in Kombination mit denen eines virtuellen Wettervorhersage-Sensors für die Berechnung einer Prognose des erzeugten Stroms der PV-Anlage des Gebäudes. Diese berechneten Daten können sehr gut als Basis für weitere Technologien, besonders im Sinne von *Green IT*, verwendet werden. Auf der Analyse aufbauend erfolgte ein umfassender Entwurf der Anwendung für die Visualisierung der Daten. Zunächst wurde die benötigte Infrastruktur beschrieben und anschließend wurden Entwürfe für die möglichen Widgets dargestellt. Dabei war ein Kernproblem der Gestaltung, den geringen Platz auf den LED-Kacheln so gut wie möglich zu nutzen. Hierfür wurden umfangreiche Tests im Testlabor getätigt, um die Entwürfe den Gegebenheiten der Medienfassade anzupassen. So wurden im Laufe der Arbeit verschiedene Probleme identifiziert und Lösungswege für diese aufgezeigt. Des Weiteren erfolgte aufbauend auf den Grundlagen ein Entwurf für den Algorithmus für die Berechnung der Stromprognose der PV-Anlage. Für diesen galt es weitere Aspekte, wie der Einfluss des Wetters oder die Bodenreflexion, genau zu betrachten und in der Berechnung zu berücksichtigen. Besonders der Einfluss des Wetters hat sich als



schwieriges Kriterium herausgestellt, welches in diesem Fall unter Zuhilfenahme der vorhandenen Datenbasis in zufriedenstellender Art und Weise gelöst werden konnte. Darüber hinaus wurden Vorschläge für die weitergehende Nutzung dieser Prognose in Bezug auf das Gebäude dargelegt.

Die Funktionsweise des Entwurfs wurde in der Implementierung eines Prototyps gezeigt. In diesem Zusammenhang wurde eine Anwendung für die LED-Kacheln der Medienfassade entwickelt, welche die vier implementierten Widgets wechselnd darstellen kann. Es existiert ein Widget für das aktuelle Wetter, welches die Daten der Wetterstation visualisiert, ein weiteres für die Wetterprognose für die nächsten drei Tage und ein Widget für die Berechnung der Prognose des Stroms der PV-Anlage. Zudem gibt es noch ein Widget für die Interaktion mit dem Dashboard: Der Nutzer hat die Möglichkeit über einen mobilen Client die Sonnenposition für einen gegebenen Zeitpunkt abzufragen. Somit ist die Basis für weitere sowohl rein informative als auch interaktive Widgets gegeben. Die Anwendung ist in der vorliegenden Form bereits vollständig an der Medienfassade einsatzfähig und bietet somit auch eine Grundlage für weitere Programme. Dabei wurden sämtliche in der Anforderungsanalyse definierten Pflichtziele erreicht und zusätzlich auch das optionale Ziel, die Möglichkeit der Interaktion, realisiert.

Um ein gutes Maß an Qualitätssicherung zu erreichen, wurden abschließend verschiedene automatisierte und manuelle Tests am Prototyp durchgeführt. Außerdem erfolgte eine Auswertung des Prognose-Algorithmus. Diese zeigte bereits gute Ergebnisse der Berechnungen und machte gleichzeitig das noch vorhandene Optimierungspotential deutlich, sodass konkrete Vorschläge zur weiteren Verbesserung gemacht werden konnten. Insgesamt ist eine einsatzfähige Anwendung für die Medienfassade entstanden, welche die Nutzung, Kombination und Visualisierung verschiedener Sensorquellen zeigt und darüber hinaus einen vielversprechenden Ausgangspunkt für weitere Entwicklungen darstellt.

## 7.2 Ausblick

Im Laufe der Arbeit wurde die Basis für verschiedene Weiterentwicklungen geschaffen. Für die Anwendung ist die Implementierung weiterer Widgets interessant, hierbei wurden einige Vorschläge bereits im Entwurf vorgestellt. Das Programm wurde so konzipiert, dass es leicht ist, weitere Widgets zu implementieren und in das Programm zu integrieren. Die Grundlage dafür ist die Nutzung von Vererbung, eine auf die Erweiterung zugeschnittene Modellierung und eine Strukturierung in Komponenten.

