

AEMS – Advanced Energy Monitoring System

Spezifikation

**Version 2.0**

abgenommen durch: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ am \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Verteiler:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fa. / Rolle | Name | Telefon | E-Mail | Bemerkungen |
| Projektleiter | Lukas Knoll | 0650/6780567 | knolli398@gmail.com |  |
| Projektmitarbeiter | Niklas Graf | 0664/2071911 | grafniklas98@gmail.com |  |
| Projektmitarbeiter | Sebastian Mandl | 0664/73480126 | sebastian-mandl@gmx.net |  |
| Betreuungslehrer | Joseph Doppelbauer | - | - |  |
| Auftraggeber | Ing. Herbert Pölzlberger | 07272/500531 | [poelzlberger@regef.at](mailto:poelzlberger@regef.at) |  |

Historie der Dokumentversionen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Datum | Autor | Status | Kap. / Änderungen |
| 0.1 | 08.07.2017 | Lukas Knoll | Nicht freigegeben | Ersterstellung |
| 1.0 | 10.07.2017 | Lukas Knoll | freigegeben | Überarbeitung |
| 1.1 | 20.07.2017 | Lukas Knoll | Nicht freigegeben | Adaptionen |
| 2.0 | 20.07.2017 | Lukas Knoll | freigegeben | Abnahme durch Arbeitgeber |

Inhaltsverzeichnis

[Historie der Dokumentversionen 2](#_Toc488393896)

[Inhaltsverzeichnis 2](#_Toc488393897)

[1 Einleitung 3](#_Toc488393898)

[1.1 Kern des Projektinhaltes 3](#_Toc488393899)

[1.2 Zweck dieses Dokuments 5](#_Toc488393900)

[2 Anforderungen 6](#_Toc488393901)

[2.1 Use Case Diagramm 6](#_Toc488393902)

[2.2 Bildschirmmasken 7](#_Toc488393903)

[2.3 Rahmenanforderungen 7](#_Toc488393904)

[2.4 Anforderungen Seitenaufbau-Webinterface 7](#_Toc488393905)

[2.5 Anforderungen Aufbau Android-Applikation 7](#_Toc488393906)

[2.6 Anforderungen Datenbank 7](#_Toc488393907)

[2.7 Anforderungen Statistiken und Berichte 8](#_Toc488393908)

[2.8 Anforderungen Report-Bot 8](#_Toc488393909)

[2.9 Anforderungen Logindaten 8](#_Toc488393910)

[2.10 Anforderungen Wasser-, Gaszähler 8](#_Toc488393911)

[2.11 Anhang 9](#_Toc488393912)

[2.11.1 Bildschirmmasken Webinterface 9](#_Toc488393913)

[2.11.2 Bildschirmmasken Android-Application 12](#_Toc488393914)

[2.11.3 Abkürzungen 13](#_Toc488393915)

# Einleitung

## Kern des Projektinhaltes

**Ausgangslage**

Zurzeit ist es in öffentlichen Gebäuden nur möglich die Daten eines einzelnen Stromzählers über ein Webinterface anzuzeigen. Jedoch gibt es keine Möglichkeit, gleich alle Zähler zusammenzufassen und Gesamtstatistiken über diese anzuzeigen.

Weiters ist es nur sehr umständlich möglich sich die Zählerdaten am Smartphone anzusehen, da das Webinterface dafür nicht optimiert ist.

**Anwender des Systems**

Genutzt werden soll das System vorrangig von Betrieben und öffentlichen Gebäuden. Auch die Verwendung durch Privathaushalte ist möglich. Voraussetzung dafür ist das Vorhandensein eines AMIS-Zählers.

**Verwendungszweck**

Gemacht wird das Projekt, um eine Übersicht über seine Zählerdaten zu erhalten. Vorrangig geht es hier um Stromdaten aus den AMIS-Zählern. Weiters sollen jedoch auch Daten von Wasser-, und Gaszählern ausgewertet werden können, um sich dessen Verbräuche anzusehen und mit den Werten aus vorhergegangenen Perioden zu vergleichen.

**Verbesserung zur Ausgangsituation**

Zurzeit ist es nur möglich sich Daten eines einzelnen Zählers anzusehen. Durch AEMS soll es möglich sein die Daten einer frei wählbaren Anzahl von Zählern zu komprimieren und sich Gesamtstatistiken, Auswertungen und Berichte anzeigen zu lassen.

