

HUMBOLDT-UNIVERSITÄT ZU BERLIN



LEBENSWISSENSCHAFTLICHE FAKULTÄT

INSTITUT FÜR BIOLOGIE

BACHELORARBEIT

ZUM ERWERB DES AKADEMISCHEN GRADES

BACHELOR OF SCIENCE

"Design, Implementierung und Qualitätssicherung/Test von
Softwarekomponenten für die Auswertung von
INAPRO-Aquaponik-Daten auf Basis eines Datenbankschemas"

"Design, implementation and quality assurance/test of software for the
evaluation of INAPRO-aquaponics-data based on a database schema"

vorgelegt von

Marti Ritter

geb. am 30.08.1993 in Berlin

angefertigt in der Abteilung Biologie und Ökologie der Fische

am Institut für Gewässerökologie und Binnenfischerei

Berlin, im Dezember 2016

Erstgutachter: Prof. Dr. Werner Kloas

Zweitgutachterin: Dr. Klaus Knopf

Zusammenfassung

Die nachfolgende Arbeit befasst sich mit Design, Implementierung und Qualitätssicherung von automatisierten Softwarekomponenten zur explorativen Analyse von INAPRO-Aquaponik-Daten auf Basis eines bereits bestehenden Datenbankschemas und den damit verbundenen Anforderungen und Lösungsansätzen. Im Rahmen dieser Aufgabenstellung wurde ein Konzept zur Visualisierung und Analyse der INAPRO-Untersuchungsdaten entwickelt. Die Funktionsansprüche umfassten eine nutzerfreundliche und direkte visuelle und mathematische Auswertung der grundlegenden Informationen, welche in der Datenbank hinterlegt sind, und die Möglichkeit diese Daten zur weiteren tiefergehenden Auswertung in Form eines Excel-Tabellendokuments zu extrahieren, um sie später mit Excel-kompatiblen Programmen (und Excel selbst) zu bearbeiten. Als Plattform zur Entwicklung des Programms wurde Visual Studio 2015 von Microsoft und die Programmiersprache C# („C sharp“) ausgewählt. Im Laufe der Entwicklung wurden neue Anforderungen an das Programm erkannt und implementiert.

Abstract

The following bachelor thesis deals with design, implementation and quality assurance of automatic software components for the explorative analysis of INAPRO aquaponics data based on a preexisting database schema and the associated requirements and possible approaches. As part of this task a concept for the visualization and analysis of the INAPRO research data was developed. The requirements included a user-friendly and direct visual and mathematical analysis of basic information included in the database and the option to extract this data into an excel-type table for further processing in excel and excel-compatible programs. The development took place using Visual Studio 2015 by Microsoft and the programming language C# (“C sharp”). In the course of development new demands towards the program were recognized and implemented.

Inhaltsverzeichnis

Zusammenfassung.....	III
Abstract	III
Inhaltsverzeichnis.....	IV
1. Einleitung.....	1
2. Aufgabenstellung.....	5
3. Methode	6
4. Lösung/Codeimplementierung.....	8
4.1 Hauptmenü	8
4.2 Server-Einstellungen.....	11
4.3 Erweiterte Einstellungen	13
4.4 Zugriff gesperrt	14
4.5 Vorgang wird ausgeführt (Ladebalken)	14
4.6 Diagramme (und Tabellen)	15
4.7 Diagramm-Einstellungen	18
4.8 Mathematische Auswertung	19
5. Qualitätssicherung.....	22
6. Test/Diskussion	23
7. Ausblick.....	25
8. Anhang.....	26
9. Quellen	34
10. Danksagung	35
Selbstständigkeitserklärung	36

1. Einleitung

Aquaponik ist die Kombination von Aquakultur und Hydroponik (erdeloser Pflanzenanbau). Dabei werden synergetische Effekte bei der Produktion von Fisch und Pflanzen als Weg genutzt, um nachhaltig Nahrung für eine wachsende Weltbevölkerung zu gewinnen. Hierbei werden die im Fischsystem produzierten Substanzen (z.B. stickstoffhaltige Verbindungen) als Dünger für die Pflanzen genutzt.

Untersuchungen mit einem einzelnen geschlossenen Aquaponik-Kreislauf haben gezeigt, dass diese Methode durchaus emissionsarm ist, jedoch eine niedrigere Produktivität hat als zwei getrennte Kreisläufe für Fische und Pflanzen¹². Grund dafür sind beispielsweise die unterschiedlichen Ansprüche von Fischen und Pflanzen an den pH-Wert. Deshalb wurden Wege gesucht, um die Nachhaltigkeit und Produktivität des Systems zu steigern, bei gleichzeitiger Senkung der Umweltbelastung. Im Rahmen des Projekts ASTAF-PRO („Aquaponic System for (nearly) emission free Tomato And Fish PROduction“) wurde eine Alternative zu herkömmlichen Aquaponiksystemen in Form eines Doppelrezirkulations-systems mit einem Einwegventil als Verbindung gefunden². Dabei wird das Wasser aus der Aquakultur durch zusätzliche Düngemittelzugabe dem Nährstoff- und Mineralstoffbedarf der Pflanzen angepasst und anschließend in den Hydroponikbereich geleitet³.

Das Nachfolgeprojekt INAPRO („*INnovative Aquaponics for PROfessional application*“)⁴ ist ein kollaboratives EU-gefördertes Projekt mit 17 internationalen Projektpartnern. Ziel des Projektes ist die weitere Entwicklung und Optimierung von neuen und innovativen Möglichkeiten zum Wasser-, Nahrungsmittel- und Energiemanagement in Aquaponiksystemen. Es beteiligen sich acht Länder am Projekt. Demonstrationsanlagen wurden dazu an vier Standorten (Waren, Abtshagen, Murcia, Shandong)⁵ errichtet. Hierdurch kann der Einfluss unterschiedlicher geografischer und klimatischer Bedingungen auf die Produktionsergebnisse untersucht werden (Fisch, Tomaten). Dabei greift das Projekt auf die Erfahrungen aus ASTAF-PRO zurück und wendet dieselbe Technologie der Doppelrezirkulation im Aquaponiksystem an⁶.

¹ Navigating towards Decoupled Aquaponic Systems: A System Dynamics Design Approach, Goddek et al. 2016

² A new concept for aquaponic systems to improve sustainability, increase productivity, and reduce environmental impacts, Kloas et al. 2015

³ Advanced Aquaponics: Evaluation of intensive tomato production in aquaponics vs. conventional hydroponics, Suhl et al. 2016

⁴ <http://www.inapro-project.eu/>

⁵ <http://www.diplomatisches-magazin.de/business-03-2015-de/A3/?PHPSESSID=ev68br2loeotdnv27tnubqraq4>

⁶ https://www.facebook.com/inaproproject/about/?ref=page_internal

Grundlage des neuen Kopplungsprinzips zwischen Aquakulturteil und Hydroponik ist, dass beide Systeme eigenständige Einheiten (Rezirkulationssysteme) sind und nur in einer Richtung direkt verbunden sind, der Leitung des Fischwassers zu den Pflanzen. Hierdurch wird es möglich, optimale Bedingungen sowohl für den Fisch- als auch für den Pflanzenteil (z.B. hinsichtlich pH-Wert und Nährstoffzusammensetzung für die Pflanzen) einzustellen, wodurch eine höhere Produktivität des Systems ermöglicht wird.

Die Fischzucht besteht aus mehreren Fischbecken, welche nach einem festgelegten Produktionszyklogramm zeitlich versetzt mit Jungfischen besetzt und auch nach definierten Wachstumsphasen wieder abgefischt werden. Die kontrollierten Produktionsbedingungen ermöglichen unter optimalen Bedingungen einen ununterbrochenen Kreislauf von Besatz, Aufzucht und Ernte.

Mechanische Filter im System dienen der Entfernung von Feststoffbestandteilen im Wasser aus dem System. Im Biofilter wird das im Fischkreislauf entstehende Ammonium durch Bakterien (Nitrosomonas und Nitrobacter u.a.) in einem zweistufigen Prozess über Nitrit zu Nitrat umgewandelt, welches als exzellenter Pflanzendünger den Pflanzen über das Fischwasser zur Verfügung gestellt wird.

Im Hydroponikbereich befindet sich zusammen mit den Pflanzrinnen und der Bewässerungstechnik auch ein Anmischbehälter zur zusätzlichen Anreicherung des Wassers aus dem Fischkreislauf mit Düngemitteln und pH-Modifikatoren. Dies ist zur Einstellung optimaler Bedingungen für die Pflanzen unbedingt notwendig, da nicht alle für die Pflanzen relevanten Makro- und Mikronährstoffe aus dem Fischteil in ausreichender Konzentration zur Verfügung stehen⁷. Das für die Pflanzen aufbereitete Wasser zirkuliert mehrfach durch die Pflanzrinnen. Das Evapotranspirationswasser aus dem Pflanzenteil wird über Kondensationsfallen an der Decke des Gewächshauses aufgefangen und wieder der Fischzucht zugeführt um Wasserverluste zu minimieren. Hierdurch lassen sich die täglichen Wasserverluste des Aquakultursystems auf 1-3% des Produktionsvolumens senken⁸.

⁷ Advanced Aquaponics: Evaluation of intensive tomato production in aquaponics vs. conventional hydroponics, Suhl et al. 2016

⁸ A new concept for aquaponic systems to improve sustainability, increase productivity, and reduce environmental impacts, Kloas et al. 2015

Im Verlauf der Erprobung des INAPRO-Systems sind ständige Messungen und Auswertungen von Daten zu Sauerstoffgehalt, pH-Wert, Leitfähigkeit, Temperatur, Sonneneinstrahlung und vieler weiterer Umwelt- und Anlagenbetriebswerte nötig, um technische und technologische Optimierungen im Systembetrieb sowie im Verlauf der Forschungen an Systemaufbau und Steuertechnik vorzunehmen⁹. Zudem sind diese Daten nötig, um eine effektive Steuerung und Regelung des Systems zu erreichen.

Die dabei anfallenden Datenmengen sind ohne technische Hilfsmittel nicht mehr zu bearbeiten. So wurden zum Beispiel bis Ende Mai 2016 60.500 Messwertdatensätze erfasst. Deswegen ist es wichtig ein benutzerfreundliches Programm zu erstellen, welches einen umfassenden und informativen Überblick über die grundlegenden Eigenschaften des erfassten Datensatzes ermöglicht sowie die entsprechenden Werte grafisch darstellen und vorauswerten kann. Die weitere wissenschaftliche Verarbeitung der Informationen sollte über bereits bestehende Programme und Tools erfolgen, beispielsweise Excel oder Statistik-Programme, welche Excel-Tabellen auslesen können.