## 7.2 Ausblick

---

Des Weiteren gibt es noch mehrere Punkte, die direkt an der Anwendung verbessert werden könnten: zum Beispiel die Integration eines Rollenkonzepts für die Interaktion oder eine einfachere Konfigurationsmöglichkeit der Anwendung in Form einer Webseite. Darüber hinaus könnte eine weitere Optimierung der Berechnung der Prognose für die PV-Anlage interessant sein. Insbesondere der Einfluss des Wetters ist ein wichtiger Faktor, der viel Verbesserungspotenzial bietet. Neben den bereits vorgestellten Aufwertungen wäre auch die Umsetzung eines Lernalgorithmus denkbar. Da in Zukunft die Daten der PV-Anlage zur Verfügung stehen werden, könnte ein direkter Vergleich der berechneten und tatsächlichen Werte erfolgen. Mit Hilfe dieser Datenbasis könnten weitere Korrekturen an der Berechnung vorgenommen werden. Eine andere Option wäre die exaktere Bestimmung des Verlustfaktors der PV-Anlage. Dies würde ebenfalls genauere Ergebnisse erzielen, die Wirkung wäre jedoch im Vergleich zu einer Optimierung des Einflusses des Wetters gering.

Sobald die Daten der PV-Anlage und des Smart Metering-Systems zur Verfügung stehen, wären diese interessante Ausgangspunkte für neue Visualisierungen. Außerdem wäre die Erweiterung der Anwendung auf die zusätzlichen Komponenten der Medienfassade, das Ambient-Light und unter Umständen sogar die Rückprojektion denkbar. Insgesamt bietet das Projekt Potential für eine Vielzahl aufbauender Entwicklungen.

# Abbildungsverzeichnis

2.1	Die Visualisierungspipeline [Krö00]	5
2.2	Astronomisches Koordinatensystem des Horizonts	11
2.3	Beispiel für den Höhenwinkel $\gamma$	12
2.4	Das Äquator-Koordinatensystem [Wet09b]	13
2.5	Zusammenhang zwischen Stundenmaß und Stundenwinkel	13
2.6	Die drei Bestandteile der Globalstrahlung: direkte, diffuse und reflektierte Strahlung [Mel97, Seite 13]	19
2.7	Minimaler Aufbau einer Photovoltaik-Anlage [Fri12, Seite 1]	22
3.1	Position des FKI-Gebäudes auf dem Campus der HTW Berlin	26
3.2	Visualisierung der Medienfassade am Modell des FKI-Gebäudes	27
3.3	Foto des FKI-Gebäudes (29.01.2013)	28
3.4	Position des FKI-Gebäudes mit Azimutwinkel der Fassadenseiten	29
3.5	Aufbau der Solarmodule an der Fassade des FKI-Gebäudes	30
3.6	Use-Case-Diagramm für das Dashboard	33
4.1	Systemarchitektur der Anwendung	41
4.2	Auszug des Klassendiagramms der Anwendung	43
4.3	Aufteilung der Fläche der LED-Wand für zwei Widgets	43
4.4	Entwurf des Widgets für den prognostizierten Strom der PV-Anlage	45
4.5	Entwurf des Widgets für die Sonnenposition zu einer bestimmten Zeit	46
4.6	Entwurf des Widgets für die aktuellen Wetterdaten der Wetterstation	47
4.7	Die verschiedenen Symbole für den Wetterzustand	48
4.8	Entwurf des Widgets für die Wettervorhersage	49
4.9	Skizze des Widgets für die Abfahrtszeiten des ÖPVN	50
4.10	Skizze für das Twitter-Widget	51
4.11	Visualisierung der Kommunikation zwischen dem mobilen Client und der Dashboard-Anwendung	52

5.1	Lebenszyklus eines Widgets . . . . .	65
5.2	Widget-Warteschlange beim Widget-Launcher . . . . .	66
5.3	Screenshot der Widgets „Wetterprognose“ und „Aktuelles Wetter“ . . .	67
5.4	Screenshot der Widgets „Sonnenposition“ und „Prognose des erzeugten Stroms“ . . . . .	67
5.5	Klassendiagramm für die Berechnung der Stromprognose . . . . .	68
5.6	Screenshot der Eingabemöglichkeit über das Smartphone . . . . .	69
6.1	Aufbau des Testlabors mit Projektionsleinwand und LED-Kacheln . . .	78
6.2	Darstellung der Schrift (Foto der Anwendung auf den LED-Kacheln im Testlabor) . . . . .	79
6.3	Darstellung der Grafik (Foto der Anwendung auf den LED-Kacheln im Testlabor) . . . . .	79
6.4	Gemessene und berechnete Globalstrahlung für den 04.12.2012 . . . . .	82
6.5	Gemessene und berechnete Globalstrahlung für den 07.12.2012 . . . . .	82