**Zielsetzung**

Das Ziel des Projekts ist es einen leichteren Überblick über seine Zählerdaten zu bekommen und automatische Berichte und Statistiken erstellen zu können um die Werte mit den Vorjahren oder anderen Gebäuden zu vergleichen.  
Weiters soll das Einsehen der Zählerdaten auch für unterwegs durch die App erleichtert werden. Warnungen und Benachrichtigungen sollen bei zu hohem Energieverbrauch im Webinterface und am Smartphone angezeigt werden.

**Aufgaben/Plicht-Ziele**

* Entwicklung eines dynamischen Webinterfaces, welches vom Nutzer konfiguriert werden kann.
* Entwicklung einer Mobile-App für Android
* Entwicklung eines Bots, welcher die Daten der einzelnen Stromzähler über das Webinterface ausliest, diese komprimiert und in der Datenbank einträgt (viertelstündliche Werte)
* Erstellung von frei konfigurierbaren Statistiken
* Reports über selbst gewählte Perioden, mit selbst gewählten Zählern. Ausreißer werden auf einer Extraseite angeführt, um diese nicht zu übersehen. Algorithmen für Anomalien (z.B: Stromverbrauch abhängig von Außentemperatur) können eingestellt werden.
* Datenbankerstellung
* Datenbankadministration
* Komprimierung der Daten in der Datenbank
* Konfiguration von Warnungen/Alamierungen --> Diese können fix vergeben werden, bei Überschreitung eines gewissen Wertes, oder prozentuell (Sprich wenn Verbrauch zu einem gewissen Grad über dem Durchschnittswert liegt.)
* Ausnahmen für Benachrichtigungen und Warnungen können für bestimmte Tage konfiguriert werden.
* Schnittstelle für andere Zählertypen (Wasserzähler, Gaszähler, Wärmemengenzähler)
* Das System kann Verlaufs-Werte von Zählern (Strom, Gas, Wasser, Wärmemengenzähler – frei definierbar) und auch von anderen physikalischen Größen (Temperatur, Luftfeuchtigkeit, usw. – frei definierbar) speichern = Speicherpunkte
* Speicherpunkte werden auf einer Übersichtsseite mit Filtermöglichkeit (Art, Medium, Standort, …) angezeigt, dort können sie angelegt, editiert und gelöscht werden.
* Speicherpunkt enthalten einstellbare Parameter: z.B:
  + Art: Zählwert, Speicherwert
  + Medium: Strom, Gas, Wasser, …
  + Einheit: kWh, m³, kg, %, …
  + Erzeugungsdatum: wann wurde der Speicherpunkt angelegt.
  + Anmerkung: freier Text
  + Standort: Adresse, Gebäudeteil, Ansprechperson, …
  + Zugangsdaten: Zugriff auf die Werte (Website, Raspberry, …)
  + Intervall: Leseintervall, ev. Lesezeitpunkt (wenn Daten z.B. immer um 23:00 Uhr gelesen werden sollen
  + Grenzwerte: Warnung, Alarmwerte und die zugehörigen Abhängigkeiten
* Zählertausch --> Zähler behält Zählpunktnummer
* Die Zählerdaten werden im viertelstündlichen Rhythmus gespeichert werden. Anschließend kann der Nutzer wählen in welchen periodischen Abständen er die Daten haben will (stündlich, zweistündig,...).
* Für jeden Zähler sollen zusätzliche Infos, wie z.B: Anmerkungen zu Zählern, Standort und Bezeichnung angemerkt werden können.
* Filtern von Daten (nach Standort, Bezeichnung,…) um diese zu editieren oder löschen.

**Erweiterungen/Nicht-Plicht-Ziele**

* Entwicklung einer Mobile-App für Windows-Phone
* Virtuelle Speicherpunkte. Zum Beispiel haben wir drei Stromzähler – deren Summe den Gesamtverbrauch des Gebäudes wiederspiegelt. Oder man hat einen Hauptwasserzähler für 2 Objekte und nur in Objekt1 einen Wasserzähler, dann wäre die Wassermenge für Objekt2 = Hauptwasserzähler-Objekt1.
* Anzeige von Grafen: Auf der Übersichtsseite kann man dann beliebige Speicherpunkte anwählen und Perioden definieren – die zugehörigen Werteverläufe werden dann in einem Diagramm angezeigt. Die Perioden sollen frei wählbar sein (z.B. sollten die Verläufe eines Zählers von zwei oder mehr verschiedenen Jahren übereinander dargestellt werden können).
  + Die Werte sollen in Excel exportiert werden können!

**Geplante Meilensteine**

* Einarbeitung
* Fertiges Userinterface der Website
* Fertige Android-App
* Statistiken und Berichte abgeschlossen
* Funktionsfähiges Projekt
* Fertigstellung Tests
* Fertige Dokumentation

## Zweck dieses Dokuments

Dieses Dokument spezifiziert den Umfang und die Features des Projektes “AEMS – Advanced Energy Monitoring System“ zum gemeinsamen Abgleich zwischen Lieferant(Projektteam) und Kunde (Ing. Herbert Pölzlberger – Energiegenossenschaft Eferding).