⁹ Dynamic modeling of the INAPRO aquaponic system, Karimanzira et al. 2016

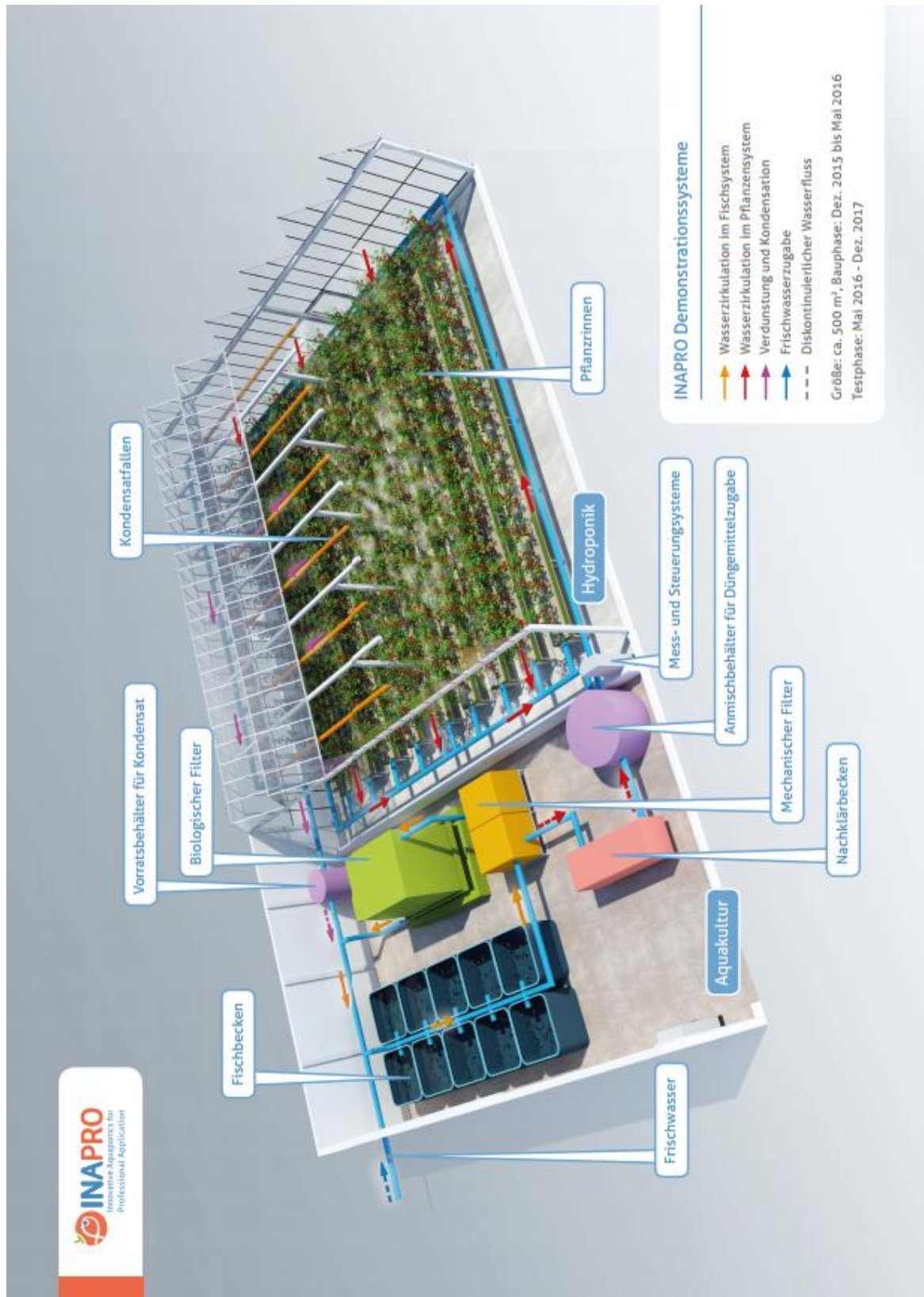


ABBILDUNG 1 BEISPIELABBILDUNG DER AQUAPONIK-ANLAGE IM DRAPS-AUFBAU

2. Aufgabenstellung

Ziel der Bachelorarbeit war die Entwicklung eines einfach zu bedienenden Programms zur explorativen Datenanalyse. Das Programm sollte es ermöglichen, die auf dem MS SQL-Server hinterlegten Daten abzurufen, abzuspeichern, zu modifizieren und lesbar darzustellen (Reporting). Zentrale Bedingung war außerdem die Möglichkeit einer nachfolgenden Auswertung auf Basis von Excel. Dies bedeutet, dass die im Programm enthaltene Reporting-Komponente keine zu hohe Komplexität benötigt und eher auf intuitive und einfache Bedienung und Darstellung orientiert ist. Dennoch sollte die enthaltene Auswertung einige grundlegende statistische Aspekte berücksichtigen, wie zum Beispiel Korrelationen und statistische Verteilungen.

TABELLE 1 – ZEITPLAN DES PROJEKTS

Zeitraum	Bearbeitungsabschnitt ¹⁰
15.08.2016 - 04.09.2016	<ul style="list-style-type: none">- Anforderungsanalyse an das Programm/ Recherche der Thematik und der Programmierwerkzeuge- Festlegung auf grundlegende Eigenschaften, Softwarearchitektur
05.09.2016 – 02.10.2016	<ul style="list-style-type: none">- Erstellung eines ersten Prototyps mit gleichzeitigen Akzeptanztests der implementierten Methoden und Forms
03.10.2016 – 09.10.2016	<ul style="list-style-type: none">- erster Anwendungstest und Aufnahme von neu erkannten Anforderungen an das Programm- grundlegendes Bugfixing
10.10.2016 – 06.11.2016	<ul style="list-style-type: none">- weiterführende Implementierung der neuen Anforderungen und Funktionen
07.11.2016 – 13.11.2016	<ul style="list-style-type: none">- zweiter Anwendungstest- abschließendes Bugfixing mit Regressionstests
14.11.2016 – 17.12.2016	<ul style="list-style-type: none">- Schreiben der Bachelorarbeit- Erstellen des Nutzerhandbuchs

¹⁰ Die Peer Review des Codes fand nicht als separater Bearbeitungsabschnitt statt, sondern lief parallel zur Implementierung

3. Methode

Die verwendete Datenbank ist auf einem lokalen Microsoft-SQL-Server mit der Softwareversion MS SQL 2016 gespeichert. Diese Datenbank spiegelt dabei die Informationen des Servers am Projektstandort und aktualisiert sich regelmäßig über einen Zugriff via VPN. Sogenannte Views zeigen unterschiedlich zusammengestellte Übersichten der verfügbaren Daten. Um die Informationen vom Server in einer individualisierten Zusammenfassung abzurufen, werden SQL-eigene Befehle, „Queries“, benötigt. Diese können auch von externen Programmen an den Server gesendet werden.

Erstellt wurde das Programm in der Entwicklerumgebung von Visual Studio 2015 von der Microsoft Corporation¹¹. Dies ist ein vielseitig anwendbares Entwickler-Kit für viele verschiedene Programmiersprachen, unter anderem auch C#. Ein entscheidender Vorteil ist dabei die Möglichkeit die visuellen Komponenten („Forms“) nach einem einfachen Baukastenprinzip zu optisch aufbereiteten und ansprechenden Menüs zusammenzustellen, ohne dabei einen umfangreichen Formatierungscode schreiben zu müssen¹². Zu den Forms gehören beispielsweise die Programmfenster selbst, aber auch alle Funktionskomponenten.

Aufgrund dieser Möglichkeit erfolgte die Entscheidung das Programm in C# zu schreiben. Die benutzerfreundliche Erstellung von grafischen Nutzeroberflächen ermöglicht eine schnelle und effiziente Anpassung an neue Anforderungen.

Diese Arbeit bezieht sich auf die Versuchsversion der Datenbank am Standort Abtshagen. Alle bei der Entwicklung berücksichtigten Messwertsätze stammen aus diesem Projektstandort. Die verwendete Sensorik ist in der folgenden Tabelle zusammengefasst.

¹¹ Einstieg in C# mit Visual Studio 2015: Ideal für Programmieranfänger geeignet, Thomas Theis

¹² C# 6 mit Visual Studio 2015: Das umfassende Handbuch: Spracheinführung Objektorientierung, Programmiertechniken, Andreas Kühnel

TABELLE 2 – TABELLE DER SENSOREN IM STANDORT ABTSHAGEN

Sensor	Messpunkt	Messbereich	Sensor- typ	Output
Niederschlag	Außen	yes/no	RAM	24 V
Wellenlänge (300-1100 nm)	Außen	0...10 klx	RAM-Light	0-10 V
Windgeschwindigkeit	Außen	0...20 m s ⁻¹	RAM-Wind	0-10 V
Windrichtung	Außen		RAM-Wind	0-10 V
Lufttemperatur	Außen	-30...+130 °C	RAM-Temp	0-10 V
Luftfeuchte	Gewächshaus	0...100%	RAM-Humidity	0-10 V
Lufttemperatur	Gewächshaus	-30...+130 °C	RAM-Temp	0-10 V
Bodentemperatur	Gewächshaus	-30...+130 °C	RAM-Temp	0-10 V
pH-Wert	Vermischungsbehälter	2...12	JUMO	4-20 mA
Leitfähigkeit	Vermischungsbehälter	0...10 mS cm ⁻¹	RAM-EC	0-10 V
Wasserstand	Vermischungsbehälter	0...250 cm	VEGABAR	4-20 mA
pH-Wert	Nährlösungsbehälter 1-5	2...12	JUMO	4-20 mA
Leitfähigkeit	Nährlösungsbehälter / 1-2 (Demonstrationsanlage)	0...10 mS cm ⁻¹	RAM-EC	0-10 V
Wasserstand	Nährlösungsbehälter 1-2 (Demonstrationsanlage)	0...250 cm	VEGABAR	4-20 mA
Wasserstand	Lagerbehälter (Grubenwasser)	0...250 cm	VEGABAR	4-20 mA
Lufttemperatur	Fischzuchthalle	-30...+130 °C	RAM-Temp	0-10 V
Wassertemperatur	Pumpensumpf	0-50 °C	JUMO	Pt100
Sauerstoff	Pumpensumpf	0...20 mg l ⁻¹	Oxygard	4-20 mA
Sauerstoff	Referenzbehälter	0...20 mg l ⁻¹	Oxygard	4-20 mA
pH-Wert	Pumpensumpf	2...12	JUMO	4-20 mA
Leitfähigkeit	Pumpensumpf	0...10 mS cm ⁻¹	RAM-EC	0-10 V
Wasserstand	Pumpensumpf	0...250 cm	VEGABAR	4-20 mA
Energieverbrauch (Strom)	Input	kWh		Imp/t
Energieverbrauch (Heizung)	Input	kCal		Imp/t
Brunnenwassermenge	Verbindungsrohr	0...1000 l h ⁻¹	JUMO	4-20 mA
Wasserverbrauch	Versorgungsanschluss	l		Imp/t
Kondensations-Wassermenge	Lagerbehälter	l		Imp/t

¹³ Siehe Glossar

4. Lösung/Codeimplementierung

4.1 Hauptmenü

Ziel des Designs dieses Forms war die Erstellung eines leicht zugänglichen und verständlichen Mittelstücks des Programms, welches einen schnellen Überblick über die verfügbaren Verwendungszwecke der Datenbankinformationen bietet und dabei die Datenbankstruktur miteinbezieht.

Die verfügbaren Einstellungen sind notwendig für die weiteren Auswertungsoptionen und betreffen die wichtigsten individuellen Konfigurationen.

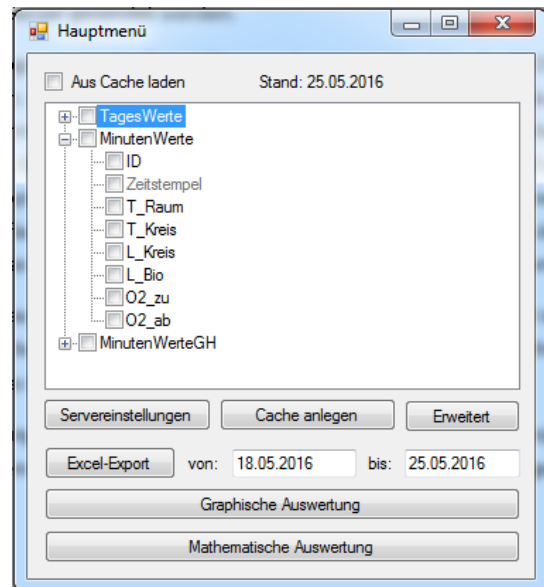


ABBILDUNG 2 HAUPTMENÜ-FORM

Programmablaufplan:

Nach Betätigung der Checkbox „Aus Cache laden“ wird zuerst ein Test ausgeführt, ob das Cache vor dem Zugriff gesperrt ist (isFileLocked (siehe 4.1.5)). Falls ja, wird ein entsprechendes Form angezeigt (4.4 Zugriff gesperrt). Andernfalls werden die Namen der Tabellen und Spalten aus dem Cache ausgelesen und im TreeView-Element angezeigt.