# Literaturverzeichnis

- [Abd11] ABDULAZEEZ, M. A.: *Artificial Neural Network Estimation of Global Solar Radiation Using Meteorological Parameters in Gusau, Nigeria*. In: *Scholars Research Library*, Seiten 586–595, 2011.
- [Ack06] ACKERMANN, N.: *Lichttechnik*. Oldenbourg Industrieverlag, 2006. ISBN 978-3835670426.
- [Böh] BÖHNEL, CLEMENS: *jQuery Mobile*. Webseite: <http://www.comspace.de/aktuelles/fachthemen-jQuery-mobile.html>. Zuletzt besucht am 15.01.2013.
- [BS10] BABOO, DR. S. SANTHOSH und I. KADAR SHEREEF: *An Efficient Weather Forecasting System using Artificial Neural Network*. In: *International Journal of Environmental Science and Development*, Band 1, Seiten 321–326, 2010.
- [BS12] BERGEMANN, STEPHAN und ROBIN SCHLEGEL: *Medienfassade - Möglichkeiten von Visualisierung und Interaktion*. Webseite: [http://inka.htw-berlin.de/Sieck/Stud\\_Projekte/bergemann\\_schlegel\\_fp2.pdf](http://inka.htw-berlin.de/Sieck/Stud_Projekte/bergemann_schlegel_fp2.pdf), 2012. Zuletzt besucht am: 19.10.2012.
- [Bun] BUNDESMINISTERIUM FÜR WIRTSCHAFT UND TECHNOLOGIE: *Was ist Green-IT?* Webseite: <http://www.green-it-wegweiser.de/Green-IT/Navigation/Basisinfos/was-ist-green-it.html>. Zuletzt besucht am 28.01.13.
- [car11] CARATART E.V.: *3D Mapping Videoprojektion*. Webseite: <http://www.caratart.de/2011/05/02/3d-mapping-videoprojektion/>, 2011. Zuletzt besucht am 14.11.2012.
- [CF99] CRUMMENERL, R. und W. FREITAG: *Das Wetter*. Was ist was. Tessloff, 1999. ISBN 978-3788602475.

- [Drü12] DRÜCK, HARALD: *Solarthermie I*. Webseite: [www.itw.uni-stuttgart.de/lehre/lehrveranstaltungen/Dokumente/solar1/solvo\\_teil1.pdf](http://www.itw.uni-stuttgart.de/lehre/lehrveranstaltungen/Dokumente/solar1/solvo_teil1.pdf), 2012.
- [DS98] DAWSON, F. und D. STENERSON: *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*. Webseite: <http://www.ietf.org/rfc/rfc2445.txt>, 1998. Zuletzt besucht am 16.01.2013.
- [FM11] FETTE, I. und A. MELNIKOV: *The WebSocket Protocol*. Webseite: <https://tools.ietf.org/rfc/rfc6455.txt>, 2011. Zuletzt besucht am 16.01.2013.
- [Fre] FREITAG TECHNOLOGIES GMBH: *GRID25*. Webseite: <http://www.ledmedia.de>. Zuletzt besucht am: 01.10.2012.
- [Fri12] FRIEDRICH, MARKUS: *Photovoltaik - Teil 1: Technische Grundlagen*. Webseite: [www.friedrich-datentechnik.de/download/basiswissen/Basiswissen-Photovoltaik.pdf](http://www.friedrich-datentechnik.de/download/basiswissen/Basiswissen-Photovoltaik.pdf), 2012. Zuletzt besucht am: 19.10.2012.
- [Geb] GEBHARDT, THOMAS: *Astronomische Koordinatensysteme*. Webseite: <http://www.zum.de/Faecher/Materialien/gebhardt/astronomie/koordsys.html>. Zentrale für Unterrichtsmedien im Internet.
- [Hei] HEINDL SERVER GMBH: *Photovoltaik: Solarstrom und Solarzellen in Theorie und Praxis*. Webseite: <http://www.solarserver.de/wissen/basiswissen/photovoltaik.html>. Zuletzt besucht am 17.10.2012.
- [Hol02] HOLZINGER, ANDREAS: *Basiswissen Multimedia*. Vogel Fachbuch, Zweite Auflage, 2002. ISBN 3802319141.
- [Ite] ITEN, KARL: *Klimadiagramm weltweit*. Webseite: <http://www.iten-online.ch/klima/europa/deutschland/berlin.htm>. Zuletzt besucht am: 21.11.2012.
- [ITWa] ITWISSEN: *CSV*. Webseite: <http://www.itwissen.info/definition/lexikon/comma-separated-values-CSV.html>. Zuletzt besucht am 15.01.13.
- [ITWb] ITWISSEN: *Datenlogger*. Webseite: <http://www.itwissen.info/definition/lexikon/Datenlogger-data-logger.html>. Zuletzt besucht am 15.01.13.