# Anforderungen

## Use Case Diagramm

## Bildschirmmasken

Bildschirmmasken sind im Anhang unter 2.11.1 und 2.11.2 zu finden.

## Rahmenanforderungen

Es sollen zwei Oberflächen erstellt werden. Ein Webinterface und eine Android-Applikation. Am Webinterface soll es die Möglichkeit geben sich seine Zähler, Berichte, Statistiken und Warnungen frei konfigurieren zu können. Weiters sollen im Webinterface die Statistiken und Warnungen zur Überschreitung der Energieverbräuche angezeigt werden. Auch der Download der Berichte soll hier möglich sein.

Die Android-Applikation soll die Möglichkeit bieten, sich am Smartphone mit seinem Konto anzumelden und sich anschließend Statistiken anzeigen zu lassen. Auch hier soll es möglich sein sich Berichte herunterzuladen. Warnungen und Benachrichtigungen sollen bei zu hohem Verbrauch direkt am Smartphone angezeigt werden.

## Anforderungen Seitenaufbau-Webinterface

Das Webinterface soll ein dynamisches Layout erhalten und sich somit bestmöglich auf alle Gerätegrößen (PC, Notebook, Tablet, Smartphone,…) anpassen. Das Design soll ansprechend und modern sein. Hier wird als Hilfe die Bootstrap-Library verwendet. Für die einfache Verwendung des Systems, soll der Aufbau des Webinterfaces intuitiv und einfach gehalten werden.

## Anforderungen Aufbau Android-Applikation

Wie auch das Webinterface, soll die Android-App ein ansprechendes, modernes Layout erhalten und der Aufbau und die Bedienbarkeit so einfach wie möglich gestaltet werden. Die App soll für Smartphones und Tablets optimiert sein und im Hoch-, und Querformat verwendet werden können.

## Anforderungen Datenbank

Die Datenbank soll anfangs die Zählerdaten in kürzeren Abständen speichern (alle 30 Minuten). Nach einer gewissen Dauer (1 Jahr, 2 Jahre, ….) sollen die Daten immer weiter komprimiert und zusammengefasst werden. Beispielsweise sollen nach einem Jahr die Daten auf nur mehr stündliche Werte komprimiert werden.

## Anforderungen Statistiken und Berichte

Statistiken sollen direkt im Webinterface und in der App angezeigt werden. Genauso wie die Berichte, soll es möglich sein diese frei zu konfigurieren. Beispiele für die Konfiguration sind die Auswahl der Zähler (sollen alle Zähler ausgewertet werden, oder nur bestimmte Zähler?), Anomalien, wie Abhängigkeit des Stromverbrauchs mit der Außentemperatur, oder wie groß die Zeitabstände der einzelnen Messungen sein sollen.

Für Berichte soll es auch noch möglich sein, Perioden zu wählen, in welchen von selbst Berichte erstellt werden. Die Berichte können selbst gewählte Statistiken enthalten.

## Anforderungen Report-Bot

Um die Stromdaten der AMIS-Zähler zu erhalten, muss ein Report-Bot programmiert werden, welcher sich mit den Userdaten eines Benutzers anmeldet, sich anschließend die Daten jedes einzelnen Stromzählers holt, die Daten anschließend komprimiert, in der Datenbank speichert und sich am Ende wieder vom Nutzerkonto abmeldet.

Der Bot soll sich selbstständig, ab der ersten Anmeldung eines Users, täglich anmelden und die täglichen Werte der Zähler auslesen. Der Bot soll so effizient wie möglich arbeiten, um nicht zu viel Netzwerktraffic zu erzeugen.

## Anforderungen Logindaten

Zur Anmeldung im AEMS sollen die selben Logindaten verwendet werden, wie auf der Energie AG Seite, mit welcher man sich dort die einzelnen Zählerdaten ansehen kann. Die Logindaten sollen verschlüsselt und anschließend in der Datenbank gespeichert werden, damit der Report-Bot jeden Tag die Logindaten zur Verfügung hat um sich anzumelden und die Daten auszulesen.

## Anforderungen Wasser-, Gaszähler

Auf Raspberry-Pies sollen eigene Zähler für Wasser und Gas erstellt werden. Die Daten dieser Zähler sollen auch, genauso wie die Stromdaten, in der Datenbank gespeichert und für Auswertungen, Statistiken, Berichte und Warnungen verwendet werden können.