Der Klick auf „Servereinstellungen“ öffnet eine Instanz der Servereinstellungen (siehe 4.2), ebenso wie der Button „Erweitert“ die erweiterten Einstellungen anzeigt (siehe 4.3). Sollte der gewählte Server geändert werden, wird daraufhin mithilfe von „DatabaseLoad“ die Struktur des Servers abgerufen und das TreeView dementsprechend angepasst.

„Cache anlegen“ führt zu einer erzwungenen Neuanlage des Caches des gewählten Servers mithilfe der Methode „CacheUpdate“. In geringerem Umfang geschieht dies auch nach der Betätigung des „Excel-Export“-Buttons. In diesem Fall werden jedoch nur die im TreeView ausgewählten Tabellen und Spalten in das sekundäre TempCache übertragen.

Der Button „Graphische Auswertung“ führt über die Methode DataLoad zu der Erstellung eines „Diagramme“-Forms (siehe 4.6) unter Verwendung der soeben geladenen Daten. Beim Darstellen von mehreren Tabellen wird pro Tabelle ein Form geöffnet. Der Ladevorgang wird von einem „Vorgang wird ausgeführt“-Ladebalken (siehe 4.5) begleitet.

Dieselben Schritte werden beim Klicken des „Mathematische Auswertung“-Button ausgeführt. Dabei werden allerdings alle Tabellen im selben Form verarbeitet und der Ladebalken erhält eine andere Beschriftung.

Wichtigste Methoden:

- 4.1.1 Form1_Load:

Nach Aufruf des Programms wird erstmals Form 1, das "Hauptmenü", geladen. Die vor dem Anzeigen des Fensters ausgeführte Methode heißt Form1_Load und beinhaltet Befehle zum Abruf der Standardeinstellungen aus den Serveroptionen und den erweiterten Funktionen. Dabei werden diese Einstellungen teils in der Programmierung vorgegeben, wie zum Beispiel in den erweiterten Einstellungen und teils während der Laufzeit aus einem bereit gestellten Textdokument ausgelesen.

- 4.1.2 Databaseload:

Der Aufruf der Datenbankabfrage findet nach dem Start des Programms und nach jeder Änderung der Servereinstellungen statt. Dabei wird der Server unter der angegebenen Adresse zuerst nach allen Tabellen und anschließend nach allen Spalten unter den jeweiligen Tabellen gefragt. Weitere Informationen wie z.B. Einheiten, Typ der Tabelle (zeit- oder id-basiert) und Datum des jüngsten Tabelleneintrags werden nebenbei aus den gelesenen Daten oder einer parallel gestellten Metadatentabelle gewonnen.

- 4.1.3 TreeView-Struktur:

Das TreeView-Steuerelement wird aufgrund von gegebenen Bedienungsregeln von einer separaten Klasse bereitgestellt, welche das von Microsoft bereitgestellte Steuerelement so modifiziert, dass es die Möglichkeit eines Doppelklicks unterbindet, um Konflikte mit den verfügbaren Events im TreeView zu vermeiden.

Die Steuerung des Elements wird über mehrere Methoden geregelt, die die möglichen Aktionen des Nutzers verwalten. Dabei führt das Anklicken der Checkbox eines Parent-Nodes zur Aktivierung aller darunterliegenden Child-Nodes und die Abwahl eines Child-Nodes zur automatischen Deaktivierung des Parent-Nodes. Dies soll eine einfachere Auswertung der Nutzerauswahl ermöglichen und verständliches Feedback über die derzeitige Konfiguration liefern.

Unter Berücksichtigung der vom Nutzer gewählten Einstellungen in „3. Erweiterte Einstellungen“ werden auch bestimmte Grundregeln in der Nutzung des TreeView-Elementes festgelegt. So werden zum Beispiel die Verwendung und Markierung von möglicherweise inkompatiblen Spalten (falsche Einheit, keine Timestamps oder Lücken) vom Nutzer gesteuert.

- 4.1.4 Dataload:

Diese Methode ruft mit den eingegebenen Login-Informationen die Daten auf dem spezifizierten Server ab und lädt diese in ein temporäres Dictionary-Konstrukt, welches unter jedem Tabellennamen die ausgewählten Spaltennamen und dann unter deren Keys letztendlich die jeweiligen Listen mit den dazugehörigen Daten hinterlegt.

Diese Methode wird erst bei Betätigung der zu den weiteren Auswertungs-Forms führenden Buttons aufgerufen und überschreibt bei jedem Aufruf eventuell bestehende vorherige Strukturen.

- 4.1.5 IsFileLocked:

Bei Zugriff auf ein Dokument kann es vorkommen, dass dieses bereits von derselben oder anderen Anwendungen verwendet wird und dadurch schreibgeschützt ist. Um eventuelle Fehler während der Laufzeit zu vermeiden wird vor jedem Aufruf einer Datei diese Methode aufgerufen, welche einen inhaltslosen Zugriff auf das gewünschte Dokument versucht. Sollte während dieses Versuchs eine fest definierte Ausnahme auftreten, deutet dies darauf hin, dass die Datei gerade verwendet wird und nicht genutzt werden kann. In diesem Fall wird das Fenster „5. Zugriff gesperrt.“ aufgerufen, welches das weitere Verfahren vermittelt.

- 4.1.6 Truncator:

Da das Excel-Nutzerinterface die Verwendung von Namen mit einer Länge von mehr als 31 Zeichen für Worksheets nicht erlaubt, muss ein einheitlicher Ersatzname während der Laufzeit gefunden werden. Um diese Namen so verständlich wie möglich zu halten, fügt diese Methode Datenbank- und Tabellennamen zusammen und kürzt gleichmäßig in beiden Abschnitten, bis die mögliche Namenslänge erreicht ist.

Sollten im Namen weitere Unterstriche vorkommen, werden diese wie zusätzliche Abschnitte behandelt und ebenso gleichmäßig gekürzt.

- 4.1.7 CacheUpdate:

Diese Methode leitet ein Update des im Installationsverzeichnis hinterlegten Excel-Caches ein, indem es die jüngsten Zeitstempel des Caches und der Datenbank vergleicht und nötigenfalls fehlende Datensätze nachlädt. Dies sollte eine vom Server unabhängige Verarbeitung der Informationen ermöglichen.

4.2 Server-Einstellungen

Dieses Form ermöglicht eine intuitive Konfiguration der Serveradresse und der Login-Informationen. Dabei sollten potentiell mehrere Server leicht zugänglich sein, weshalb eine vom Nutzer modifizierbare .ini-Textdatei verwendet wird. In dieser Datei sind die bekannten Server mit ihren Zugangsdaten und ihrer jeweiligen Datenbank hinterlegt. Der SQL-Zugriff muss immer auf eine definierte Datenbank erfolgen, deswegen muss bei Zugriff auf mehrere Datenbanken auf denselben Server ein jeweils eigener Eintrag in der .ini-Datei existieren. Da die Servernamen frei vergeben werden können, kann dies jedoch in der Zeile für den Namen vermerkt werden.

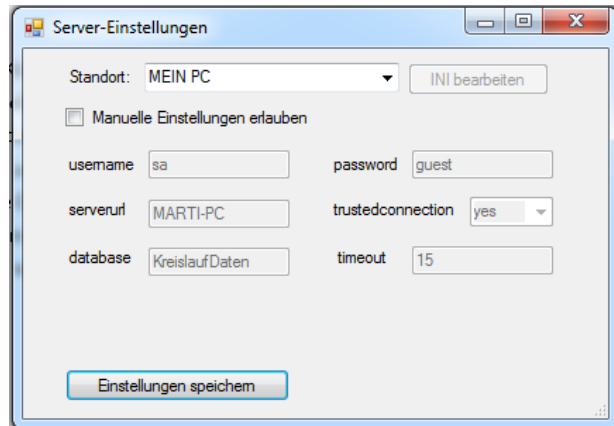


ABBILDUNG 3 SERVER-EINSTELLUNGEN-FORM

Programmablaufplan:

Solange die Checkbox „Manuelle Einstellungen erlauben“ nicht gefüllt ist, sind alle weiteren Optionen, von der Dropdown-Liste einmal abgesehen, gesperrt. Bei einer Änderung des gewählten Servers in der Liste werden alle weiteren Angaben in den entsprechenden Textboxen mithilfe von `INIReadSettings` (siehe 4.2.4) aus der .ini-Konfigurationsdatei ausgelesen und eingetragen. Vorausgesetzt die Checkbox wurde gefüllt, können alle entsprechenden Zeilen auch von Hand eingetragen und modifiziert werden. Außerdem kann ab diesem Punkt auch der Button „INI bearbeiten“ geklickt werden, was zu einem temporären Blockieren des Forms mit Freigabe und Öffnen der entsprechenden .ini-Textdatei führt. Wenn die Bearbeitung der Datei abgeschlossen ist, weil das Textbearbeitungsprogramm geschlossen wurde, wird dieses Form wieder freigegeben und der Inhalt der Datei wird durch `INIReadLocations` (siehe 4.2.3) und anschließend `INIReadSettings` (siehe 4.2.4) erneut eingelesen. Wenn alle Eingaben abgeschlossen sind, führt der Button „Einstellungen speichern“ zur Übertragung der neuen Daten an das Hauptmenü-Form und dem Schließen dieses Fensters.

Wichtigste Methoden:

- 4.2.1 ServerSettingsForm:

Der Aufruf dieses Forms vom Hauptmenü aus führt direkt zur Erstellung eines neuen ServerSettingsForm-Objekts. Die Standardeinstellungen und bekannten Serverdaten werden aus der INI ausgelesen und in der Combobox gelistet. Dabei wird der erstgenannte Server als Standard gewählt. Dadurch, dass eine Load-Methode vermieden wird, muss das Hauptmenü die Servereinstellungen selbst nicht öffnen und kann die benötigten Informationen auch vorher aus dem Objekt auslesen.

- 4.2.2 Steuerungsbefehle:

Die Methode ComboBox2_SelectedIndexChanged passt, wenn ein neuer Server in der Combobox gewählt wurde, die dazugehörigen Einstellungen an und trägt sie in den entsprechenden Textfeldern ein. Um als Nutzer selbstständig Änderungen an den Textfeldern vorzunehmen, muss das Häkchen in der entsprechenden Checkbox gesetzt werden.

- 4.2.3 INIReadLocations:

INIReadLocations wird bei der Erzeugung des Form-Objektes ausgeführt, ebenso wie bei jeder weiteren Änderung des gewählten Servers. Die Locations sind dabei die in eckigen Klammern gehaltenen Namen der bekannten Server und nur sie werden ausgelesen.

- 4.2.4 INIReadSettings:

Diese Methode ist das Gegenstück zu ReadLocations und liest die Informationen des gewählten Servers unter seiner Kopfzeile aus. Diese Trennung soll das Laden unnötiger Informationen vermeiden.

4.3 Erweiterte Einstellungen

Die erweiterten Einstellungen beinhalten zusätzliche Optionen, welche thematisch nicht zu den Servereinstellungen passten. Dabei handelt es sich um unterschiedliche Möglichkeiten die ausgelesenen Daten zu verarbeiten, zum Beispiel unter Berücksichtigung einer fest definierten Zeitstempel-Spalte. Dies spart Zeit und Rechenleistung bei der Suche nach potentiellen verwendbaren Tabellenspalten

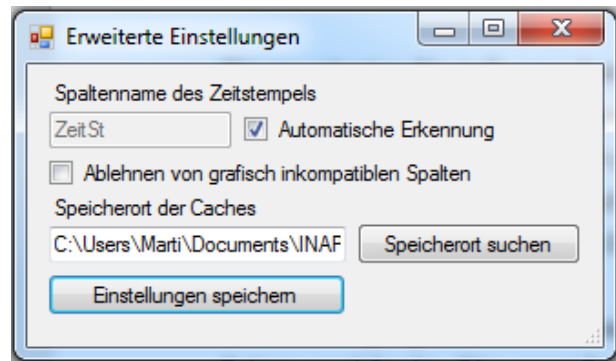


ABBILDUNG 4 ERWEITERTE EINSTELLUNGEN-FORM

und ermöglicht es auch Datensätze mit mehreren Zeitspalten zu verarbeiten. Die Bestimmung des Speicherpfades des Cache findet hier statt.