- [ITWc] ITWISSEN: *DirectX*. Webseite: <http://www.itwissen.info/definition/lexikon/DirectX.html>. Zuletzt besucht am 17.01.13.
- [ITWd] ITWISSEN: *JSON*. Webseite: <http://www.itwissen.info/definition/lexikon/JSON-JavaScript-object-notation.html>. Zuletzt besucht am 16.01.13.
- [ITWe] ITWISSEN: *Neuronales Netz*. Webseite: <http://www.itwissen.info/definition/lexikon/Neuronales-Netz-artificial-neural-networks.html>. Zuletzt besucht am 16.01.13.
- [ITWf] ITWISSEN: *REST*. Webseite: <http://www.itwissen.info/definition/lexikon/representational-state-transfer-REST.html>. Zuletzt besucht am 15.01.13.
- [ITWg] ITWISSEN: *RSS*. Webseite: <http://www.itwissen.info/definition/lexikon/RSS-Feed-RSS-really-simple-syndication.html>. Zuletzt besucht am 16.01.13.
- [ITWh] ITWISSEN: *SOAP*. Webseite: <http://www.itwissen.info/definition/lexikon/simple-object-access-protocol-SOAP-SOAP-Protokoll.html>. Zuletzt besucht am 21.01.13.
- [ITWi] ITWISSEN: *XML*. Webseite: <http://www.itwissen.info/definition/lexikon/extended-markup-language-XML.html>. Zuletzt besucht am 16.01.13.
- [Kie11] KIESSLING, MANUEL: *The Node Beginner Book*. Leanpub, 2011. ISBN 1471628442.
- [Kle05] KLEIN, MICHAEL: *Der Tod des Schneemanns*. Webseite: <http://www.donnerwetter.de/news/5403/Der-Tod-des-Schneemanns>, 2005. Zuletzt besucht am: 21.11.2012.
- [Kon08] KONRAD, FRANK: *Planung von Photovoltaik-Anlagen: Grundlagen und Projektierung*. Vieweg+Teubner Verlag, Zweite Auflage, 2008. ISBN 978-3834805867.
- [KRMR07] KRISHNAIAH, T., S. SRINIVASA RAO, K. MADHUMURTHY und K.S. REDDY: *Neural Network Approach for Modelling Global Solar Radiation*. In: *Journal of Applied Sciences Research*, Seiten 1105–1111, 2007.

- [Krö00] KRÖMKER, DETLEF: *Visualisierung*. Webseite: [http://www.gdv.informatik.uni-frankfurt.de/lehre/ws2000/Folien/Visualisierung/visualisierung03Visualisierungsprozess\\_V1.pdf](http://www.gdv.informatik.uni-frankfurt.de/lehre/ws2000/Folien/Visualisierung/visualisierung03Visualisierungsprozess_V1.pdf), 2000. Zuletzt besucht am: 23.11.2012.
- [Lan] LANTINGA, SAM: *SDL\_ttf 2.0*. Webseite: [http://www.libsdl.org/projects/SDL\\_ttf](http://www.libsdl.org/projects/SDL_ttf). Zuletzt besucht am 01.10.2012.
- [LE] LANTINGA, SAM und MATTIAS ENGDEGÅRD: *SDL\_image 1.2*. Webseite: [http://www.libsdl.org/projects/SDL\\_image/](http://www.libsdl.org/projects/SDL_image/). Zuletzt besucht am 11.01.2013.
- [Mal06] MALBERG, H.: *Meteorologie und Klimatologie: Eine Einführung*. Springer-Lehrbuch. Springer, 2006. ISBN 978-3540372196.
- [Mel97] MELISS, MICHAEL: *Regenerative Energiequellen*. Springer Verlag, 1997. ISBN 3540632182.
- [Mül05] MÜLLER, OTFRIED: *Chemie einfach und verständlich*. Books on Demand, 2005. ISBN 978-3833421785.
- [Neu04] NEUHAUS, CHRISTIAN: *Der Wind*. Webseite: <http://www.top-wetter.de/kolumne/maerz2004.htm>, 2004. Zuletzt besucht am 17.10.2012.
- [OAu] OAUTH: *OAuth*. Webseite: <http://oauth.net/>. Zuletzt besucht am 21.11.2012.
- [OS09] OSTERMEIER, DANIEL und JASON SANKEY: *C++ Unit Testing With Boost.Test*. Webseite: <http://www.alittlemadness.com/2009/03/31/c-unit-testing-with-boosttest/>, 2009. Zuletzt besucht am 19.12.2012.
- [Pola] POLAK, FRANZY: *Wetterlexikon - Bewölkungsgrad*. Webseite: <http://www.wetter.net/lexikon/bewoelkungsgrad.html>. Zuletzt besucht am 29.10.2012.
- [Polb] POLAK, FRANZY: *Wetterlexikon - Lufttemperatur*. Webseite: <http://www.wetter.net/lexikon/lufttemperatur.html>. Zuletzt besucht am 29.10.2012.
- [Polc] POLAK, FRANZY: *Wetterlexikon - Niederschlag*. Webseite: <http://www.wetter.net/lexikon/niederschlag.html>. Zuletzt besucht am 29.10.2012.



