

Informatik im Bauwesen I

Aufgabenstellung

zum 1. Teil der 1. Blockübung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Univ.-Prof. Dr.-Ing. Uwe Rüppel
Meiling Shi, M.Sc.
Wintersemester 2018/2019

14. November 2019

Erweiterung einer Modellierungs-Software um ein Planung-Tool für Photovoltaikanlagen Teil 1: Windows Forms-Applikation



Abbildung 1: Vom Dach gerissene PV Anlage (Michael Streib, Zuzenhausen)

Einleitung

Als Ergebnis des stärker werdenden ökologischen Bewusstseins und staatlicher Förderung werden vermehrt Photovoltaik-Anlagen verbaut. Ohne eine korrekte Dimensionierung birgt eine solche Anlage jedoch die Gefahr einer Überlastung, die einen Brand verursachen kann.

Daher wurden Sie von einem Ingenieurbüro damit beauftragt eine Software zu erstellen, die die Dimensionierung einer PV-Anlage übernehmen kann. Dabei sollen verschiedene Bauarten (z.B. Dach-, Insel- oder Fassaden-Anlagen) berücksichtigt werden. In Ihrem Prototypen des Programms soll zunächst nur eine Dimensionierung von transparente PV-Module, die im Fenster als Fensterelement eingebaut wird, vorgenommen werden, eine Erweiterung auf die anderen Bauarten jedoch einfach umsetzbar sein.

Hierfür müssen folgende Aspekte der Räume berücksichtigt werden:

- Maße der nach Süden und Western ausgerichteten Fenstern
- Abmaße und Kennzahlen (Leistung, Spannung, Spannungskoeffizient) der PV-Module
- Anzahl der Stränge
- Kennzahlen der Wechselrichter

Für die Modellierung Ihrer Software können Sie annehmen, dass die Stromstärken der verbauten Module sich addieren. Bei einer Reihenschaltung von Modulen summiert sich ebenfalls deren Spannung, dies kann durch eine Parallelschaltung von Strängen umgangen werden. In dem Fall beträgt die Spannung den Höchstwert aller Spannungen der angeschlossenen Stränge.

Die Software soll die Anzahl der installierbaren Module ermitteln und den passenden Wechselrichter ermitteln. Zudem soll sie die maximale Spannung ermittelt werden. Die hierfür notwendigen Eigenschaften der Module sind die Leerlaufspannung (U_{OC}) und der Spannung bei Nennleistung (U_{MPP}) sowie der Spannungskoeffizient ($TK(U)$). Die anzunehmenden Grenztemperaturen liegen erfahrungsgemäß bei -10°C und 70°C , die Standard-Test-Bedingungen schreiben eine Temperatur von 25°C vor.

$$U_{max} = \max \begin{cases} U_{OC} + TK(U) \cdot (T_{min} - T_{STD}) \\ U_{MPP} + TK(U) \cdot (T_{min} - T_{STD}) \\ U_{MPP} + TK(U) \cdot (T_{max} - T_{STD}) \end{cases} \quad (1)$$

Anhand der maximalen und der für den Wechselrichter zulässigen Spannung soll eine automatische Einteilung der Module in Stränge erfolgen. Für die Bedienung Ihres Programms ist eine graphische Benutzeroberfläche zu erstellen, in der der Nutzer neue Daten eingeben und eine Dimensionierung aktivieren kann.

Aufgabe 1 Grundlage

Geben Sie eine kurze Einleitung in die Grundlagen zur Photovoltaik-Anlage

Aufgabe 2 Modellierung

Erläutern Sie, welche Anforderungen die Software erfüllen und welche Funktionalitäten sie bieten soll. Ermitteln Sie die funktionale und nicht-funktionale Anforderungen. Stellen Sie Ihre Überlegungen in der schriftlichen Ausarbeitung, sowie in einem Anwendungsfalldiagramm (Use-Case Diagram) nach UML Notation dar.

Erstellen Sie ein Objektmodell für die Verwaltung der notwendigen Informationen. Dabei soll das UML – Klassendiagramm als Hilfsmittel verwendet werden, um die Objekte und ihre Beziehungen darzustellen. Setzen Sie Vererbungsstrukturen und Assoziationen zwischen Klassen sinnvoll ein.

Aufgabe 3 Implementierung der vorgegebenen Klassen

Erstellen Sie mit Hilfe von Visual Studio eine Windows-Applikation und implementieren Sie die in Aufgabe 2 spezifizierten Klassen als C#-Klassen.