Programmablaufplan:

Solange die Checkbox „Automatische Erkennung“ gefüllt ist, ist die danebenliegende Textbox gesperrt und die Erkennung der Spalte mit den Zeitstempeln wird automatisch durchgeführt. Dabei wird nach der am weitesten links liegenden Spalte gesucht, welche durchgehend mit Werten des Typs DateTime gefüllt ist. Sollte die Checkbox nicht gefüllt sein, wird der Inhalt der Textbox als Name der Zeitstempel-Spalte verwendet und bei der Auswertung wird vorausgesetzt, dass eine Spalte dieses Namens existiert. Sollte dies nicht der Fall sein, wird die Tabelle als nicht lesbar gewertet.

Die Checkbox „Ablehnen von grafisch inkompatiblen Spalten“ regelt die Auswertung von Spalten, welche Daten beinhalten, die nicht als Diagramm darstellbar sind. Dazu gehören beispielsweise true/false-Angaben oder lückenhafte Spalten. Diese Einstellung hat lediglich Einfluss auf die Diagramme (siehe 4.6), wo diese Spalten als inkompatible Werte in einer aufklappbaren Tabelle untergebracht werden. Die mathematische Auswertung (siehe 4.8) ignoriert Spalten dieses Typs automatisch und somit hat diese Einstellung keine weiteren Auswirkungen.

Beim Klicken auf den Button „Speicherort suchen“ wird das aus dem Windows-Betriebssystem bekannte Explorer-Fenster geöffnet, in welchem der Speicherort des Cache festgelegt werden kann. Beim Schließen dieses Fensters wird der zuletzt geöffnete Pfad in die Textbox daneben übertragen und bei den Zugriffs- und Speichervorgängen verwendet.

4.4 Zugriff gesperrt

Bei Zugriff auf eine bereits verwendete und deswegen schreibgeschützte Datei wird dieses Fenster angezeigt. Der dem Nutzer gezeigte Text wird dabei von der erzeugenden Methode festgelegt und ist deshalb der Situation angepasst, das heißt Name und Pfad der Datei werden während der Laufzeit eingetragen. Diese Möglichkeit kann auch genutzt werden, um das Form anderen Situationen anzupassen, wie zum Beispiel einer beliebigen Ja-Nein-Abfrage.

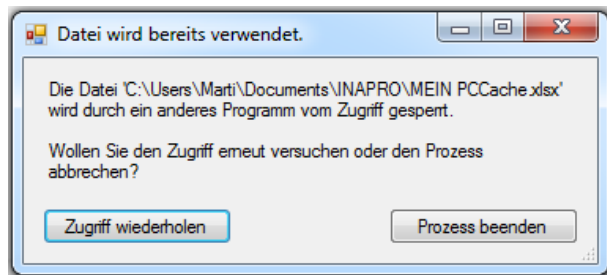


ABBILDUNG 5 ZUGRIFF GESPERRT-FORM

Programmablaufplan:

Bei einem Klick auf den linken Button wird ein „true“-Wert an das erzeugende Form übertragen, der rechte führt zu einem „false“-Return. Die weitere Auswertung dieser Aussagen erfolgt in der ursprünglichen Methode und ist kontextabhängig.

4.5 Vorgang wird ausgeführt (Ladebalken)

Um für den Nutzer den Arbeitsvorgang des Programms (vor allem auf langsameren PCs) zu visualisieren, wird bei jedem Lade- oder Erstellungsvorgang ein angepasster Ladebalken angezeigt. Dabei ist die Anzahl der Gesamtschritte von der jeweiligen Aktion abhängig. Bei einem Excel-Zugriff wird zum Beispiel die Gesamtzahl der Felder betrachtet und bei dem Ladevorgang der Diagramm-Forms die Anzahl der bereits geladenen Forms.

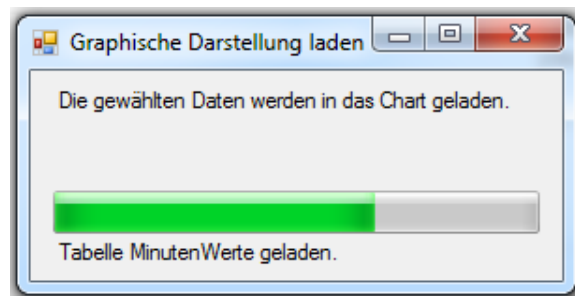


ABBILDUNG 6 LADEBALKEN-FORM

Programmablaufplan:

Dieses Form beinhaltet keine Nutzersteuerung und ist vollständig vom Erzeugervorgang abhängig: Einleitungstext, Maximallänge des Ladebalkens sowie der derzeitige Stand werden alle vom erzeugenden Form übertragen (im Falle des Ladebalkens selbst sogar bei jedem Schritt aufs Neue).

4.6 Diagramme (und Tabellen)

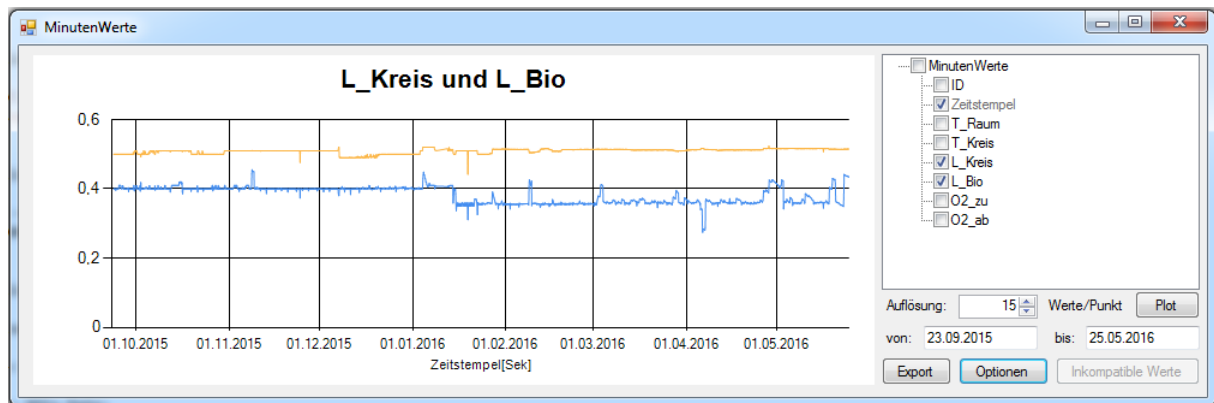


ABBILDUNG 7 DIAGRAMM- FORM

Teil der Anforderungen an das Programm war die Möglichkeit publikationsfähige Diagramme und Reports zu schaffen. Der visuelle Teil dieser Aufgabe wird durch dieses Form erfüllt, welches die Zusammenstellung und Modifikation von Linien- und Balkendiagrammen aus gegebenen Daten ermöglicht. Es können Anfang und Ende des Diagrammes, die Auflösung der gezeichneten Punkte und Minima und Maxima des gezeigten Graphen festgelegt werden. Pro Fenster wird eine Tabelle abgebildet, um erste Vergleiche der Graphen zu ermöglichen.

Programmablaufplan:

Das Betätigen einer Checkbox im TreeView-Element führt zu einem Update der Chart-Oberfläche (großes Fenster links). Dabei wird die gesamte Spalte unter dem ausgewählten Namen von der FitSeries-Methode (siehe 4.6.3) mit der erkannten Zeitstempel-Spalte zu Chart-Punkten zusammengefasst. In der Grundeinstellung ist die Auflösung 1 Messwertsatz pro Grafikpunkt.

Sollte die Auflösung mithilfe des Reglers unterhalb des TreeView-Elements geändert werden, kann mithilfe des „Plot“-Buttons die Spalte neu berechnet werden. Es wird wieder die FitSeries-Methode verwendet, allerdings diesmal mit einem anderen Auflösungs-Argument.

Der „Export“-Button erstellt eine Kopie des gerade angezeigten Charts im EmfDual-Format und legt Sie im lokalen Ordner ab.

Ein Klick auf „Optionen“ führt zur Erstellung eines Diagramm-Einstellungen-Forms (siehe 4.7) und zeigt dieses an.

Wichtigste Methoden:

- 4.6.1 Form4_Load:

Während des Ladevorgangs des Diagramms werden alle lesbaren und vorher im Hauptmenü gewählten Spalten der zugehörigen Tabelle gegen die verfügbare x-Achse aufgetragen. Je nach Tabelle ist das vorzugsweise eine Zeitachse oder notfalls eine ID-Liste, welche zu einem Balkendiagramm führt, um, zum Beispiel die Futtermenge pro Becken zu vergleichen.

- 4.6.2 FillDataGrid:

Sollte eine Spalte nicht graphisch auswertbar sein, sei es, weil sie Lücken enthält oder weil sie nur true/false-Werte beinhaltet, dann wird ihr gesamter Inhalt mit ihrer entsprechenden x-Achse in Form einer Tabelle unter dem Button „inkompatible Werte“ abgelegt.

- 4.6.3 FitSeries:

Die FitSeries-Methode verbindet die y-Werte mit den x-Werten. Dabei werden Punktserien geschaffen, welche unter einer systembedingt maximalen Länge liegen (5000 Punkte) und miteinander verbunden werden, um die vollständige Liste zu erhalten. Bei der Auswahl der Spalte in der Treeview-Struktur wird die entsprechende Serie in das Diagramm geladen.

- 4.6.4 Chart1_MouseMove:

Diese Methode erlaubt die Darstellung von Tooltips. In diesem Fall ist es ein Tooltip mit den Werten des unter der Maus befindlichen Punktes.

- 4.6.5 Treeview-Struktur:

Die in diesem Form verwendete Treeview ist eine simplere Form des Hauptmenü-Treeviews, in dem die Färbung von unpassenden Werten entfällt und die Steuerung keine Unterscheidung der Einträge zueinander vornimmt. Die Auswahl einer Spalte führt zu ihrer Darstellung im Diagramm.

- 4.6.6 Hilfsfunktionen:

Da die Anfangs- und Enddaten der geladenen Datenserie zwar anfangs in den entsprechenden Textboxen dargestellt werden, aber überschrieben werden können, sorgt ein Doppelklick auf das jeweilige Label für ein Reset der Einstellung auf den Standardwert.

- 4.6.7 ListBetween:

Dies ist eine zurzeit ungenutzte Funktion, welche ursprünglich als Weg geplant war, um alle Punkte zwischen den gewählten Daten darzustellen. Diese Funktion wurde erreicht, indem das Sichtfenster des Diagrammes auf den gewünschten Abschnitt verkleinert wird. Die Funktion ist hilfreich bei der Verwendung von zu großen Spalten, um die geladene Datenmenge zu reduzieren.

- 4.6.8 CutSeries:

CutSeries sorgt für das Abschneiden von zu hohen oder zu niedrigen Punkten im Diagramm. Dadurch wird dem Nutzer beispielsweise ermöglicht etwaige Messausfälle in Form von Nulleinträgen herauszufiltern, indem er die untere Abschnittsgrenze auf 0 setzt und somit diese Einträge ignoriert.

4.7 Diagramm-Einstellungen

Dieses Form ermöglicht die Modifikation des angezeigten Diagramms. Dazu gehören Größe, Form und Text von Achsenbeschriftungen und Titel und die Möglichkeit, Punkte ober- oder unterhalb eines Grenzwertes abzuschneiden und als Outlier darzustellen. Außerdem kann die automatisierte Legende ein- oder ausgeschaltet werden.

The screenshot shows a Windows-style dialog box titled 'Diagramm-Einstellungen'. It has a table for text settings and a section for chart options.