- [QE] QUASCHNING, VOLKER und MATTHIEU EBERT: *HTW Wetterstation*. Webseite: <http://wetter.htw-berlin.de/wetterstation.php>. Zuletzt besucht am: 19.10.2012.
- [Qua09] QUASCHNING, VOLKER: *Regenerative Energiesysteme*. Hanser Verlag, Sechste Auflage, 2009. ISBN 978-3446421516.
- [Qua12] QUASCHNING, VOLKER. Persönliche Mitteilung, Oktober 2012.
- [Rau12] RAUCH, GUILLERMO: *Introducing Socket.IO*. Webseite: <http://socket.io/>, 2012. Zuletzt besucht am 11.01.2013.
- [Riv92] RIVEST, R.: *The MD5 Message-Digest Algorithm*. Webseite: <http://www.ietf.org/rfc/rfc1321.txt>, 1992. Zuletzt besucht am 16.01.2013.
- [Rou09] ROUSE, MARGARET: *Remote Procedure Call (RPC)*. Webseite: <http://searchsoa.techtarget.com/definition/Remote-Procedure-Call>, 2009. Zuletzt besucht am 21.01.13.
- [Rut] RUTKOWSKI, HUBERT: *SDL\_Config*. Webseite: [http://koshmaar.linuxpl.info/sdl\\_config/](http://koshmaar.linuxpl.info/sdl_config/). Zuletzt besucht am 11.01.2013.
- [SCH11] SCHÜCO: *Bedienungsanleitung Schüco Sunalyzer Web PR*, 2011.
- [Sch12] SCHLEGEL, ROBIN: *Spiele als Multiuser-Anwendungen für Medienfassaden*. Webseite: [http://inka.htw-berlin.de/Sieck/Stud\\_Projekte/schlegel\\_fp2.pdf](http://inka.htw-berlin.de/Sieck/Stud_Projekte/schlegel_fp2.pdf), 2012. Zuletzt besucht am: 19.10.2012.
- [SDL] SDL: *SDL*. Webseite: <http://www.libsdl.org/>. Zuletzt besucht am 01.10.2012.
- [SLM] SAM LANTINGA, ROAY WOOD und MASAHIRO MINAMI: *SDL\_net 1.2*. Webseite: [http://www.libsdl.org/projects/SDL\\_net](http://www.libsdl.org/projects/SDL_net). Zuletzt besucht am 01.10.2012.
- [Sob] SOBCZAK, MACIEJ: *SOCI - The C++ Database Access Library*. Webseite: <http://soci.sourceforge.net/>. Zuletzt besucht am 19.12.2012.
- [Ste] STENBERG, DANIEL: *libcurl - the multiprotocol file transfer library*. Webseite: <http://curl.haxx.se/libcurl/>. Zuletzt besucht am 19.12.2012.