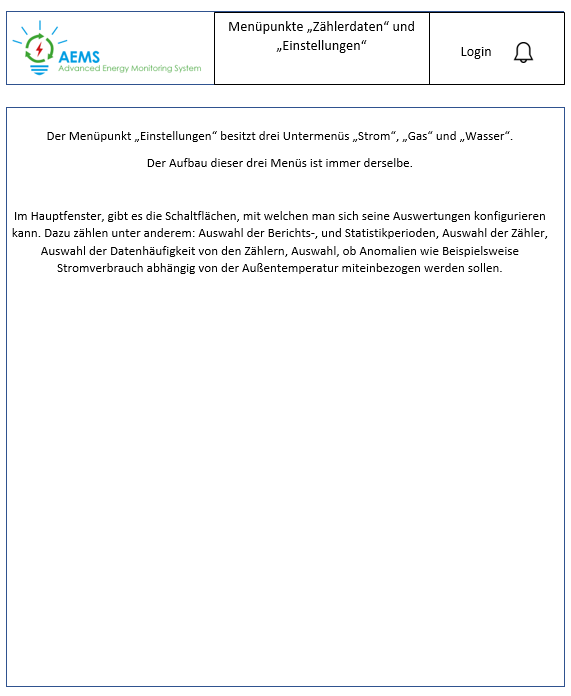
## Anhang

### Bildschirmmasken Webinterface

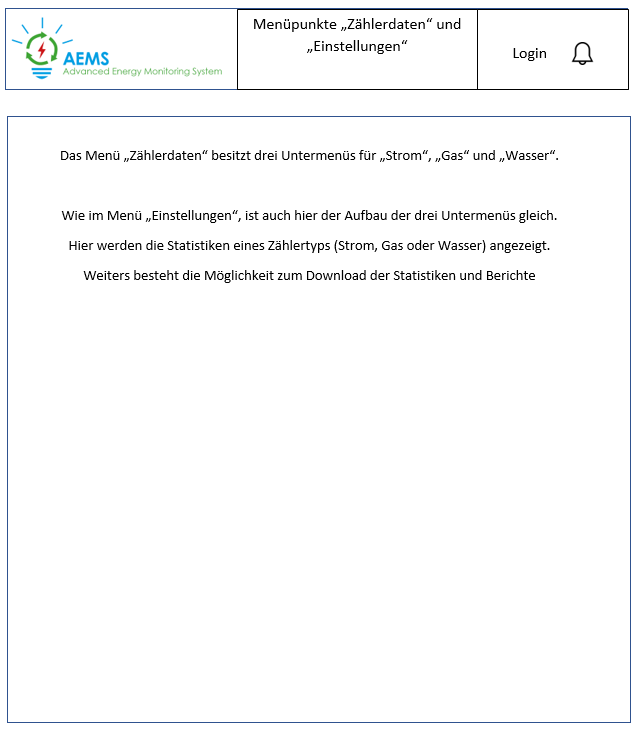
**Startseite**

****

**Menü „Einstellungen“**

****

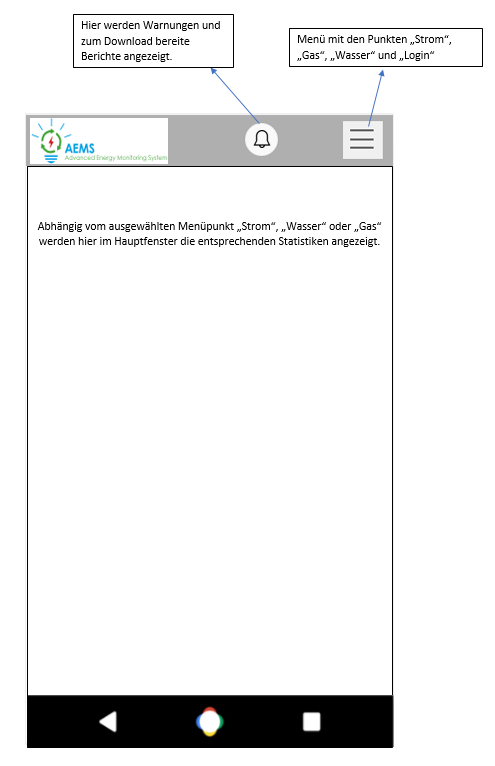
**Menü „Zählerdaten“**

****

### Bildschirmmasken Android-Application

**Aufbau Android-App**

Da der Aufbau der Android-App in jedem Menü identisch ist, gibt es hierzu nur eine Bildschirmmaske.



### Abkürzungen

AEMS – Advanced Energy Monitoring System