	Text	Schriftgröße	Stil
<input checked="" type="checkbox"/> Titel	L_Kreis und L_Bio	16	Fett
<input checked="" type="checkbox"/> X-Achsentitel	Zeitstempel[Sek]	8	Normal
<input type="checkbox"/> Y-Achsentitel	Multiple Values	12	Normal

Below the table, there are two rows of settings:

- ☒ Minimum: 0.2
- ☐ Maximum: 0
- ☐ Outlier als Punkte darstellen
- ☐ Outlier in Legende vermerken
- ☐ Legende anzeigen

At the bottom, there are two buttons: 'Speichern' and 'Zurücksetzen'.

ABBILDUNG 8 DIAGRAMM-EINSTELLUNGEN-FORM

Programmablaufplan:

Jede der Checkboxes steht für eine true/false-Abfrage, welche nach dem Schließen dieses Forms mit dem „Speichern“-Button an das erzeugende Diagramm-Form übertragen und ausgewertet wird. Der Inhalt der oben gelegenen Textboxen wird bei Bedarf in die entsprechenden Beschriftungen im Chart übernommen und die rechts danebengelegenen Angaben zu Schriftgröße und Stil dienen als Argumente in der Erstellung eines Title-Objektes im Diagramm-Form.

Die unteren Einstellungen dienen entweder der Anpassung des Graphen (Minimum, Maximum, Outlier als Punkte) oder der Legende (Outlier in Legende, Legende anzeigen). Sollten die Checkboxes bei Minimum und Maximum gefüllt sein, werden die entsprechenden Werte an das erzeugende Diagramm-Form übertragen und dort in der CutSeries-Methode verwendet, um außerhalb des gewählten Bereiches liegende Punkte aus dem Graphen zu entfernen. Die Checkbox „Outlier als Punkte darstellen“ ermöglicht es diese Punkte trotzdem einlesen zu lassen und als zusätzliche Serie im Chart darzustellen. Sofern es gewünscht ist, kann diese Serie auch in der Legende vermerkt werden („Outlier in Legende vermerken“) oder die Legende kann vollständig entfernt werden („Legende anzeigen“ ist nicht gefüllt). Im letzteren Fall führt der „false“-Wert der entsprechenden Anfrage dazu, dass das Legend-Objekt nicht im Chart erstellt wird und, falls bereits vorhanden, entfernt wird.

4.8 Mathematische Auswertung

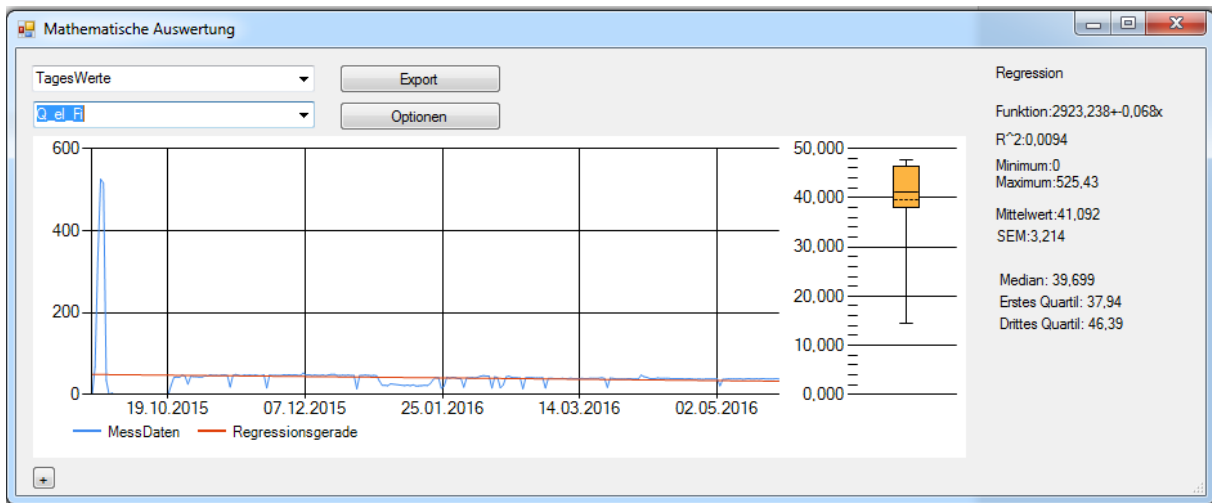


ABBILDUNG 10 MATHEMATISCHES AUSWERTUNGS-FORM (KLEIN)

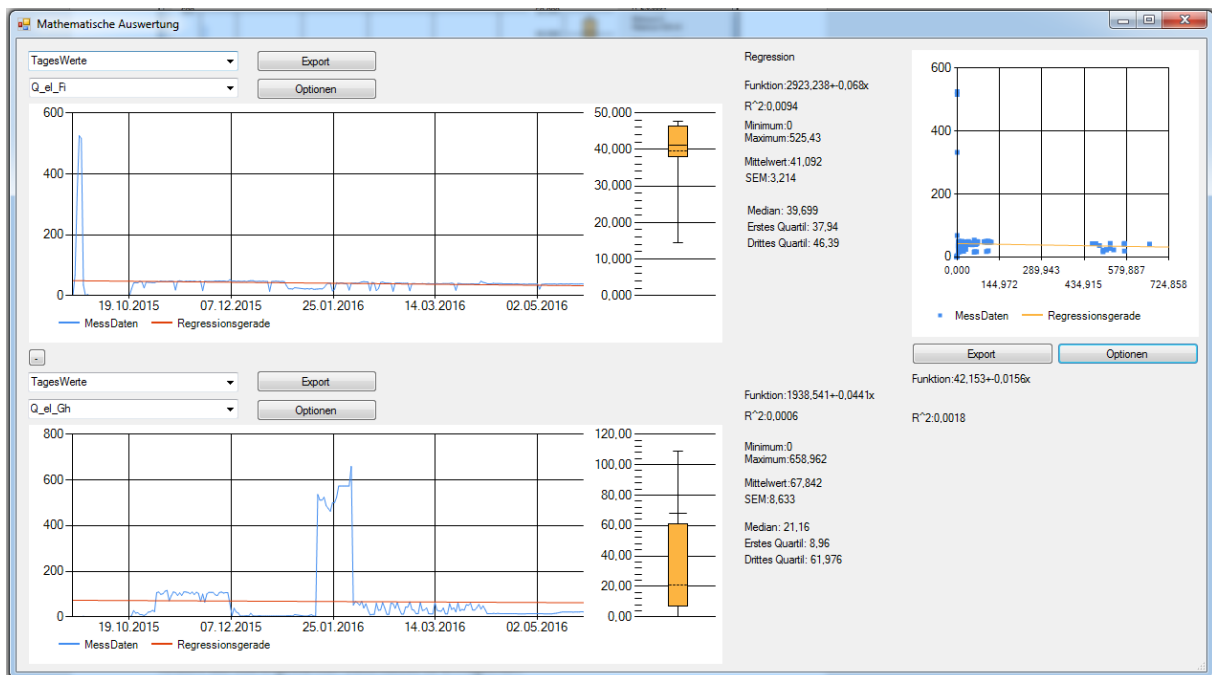


ABBILDUNG 9 MATHEMATISCHES AUSWERTUNGS-FORM (AUFGEKLAFFT)

In der mathematischen Auswertung werden grundlegende statistische Eigenschaften der gemessenen Datensätze berechnet und für den Nutzer in einem kompakten Überblick dargestellt. Die berechneten Werte beinhalten beispielsweise Minimum und Maximum, die Standardabweichung, den Median und eine geschätzte lineare Abhängigkeit der gemessenen Kurve. Um diese Werte zu bestimmen, wurden teilweise in Visual Studio vorhandene Methoden zur Rundung oder Wurzelberechnung genutzt. Größtenteils wurden jedoch die Standardformeln für die dargestellten Werte verwendet.

Programmablaufplan:

In der Grundversion des Forms werden zwei Drittel durch den eingeschränkten Umriss des Sichtfensters verborgen. Solange in diesem Zustand Veränderungen in der Auswahl der Tabelle und Spalte in den beiden oberhalb des Chart-Fensters gelegenen Dropdown-Listen stattfinden, wird lediglich das obere Chart-Fenster neu gezeichnet.

Mithilfe des „Export“-Buttons kann, wie auch schon im Diagramme-Form (siehe 4.6), eine Kopie des gezeigten Charts im EmfDual-Format im lokalen Ordner abgelegt werden.

Der darunter gelegene „Optionen“-Button öffnet dasselbe Diagrammeinstellungs-Fenster (siehe 4.7), wie schon im Diagramme-Form gezeigt. Nach Schließen der Einstellungen mit „Speichern“ werden die nötigen Einstellungen in das unter dem Button gelegene Chart übernommen.

Unter dem linken oberen Chart befindet sich ein „+“-Button, der ein Aufklappen des Fensters ermöglicht und so die beiden verbliebenen Charts sichtbar macht. Solange in diesem Zustand zwei verschiedene Spalten in den beiden linken Dropdown-Listen ausgewählt sind, werden diese gegeneinander aufgetragen und im rechten Chart als Serie von Punkten dargestellt.

Wichtigste Methoden:

- 4.8.1 Form6_Load:

Während des Ladevorganges dieses Forms werden viele der Prozesse aus der graphischen Auswertung übernommen und vereinfacht auf die drei verfügbaren Charts übertragen. Im Anfangszustand des Forms sind zwei der drei Chart-Oberflächen verborgen, um eine übersichtliche Ansicht des zuerst gewählten Datensatzes zu ermöglichen.

- 4.8.2 Steuerbefehle:

Sollte ein Vergleich zweier Spalten gewünscht sein, kann unter Nutzung des “+” Buttons die Oberfläche aufgeklappt werden, um die beiden anderen Charts zugänglich zu machen. Da die Diagramme jeweils nur eine Serie von Messpunkten enthalten sollen und dürfen, erfolgt die Auswahl der jeweiligen Datensätze über Comboboxen. Dabei sind diese so verknüpft, dass die Auswahl einer Option in einer bereits im anderen Chart geöffneten Spalte unterbunden wird, indem deren Name bei jedem Aufruf aus der Liste der verfügbaren Optionen entfernt wird. Sollte eine neue Tabelle mit anderen Spalten in der oberen Combobox gewählt werden, wird die Liste der unteren Combobox mit der vollständigen Menge an passenden Optionen aus einem Schablonen-Objekt befüllt.

- 4.8.3 Statistische Verarbeitung:

Die drei Methoden DrawTop, DrawBottom und DrawRight werden in dieser Reihenfolge aufgerufen, um zuerst den oberen und, falls nötig, den unteren und rechten Abschnitt des Forms zu berechnen und zu zeichnen. Dabei basieren die Daten und die Gerade des rechten Charts auf der Kombination der jeweils zueinander gehörigen Messpunkte in den beiden anderen Charts. Die Erzeugung der Punkteserien erfolgt mit einer weitestgehend aus der graphischen Auswertung übernommenen FitSeries-Methode.

- 4.8.4 Quantil:

Quantil ist eine mathematische Hilfsmethode, welche von einer gegebenen Menge von Werten und einem Quantilwert q den Wert des q -ten Quantils berechnet. Diese bilden in den Boxplots dann die entsprechenden Ordinalwerte von Box und Whiskers.