- [UB02] UNSÖLD, A. und B. BASCHEK: *Der neue Kosmos: Einführung in die Astronomie und Astrophysik*. Physics and astronomy. Springer, 2002. ISBN 978-3540421771.
- [Wet09a] WETZEL, S.: *Astronomisches Koordinatensystem des Horizonts*. Webseite: <http://commons.wikimedia.org/wiki/File:HorSys.svg>, 2009. Zuletzt besucht am 29.11.2012.
- [Wet09b] WETZEL, S.: *Astronomisches Koordinatensystem des Äquators, ortsfest*. Webseite: <http://commons.wikimedia.org/wiki/File:%C3%84quSys0rt.svg>, 2009. Zuletzt besucht am 29.11.2012.
- [WFH11] WITTEN, IAN H., EIBE FRANK und MARK A. HALL: *Data Mining*. Morgan Kaufmann, Dritte Auflage, 2011. ISBN 978-0123748560.
- [Wil11] WILKINSON, JOHN W.: *JSON Spirit: A C++ JSON Parser/Generator Implemented with Boost Spirit*. Webseite: <http://www.codeproject.com/Articles/20027/JSON-Spirit-A-C-JSON-Parser-Generator-Implemented>, 2011. Zuletzt besucht am 19.12.2012.
- [WSF03] WALCH, D., K. SCHWANKE und H. FRATER: *Wetter und Klima: Das Spiel der Elemente; atmosphärische Prozesse verstehen und deuten*. Phänomene der Erde. Springer, 2003. ISBN 978-3540008392.

# Glossar

**CSV** (Comma Separated Values) ist ein Textformat, bei dem die einzelnen Texte einer Zeile durch Kommas oder Semikolons getrennt werden [ITWa]. 32, 47, 57

**Datenlogger** dienen zur Erfassung von physikalischen Messwerten. Die Messwerte werden im Datenlogger gespeichert und können so über einen längeren Zeitraum ausgewertet werden [ITWb]. 31

**DirectX** ist die von Microsoft für Windows entwickelte Programmierschnittstelle für interaktive Technologien, wie zum Beispiel grafische Funktionen, Sound, Multimedia und Streaming [ITWc]. 72

**DMX** (Digital Multiplex) ist ein digitales Multiplex-Signal zum Übertragen von Steuereinformationen, welches primär in der Lichttechnik eingesetzt wird [Ack06]. 28, 42

**Flash-Socket** ist ein Objekt in ActionScript, mit dessen Hilfe eine Socket-Verbindung aufgebaut werden kann. 53

**FreeType 2** ist eine offene und kostenlose Bibliothek, um Schriften zu rendern. 72

**Green IT** bezeichnet den Einsatz von Informations- und Kommunikationstechnologien und deren Anwendung, die zu einer deutlichen Entlastung der Umwelt führen. [Bun]. 83

**iCalendar** ist ein standardisiertes Dateiformat zum Austausch von Kalenderinhalten [DS98]. 35, 51

**JSON** (JavaScript Object Notation) ist ein kompaktes Format zum Austausch von Daten [ITWd]. 49, 73

**MD5** (Message-Digest Algorithm 5) ist ein Algorithmus, der als Eingabe eine Nachricht beliebiger Länge nimmt und als Ausgabe einen dazu passenden 128 Bit Fingerabdruck liefert [Riv92]. 74

**Neuronales Netz** ist ein Computermodell, das sich an der Denkweise des Menschen orientiert und die Informationsverarbeitung, die Speicherung der Informationen und den Lernprozess des menschlichen Gehirns nachbildet [ITWe]. 37

**Remote Procedure Call** (RPC) ist ein Protokoll, das einer Anwendung ermöglicht, auf Prozeduren auf einem anderen Computer zuzugreifen [Rou09]. 70

**REST** (Representational State Transfer) ist ein Architekturstil, mit dem Webservices realisiert werden können [ITWf]. 32, 35, 47–49, 51, 74

**RSS-Feed** ist eine textbasierte Datei im XML-Format, die aus einer Liste von Einträgen besteht. Der RSS-Feed wird vor allem von Online-Redaktionen für eine strukturierte Veröffentlichungen von Änderungen auf der Webseite genutzt [ITWg]. 35, 51, 52

**SOAP** (Simple Object Access Protocol) ist ein Kommunikationsprotokoll, mit dessen Hilfe Remote Procedure Calls über das Internet ausgeführt werden können [ITWh]. 70

**Websockets** ermöglichen eine bidirektionale Kommunikation zwischen einer Webclient-Anwendung und einer Webserver-Anwendung [FM11]. 52, 53

**XML** (Extensible Markup Language) ist eine erweiterbare Auszeichnungssprache und dient der Beschreibung sowie dem Austausch von komplexen Datenstrukturen [ITWi]. 49

# Eigenständigkeitserklärung

„Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel verfasst habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.“

Berlin, den 1. Februar 2013

Robin Schlegel