Aufgabe 4 Implementierung der grafischen Benutzeroberfläche (GUI)

Entwickeln Sie eine grafische Benutzeroberfläche (GUI) unter Verwendung der Windows-Steuerelemente von Visual Studio auf Basis Ihrer Analyse aus Aufgabe 1. Anschließend sollen Interaktionsmöglichkeiten für den Benutzer geschaffen werden, sodass Erzeugen, Ändern und Löschen von Instanzen der Objekte aus Aufgabe 3 zur Laufzeit über die Eingabemasken möglich ist.

Achten Sie bei der Umsetzung darauf, dass falsche Benutzereingaben über try – catch Blöcke abgefangen werden und trennen Sie Darstellungsebene und Anwendungslogik. Achten Sie bei der Gestaltung der Benutzeroberflächen auf einen benutzerfreundlichen Aufbau und verwenden sie die gängigen Elemente entsprechend ihrer gewohnten Einsatzgebiete.

Aufgabe 5 Serialisierung

Um die zur Laufzeit erzeugten Instanzen verlustfrei speichern und laden zu können, sollen die in Aufgabe 2 spezifizierten Objekte erweitert werden, sodass eine Serialisierung und Deserialisierung aller Objekte, Attribute und Beziehungen möglich wird.

Dokumentation

Bitte beachten Sie auch den Quellcode zu dokumentieren!

Formatvorlagen finden Sie unter:

- Offizielles TUD Corporate Design

Allgemeine Hinweise

- Die Betreuungstermine der Blockübung sind unter http://www.iib.tu-darmstadt.de/lehre_iib/mastermodule/iib1/iib1.de.jsp zu finden. Auf Änderungen wird per Moodle Nachrichtenforum hingewiesen.
- Die Abgabe der Übungsergebnisse sowie der Dokumentation muss bis zum **Sonntag, den 09.12.2018, um 23:55 Uhr** über Moodle erfolgen.
- Die Kolloquien zur Übung finden voraussichtlich in der 49. KW statt. Eine Terminabstimmung wird noch übermittelt.
- Die Durchführung der Blockübung und der Kolloquien erfolgt in Gruppen zu je zwei Studierenden.
- Für die Datensicherung ist jede Gruppe selbst verantwortlich.
- **Übungen, die die vorgegebenen Namenskonventionen nicht beachten, werden nicht korrigiert!**
- Es wird nur die Abgabeversion beim Kolloquium bewertet. Sollte im Kolloquium versucht werden, eine aktualisierte Version zu verwenden, wird die Übung mit nicht bestanden gewertet.
- Während des Kolloquiums stehen nur der PC im Raum 222 zur Verfügung.
- Bei ungenügenden Abgaben können zum Bestehen erforderliche Nachbesserungen verlangt werden, in diesem Fall kann eine Übung jedoch nur maximal mit 1 Punkt bewertet werden.

Abzugeben sind:

a) Alle erstellten Dateien

Projektnamenskennung für Visual Studio: **IIB1_UE1_GruppeXX**

b) Dokumentation (doc- oder pdf-Datei)

- Dateinamenskennung: **IIB1_UE1_GruppeXX_Doku.doc** (bzw. .pdf)
- Sämtliche UML Diagramme sind **sowohl** in die Dokumentation einzubetten, als auch als einzelne Dateien abzugeben.
- Bitte die Formatvorlage der TU Darmstadt verwenden und auf dem Titelblatt Gruppennummer, sowie Namen und Matrikelnummern der Teilnehmer vermerken.

c) Packen der Dateien (Zip-Datei)

- Der Projektordner und die Dokumentation beider Übungsteile müssen in eine Zip-Datei gepackt werden.
- Diese Zip-Datei hat die Namenskonvention: **IIB1_UE1_GruppeXX.zip**
- **Nur ein Gruppenpartner** muss die Lösung in Moodle stellen.
- Die Zip-Datei ist bis zu dem oben angegebenen Zeitpunkt in Moodle einzustellen.
- **Bitte überprüfen Sie vor dem Upload, ob alle Dateien vollständig sind und die richtige Version zum Hochladen ausgewählt wurde.**
- **Bitte überprüfen Sie nach dem Upload, ob dieser erfolgreich war!**

Viel Erfolg!