5. Qualitätssicherung/ Tests

Folgende Maßnahmen wurden durchgeführt, um die erforderliche Qualität der Software zu sichern:

- Codeinspektion: Dies beinhaltet das Korrekturlesen des geschriebenen Codes und die Vermeidung von Fehlern noch vor der ersten Ausführung. Dabei unterstützt der Visual Studio Code Editor in der Entwicklungsumgebung bereits eine grundlegende Syntaxprüfung durch Syntaxhervorhebung, IntelliSense- und Autocomplete-Funktionen.
- Peer Review: Im Rahmen der Qualitätssicherung wurde während der Implementierung aller Funktionen eine Peer Review in Form einer stichprobenartigen Codeinspektion durch die Mitarbeiter des IGB durchgeführt.
- Funktionale Tests/ Regressionstests in der Entwicklungsumgebung: Durch das Ausführen des Codes in Visual Studio können möglicherweise auftretenden Fehlerwarnungen behoben werden. Dabei fand auch die Überprüfung des korrekten Programmoutputs statt.
- Funktionale Tests in der Laufzeitumgebung des Anwenders (IGB): Mehrere Anwendungstests der derzeitigen Version des Programms wurden in festgelegten Zeiträumen durch den Nutzer durchgeführt mit gleichzeitigem Beheben von auftretenden Programmfehlern.
- Akzeptanztests: Der Anwender wurde frühzeitig in Entscheidungen zu Design und Funktionalität der Anwenderoberfläche einbezogen, die dann vor Ort zur Zufriedenheit abgenommen wurden.

In der abschließenden Erprobungsphase durch den Anwender wurden alle Funktionen getestet und funktionierten fehlerfrei.

6. Diskussion

Die erste Operation des Programms ist der Aufruf der Datenbankstruktur nach dem Öffnen des Programms und unmittelbar vor der Anzeige des Hauptmenüs. Dieser Zugriff dauert im Schnitt 147 Millisekunden und beinhaltet eine Abfrage der verfügbaren Tabellennamen und daraufhin eine Abfrage aller Spaltennamen in jeder Tabelle, welche der Reihe nach abgearbeitet werden. Während dieses Zugriffs kam es zu keinen Fehlern und das Hauptmenü wurde korrekt mit der Datenbankstruktur beschrieben.

Eine der möglichen nachfolgenden Operationen ist die Erstellung eines Cache des Datenbankinhaltes. Der gesamte Vorgang benötigt durchschnittlich 2449 Millisekunden um 793.496 Werte aus drei Tabellen einzulesen, beziehungsweise zu erstellen und anschließend in ein bereits bestehendes Excel-Dokument zu übertragen. Derselbe Vorgang dauerte ohne ein bereits bestehendes Zieldokument 2738 Millisekunden. Bei der Übertragung und einer Kontrolle in der vollständigen Zieltabelle wurden keine Fehler festgestellt und der eingelesene Cache führte zu denselben Ergebnissen wie die Verwendung der ursprünglichen serverbasierten Datenbank.

Der nächste Schritt ist die Darstellung der eingelesenen Daten in einem Diagramm. Der Einlesevorgang dauerte für alle drei bereits verwendeten Tabellen 3673 Millisekunden und führt zu drei voneinander getrennten Diagrammfenstern. Die eingelesenen Daten entsprechen dabei 45 Spalten, wovon drei Zeitstempel, drei IDs und eine inkompatible true/false-Angaben beinhalten. Dies führt zu 38 eigenständigen Graphen und eine Tabelle mit inkompatiblen Daten. Eine Darstellung einer Stichprobe der eingelesenen Daten führte bei keiner der drei Tabellen zu Fehlern.

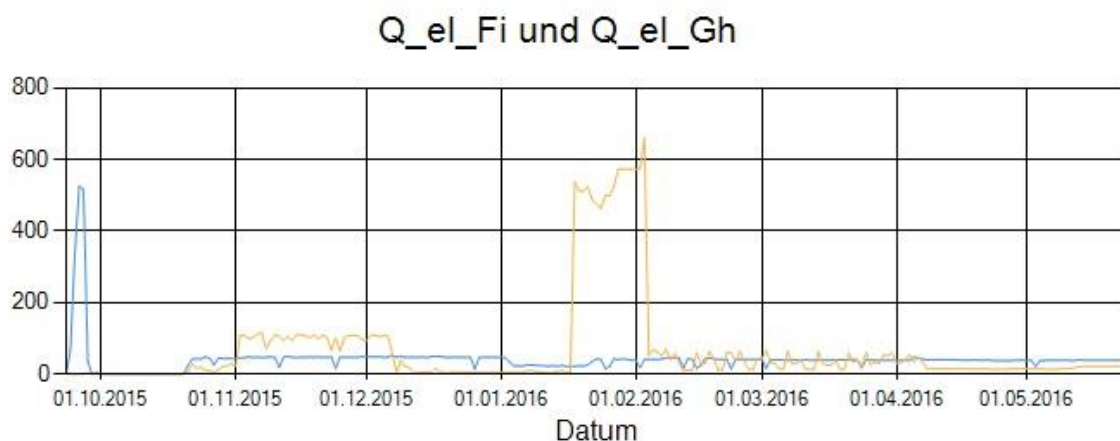


ABBILDUNG 11 BEISPIELDIAGRAMM DER TAGESWERTE, 241 PUNKTE PRO GRAPH

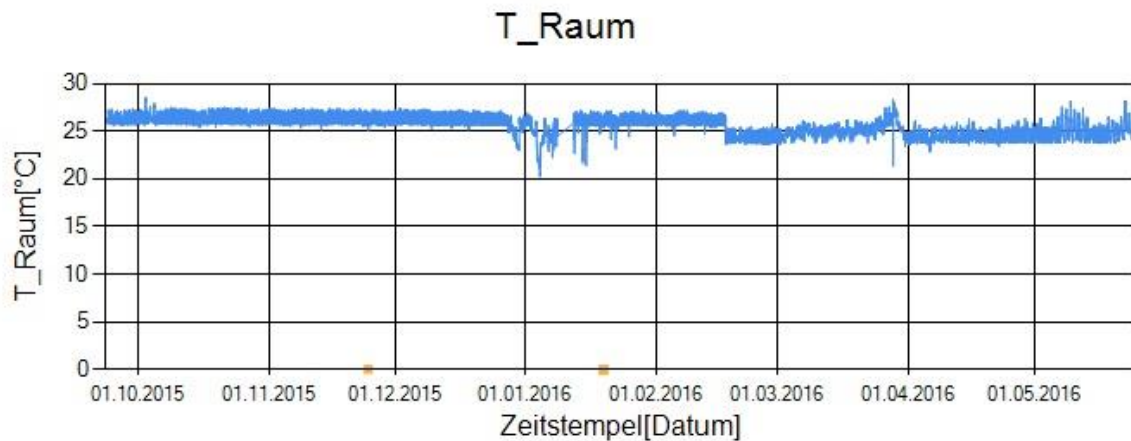


ABBILDUNG 12 BEISPIELDIAGRAMM DER MINUTENWERTE, 34665 PUNKTE PRO GRAPH

Der letzte Schritt der Datenverarbeitung im Programm ist die mathematische Auswertung. Während des Probelaufs dauerte das Übertragen der Daten an das neue Form zusammen mit der Berechnung der statistischen Werte und dem Erstellen des Graphen 1944 Millisekunden. Dieser Vorgang verlief also schneller als im vorherigen Form, was darauf zurückzuführen ist, dass in diesem Fall lediglich der Graph der gerade gewählten Spalte erstellt wird. In der graphischen Auswertung werden hingegen die Graphen aller verfügbaren Spalten erstellt, um den Nutzer einen schnelleren Zugriff zu erlauben. Bei der Kontrolle der statistischen Werte und der Graphenstruktur wurden keine Fehler gefunden.

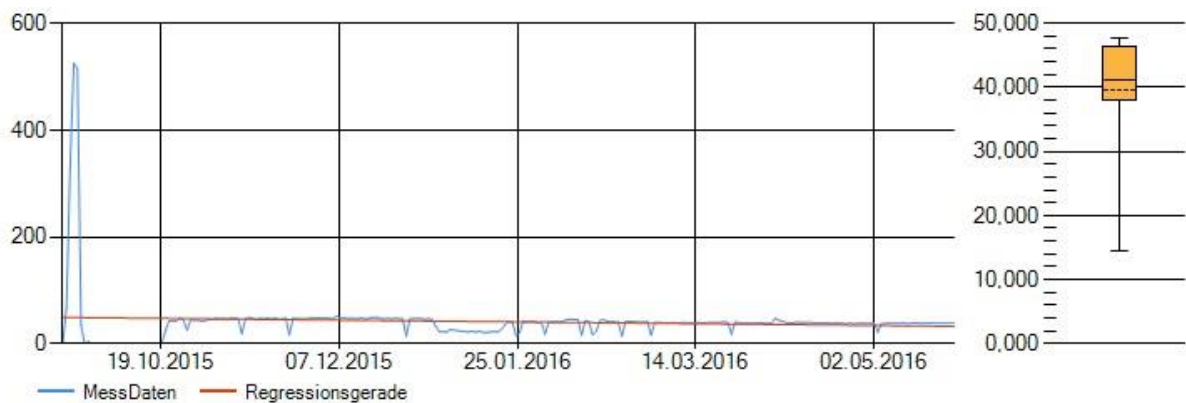


ABBILDUNG 13 GRAPH DER SPALTE Q_EL_FI AUS DER TAGESWERT-TABELLE MIT DAZUGEHÖRIGEM BOXPLOT UND LINEARFIT

Unter Berücksichtigung der an das Programm gestellten Erwartungen lässt sich sagen, dass der derzeitige Stand den geforderten Standards entspricht und eine publikationsfähige Anpassung der Diagramme erlaubt. Die Bearbeitung der verfügbaren Dateien ist fehlerfrei möglich. Eine Optimierung des Codes könnte zwar eine Beschleunigung der Arbeitsschritte mit sich bringen, würde aber bei den zurzeit verwendeten Datenmengen einen eher geringen Nutzen bringen. Im Rahmen der visuellen Aspekte des Programms gibt es stets Raum zur Ergänzung und Erweiterung der verfügbaren Optionen.

7. Ausblick

Es zeigte sich während der Bearbeitung des Projektes, dass die benötigten Rechenzeiten minimiert und die Auswertbarkeit vereinheitlicht werden können, indem Werte der Zwischenauswertung und bestimmte Metadaten bereits in der Datenbank hinterlegt werden, um zu verhindern, dass diese während der Laufzeit ermittelt werden müssen.

Dazu gehören beispielsweise die Datentypen der einzelnen Spalten, um eine leichtere Unterscheidung von grafisch kompatiblen und inkompatiblen Messwertsätzen zu erlauben, oder die Normal- und Grenzbereiche, um schneller eine optisch ansprechende Einteilung des Diagramms zu erreichen. Eine weitere Möglichkeit wäre auch die Bereitstellung von Informationen zu dem Inhalt der einzelnen Spalten. Derartige Beschreibungen können nach Anpassung des Programms dem Nutzer während der Auswahl der entsprechenden Spalte angezeigt und somit ein größerer Informationsgehalt vermittelt werden. Diese Möglichkeit wurde bei der Erstellung dieses Auswerteprogramms erkannt und resultierend daraus wird vom Projektpartner autosoft in Zusammenarbeit mit dem IGB eine Metadatentabelle zur Integration in die Datenbank erarbeitet.

Eine wichtige weitere Entwicklung des Programms ist die Einbindung mehrerer Standorte und ihrer Datenbanken im Netzwerk, um eine vollständige Bearbeitung der Daten des INAPRO-Projektes zu ermöglichen. Dabei müssen jedoch Fragestellungen zu Kompatibilität und Infrastruktur bearbeitet werden, damit diese Daten benutzerfreundlicher werden. Bei der endgültigen Implementation der finalen Datenbankstruktur ist auch eine effizientere und schnelle Anpassung des Programms an neue Aufgabenstellungen denkbar.

Zukünftig sollte die Auswertesoftware dahingehend erweitert werden, dass die Datensätze aus den verschiedenen Datenbanken (=Standorten) miteinander verglichen werden können und darüber hinaus verschiedene Zeiträume (z.B. verschiedene Jahre) desselben Datensatzes.

Letztendlich ist das erstellte Programm aufgrund seiner Verwendung von Laufzeiterkennung aller wichtigen Informationen zum Inhalt der Datenbank sehr flexibel. Ein Nachteil dieser Flexibilität ist jedoch auch die sehr unspezifische Erarbeitung von Informationen, wie zum Beispiel die fehlende Möglichkeit die Einheiten einer gegebenen Spalte aus dem Messwertsatz selbst zu ermitteln.

Wenn Metadaten dem Nutzer zur Verfügung gestellt werden können, würde sich die Nutzerfreundlichkeit des Programms erheblich erhöhen. Zu beachten ist dabei, dass bei einer Weiterentwicklung möglichst die Flexibilität des Programms erhalten bleibt, die eine leichte Anpassung an alle weiteren Entwicklungen des INAPRO-Projektes und der darin ermittelten Messwerte gewährleistet.

8. Anhang

Inhaltsverzeichnis Anhang	
8.1 Programmablaufplan.....	27
8.2 Nutzerhandbuch.....	28
8.2.1 Installation.....	28
8.2.2 Hauptmenü.....	28
8.2.3 Graphische Auswertung	29
8.2.4 Mathematische Auswertung	30
8.3 Beispiel für eine .ini-Datei	31
8.4 Abbildungsverzeichnis.....	32
8.5 Tabellenverzeichnis	32
8.6 Glossar	33

8.1 Programmablaufplan

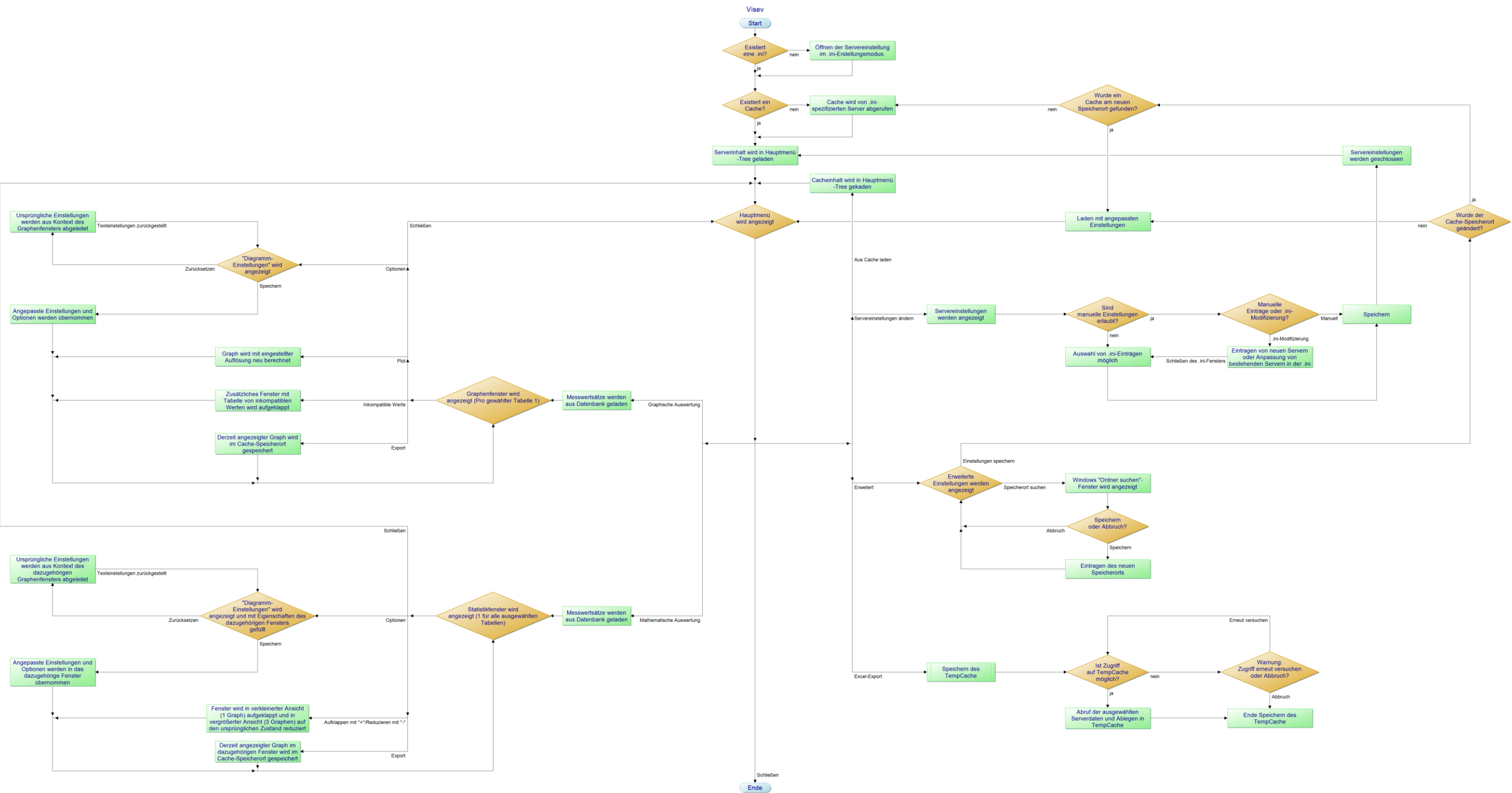


ABBILDUNG 14 PROGRAMMABLAUFPLAN

8.2 Nutzerhandbuch

8.2.1 Installation

Die Installation des Programms besteht lediglich aus der Platzierung der .exe in einem vorhandenen Ordner. Dabei muss garantiert sein, dass Zugriffsrechte auf den Ordnerinhalt bestehen, da sonst Probleme bei der Ausführung des Programms auftreten können.

Sobald die .exe ausgeführt wird, ohne dass eine passende .ini sich in demselben Ordner befindet, wird das Servereinstellungs-Form im .ini-Erstellungsmodus ausgeführt. Dabei wird verlangt, dass ein valider Login zu einem bestehenden Server eingetragen wird. Anschließend wird die Verbindung zu dem eingetragenen Server aufgebaut und, falls auch kein Cache des gewählten Servers vorhanden ist, ein neuer Cache angelegt.

Nachdem diese Schritte fehlerfrei durchgeführt wurden, wird das Hauptmenü angezeigt.

8.2.2 Hauptmenü

Das Hauptmenü bildet den Knotenpunkt aller verfügbaren Operationen. Hier können die Servereinstellungen geändert und die Auswertungsoptionen angepasst werden. Außerdem stehen graphische und mathematische Auswertungen zur Verfügung.

Wenn der Serverinhalt (bzw. Cacheinhalt) geladen wurde, wird die verfügbare Datenstruktur im zentralen TreeView-Fenster angezeigt. Mit einem Klick auf „+“ bzw. „-“ neben den Tabellennamen können die dazugehörigen Spaltennamen auf- bzw. zugeklappt werden.

Sollte die Checkbox neben dem Tabellennamen angeklickt werden, können so alle

darunterliegenden Spaltennamen zugleich an- oder abgewählt werden. Sofern die entsprechende Option unter „Erweitert“ gewählt ist, werden dabei automatisch grafisch inkompatible Spalten (in rot markiert) nicht mit ausgewählt. Die grau unterlegten Spalten sind auswertungskritische Informationen, wie zum Beispiel die Zeitstempel der gemessenen Daten oder die IDs der Messwertzeilen, sollte kein Zeitstempel vorhanden sein. Diese Spalten werden bei Auswahl einer beliebigen anderen Spalte unter derselben Tabelle automatisch angewählt und können nicht für sich alleine geladen werden.

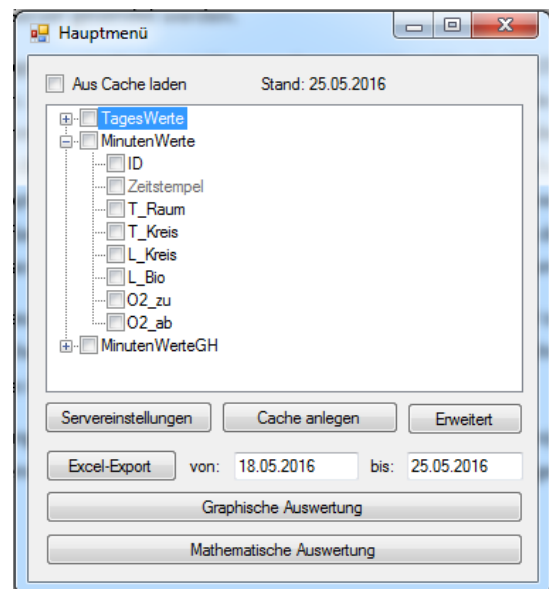


ABBILDUNG 15 DAS HAUPTMENÜ

Die ausgewählten Spalten (bzw. Tabellen) sind jene, welche bei Auswahl der graphischen oder mathematischen Auswertung in die dazugehörigen Forms geladen werden und zur weiteren Bearbeitung zur Verfügung stehen.

Weitere Optionen sind die Verwendung eines Cache anstelle eines Servers als Ursprung der geladenen Daten, die Anpassung der Serverzugriffsdaten (und die Eintragung von mehreren Servern), das erzwungene Neuanlegen eines Cache bzw. die Anlage eines teilweisen Cache zur Ablage von ausgewählten Messwerten. Dabei wird der eingetragene Zeitraum aus den im TreeView ausgewählten Spalten in ein separates TempCache geladen.

8.2.3 Graphische Auswertung

In der graphischen Auswertung stehen grundlegende Optionen zur Visualisierung und optischen Anpassung der ausgewählten Messdaten bereit. Dabei werden lediglich kompatible Spalten bearbeitet; alle weiteren inkompatiblen Informationen werden in einer aufklappbaren Tabelle am rechten Formrand abgelegt.

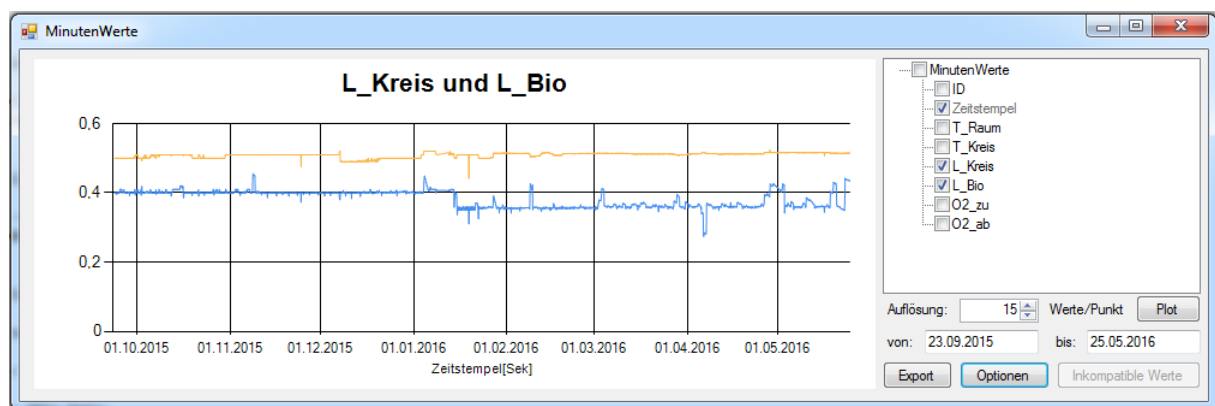


ABBILDUNG 16 EIN GRAPHISCHES AUSWERTUNGSFENSTER

Wie am rechten Rand zu sehen ist, befindet sich auch in diesem Form ein TreeView-Fenster, in welchem die darzustellenden Daten ausgewählt werden können. Dabei gelten dieselben Regeln wie im TreeView-Fenster des Hauptmenüs, nur dass in diesem Fall die rot unterlegten Spalten bereits aussortiert wurden, weil sie sich in der Tabelle für inkompatible Werte befinden.

Die Auflösung der Graphen kann über den Auflösungsregler am unteren Rand des TreeView-Elements eingestellt und per „Plot“ übernommen werden. Dieser Button wird auch verwendet, um den angezeigten Zeitbereich zu verändern, der mithilfe der Textboxen darunter angepasst werden kann.

Des Weiteren kann das angezeigte Diagramm im derzeitigen Zustand in Form einer Vektorgrafik abgespeichert werden. Das dabei verwendete Format ist EMF Dual. Alle Beschriftungen sowie die Legende und potentielle Eingrenzungen des Graphen auf einen bestimmten Wertebereich können über die Optionen angepasst werden.

8.2.4 Mathematische Auswertung

Die mathematische Auswertung bietet einen Zugang zu grundlegenden statistischen Eigenschaften der gewählten Daten. Dabei werden Median, Mittelwert, Varianz und Quantile der Spalten ermittelt und angezeigt. Außerdem ist eine Auftragung von Spalten gegeneinander und die lineare Regression der Messwerte möglich.

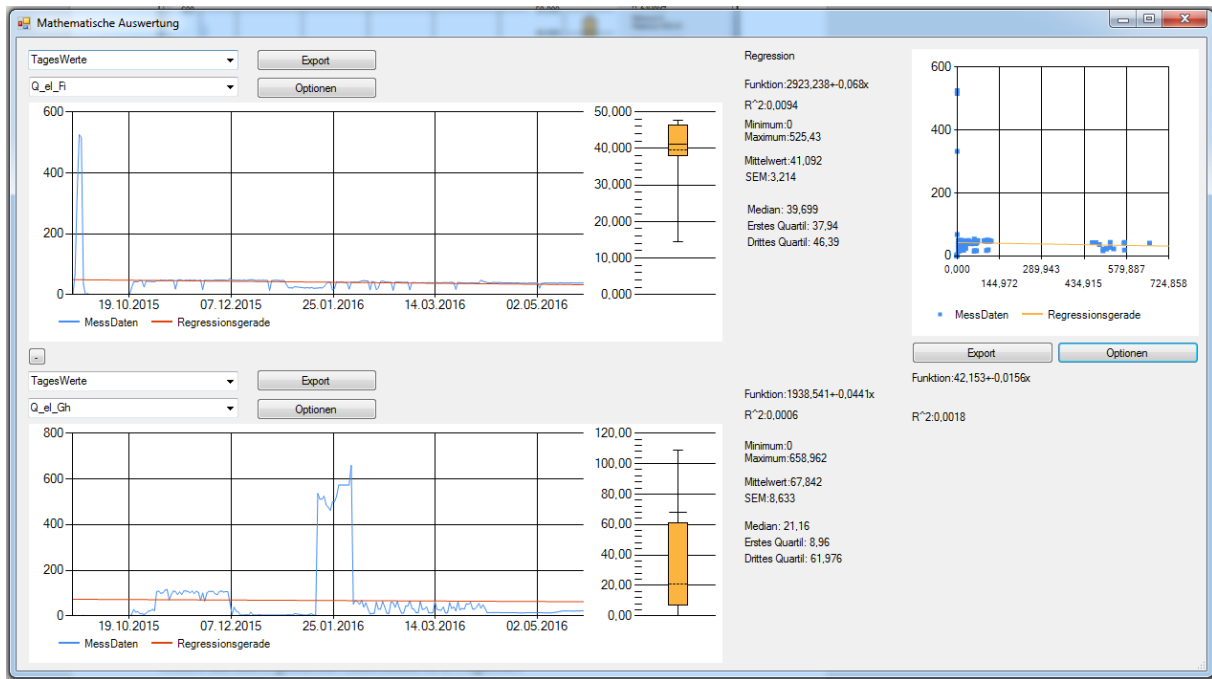


ABBILDUNG 17 AUFGEKLAPPTES FENSTER DER MATHEMATISCHEN AUSWERTUNG

Im Ursprungszustand ist dieses Fenster auf das obere linke Diagramm beschränkt, wobei alle verfügbaren Spalten dargestellt werden können und die automatisch berechneten statistischen Informationen rechts davon abgebildet werden. Die Auswahl der gewünschten Spalte bzw. Tabelle findet mithilfe der beiden Dropdown-Listen in der oberen linken Ecke statt. Sollte sich die ausgewählte Tabelle ändern, wird automatisch die erste Spalte aus dieser Tabelle ausgewählt und dargestellt.

Sobald das Fenster aufgeklappt wird, wird automatisch das untere linke Diagramm erstellt. Dazu wird die erste Spalte aus der ersten Tabelle geladen, die noch nicht im oberen Diagramm gewählt ist. Anschließend wird im rechten Diagramm der Inhalt beider Diagramme gegeneinander aufgetragen und nach einer möglichen linearen Abhängigkeit gesucht.

Die Dropdown-Listen der beiden linken Diagramme sind so miteinander verbunden, dass nie zwei gleiche Spalten dargestellt werden können. Graphisch inkompatible Spalten werden, wie bereits in der graphischen Auswertung, aussortiert und nicht in der Liste aufgezählt.

Mithilfe der „Export“- und „Optionen“-Buttons neben jedem Fenster kann das dazugehörige Diagramm als Vektorgrafik exportiert oder seine Beschriftung mithilfe der aus der graphischen Auswertung bekannten Optionen angepasst werden.

8.3 Beispiel für eine .ini-Datei

TABELLE 3 – BEISPIELSTRUKTUR FÜR EINE EINLESBARE .INI-DATEI

frei verfügbare Zeichen 1 2 3 4 5	← Der Beginn des Dokuments und alle Angaben, welche nicht unter einem Befehl in eckigen Klammern stehen, sind beliebig und werden nicht gelesen.
[Standort] Test1 Test2 Test3 Nicht gelesene Zeichen 1 2 [Test1] testuser1 passwort1 ServerURL1 Datenbank1 15 yes [Test2] testuser2 passwort2 127.0.0.1 Datenbank2 30 yes [Test3] testuser3 passwort3 IGBCOMPUTER204 Datenbank3 150 yes [Test4] testuser4 passwort4 LokalerPC Datenbank4 15 yes	← Der [Standort]-Block bietet eine Liste aller validen Server bzw. Logindaten. Sollte ein Block hier nicht mit Namen aufgeführt sein, wird er nicht berücksichtigt, wie z.B. [Test4]. Alle Daten, welche nicht unmittelbar unter [Standort] stehen, werden nicht gelesen. ← Die Namen der verfügbaren Server werden in die Dropdown-Liste der Servereinstellungen eingelesen und sollte ein Server gewählt werden, werden die auf seinen Namen folgenden Angaben in das Form eingelesen und für den Login genutzt. ← Nutzernamen, Passwort und URL sind vom gewählten Server abhängig. Die URL ist in einem lokalen Netzwerk der Name des Rechners. Die Timeout-Angabe dient zur Bestimmung eines Zeitfensters, in welchem der Server antworten muss, bevor die Verbindung getrennt wird. Die Einstellung „trusted connection“ muss im Rahmen des Programms immer eine „true“-Aussage sein, was in dieser Datei als „yes“ eingelesen wird.

8.4 Abbildungsverzeichnis

Abbildung 1 Beispielabbildung der Aquaponik-Anlage im DRAPS-Aufbau	4
Abbildung 2 Hauptmenü-Form.....	8
Abbildung 3 Server-Einstellungen-Form	11
Abbildung 4 Erweiterte Einstellungen-Form	13
Abbildung 5 Zugriff gesperrt-Form.....	14
Abbildung 6 Ladebalken-Form	14
Abbildung 7 Diagramm- Form	15
Abbildung 8 Diagramm-Einstellungen-Form	18
Abbildung 9 Mathematisches Auswertungs-Form (aufgeklappt)	19
Abbildung 10 Mathematisches Auswertungs-Form (klein)	19
Abbildung 11 Beispieldiagramm der Tageswerte, 241 Punkte pro Graph	23
Abbildung 12 Beispieldiagramm der Minutenwerte, 34665 Punkte pro Graph.....	24
Abbildung 13 Graph der Spalte Q_el_Fi aus der Tageswert-Tabelle mit dazugehörigem Boxplot und Linearfit	24
Abbildung 14 Programmablaufplan	27
Abbildung 15 Das Hauptmenü	28
Abbildung 16 Ein graphisches Auswertungsfenster	29
Abbildung 17 Aufgeklapptes Fenster der mathematischen Auswertung	30

8.5 Tabellenverzeichnis

Tabelle 1 – Zeitplan des Projekts.....	5
Tabelle 2 – Tabelle der Sensoren im Standort Abtshagen	7
Tabelle 3 – Beispielstruktur für eine einlesbare .ini-Datei.....	31

8.6 Glossar

Aquakultur: Die Zucht von aquatischen Organismen (darunter Fische) in einer abgeschlossenen und kontrollierten Umgebung.

Aquaponik: Die Vereinigung von erdlosem Pflanzenanbau (Hydroponik) und Aquakultur in einem System, welches eine besonders umweltverträgliche und nachhaltige Produktion ermöglicht.

C#: Eine Programmiersprache, die die intuitive und unkomplizierte Erstellung von Programmen unter Verwendung von sogenannten „Forms“ erlaubt.

Forms: Forms sind die „Leinwand“ der Programmfenster in Visual Studio und bilden somit den Grundbaustein eines Programms. Auf ihnen können die Steuerelemente angeordnet werden, die mithilfe der Methoden gesteuert werden.

Methoden: Methoden sind die Codeblöcke, welche nach Aufruf durch anderen Code oder Events ausgeführt werden. Sie können weitere Methoden aufrufen oder Steuerelemente manipulieren.

Quantil: Ein Abschnitt einer Menge, welcher einen festen Prozentsatz der geordneten Menge ausmacht. Ein Quartil (Viertelquantil) wäre beispielsweise die Menge der unteren 25% der Ursprungsmenge.

RAS: Eine Abkürzung für „Rezirkulierendes Aquakultur-System“. Der Abschnitt der Anlage, der der Fischzucht dient.

Visual Studio: Visual Studio ist eine Entwicklungsumgebung, die von Microsoft zum Programmieren in verschiedenen Hochsprachen, u.a. C# entwickelt wurde.

Windows Enhanced MetaFile(Emf): Ein Speicherformat von Vektorgrafiken, welche ein gegebenes Bild als Sammlung von manipulierbaren Punkten und Geraden darstellt. Das Enhanced Format beinhaltet zudem die Möglichkeit mit parametrisch modellierten Kurven zu arbeiten. Das im Programm verwendete EMF Dual-Format verbindet eine EMF- mit einer EMF Plus-Grafik, welche eine erweiterte Struktur aufweist.

9. Quellen

Goddek S, Espinal CA, Delaide B, Jijakli MH, Schmautz Z, Wuertz S und Keesman KJ (2016)
Navigating towards Decoupled Aquaponic Systems: A System Dynamics Design Approach.
Water 2016 Volume 8 Issue 7

Karimanzira D, Keesman KJ, Kloas W, Baganz D und Rauschenbach T (2016)
Dynamic modeling of the INAPRO aquaponic system. Aquacultural Engineering Volume 75

Kloas W, Groß R, Baganz D, Graupner J, Monsees H, Schmidt U, Staaks G, Suhl J, Tschirner M,
Wittstock B, Wuertz S, Zikova A und Rennert B (2015)
A new concept for aquaponic systems to improve sustainability, increase productivity, and
reduce environmental impacts. Aquaculture Environment Interactions

Suhl J, Dannehl D, Kloas W, Baganz D, Jobs S, Scheibe G und Schmidt U (2016)
Advanced Aquaponics: Evaluation of intensive tomato production in aquaponics vs.
conventional hydroponics. Agricultural Water Management

<http://www.inapro-project.eu/> (abgerufen am 18.11.2016, 20:30)

https://www.facebook.com/inaproproject/about/?ref=page_internal
(abgerufen am 18.11.2016, 20:30)

[http://www.diplomatisches-magazin.de/business-03-2015-
de/A3/?PHPSESSID=ev68br2loeotdnv27tnubqraq4](http://www.diplomatisches-magazin.de/business-03-2015-de/A3/?PHPSESSID=ev68br2loeotdnv27tnubqraq4) (abgerufen am 18.11.2016, 20:30)

Kühnel A (2015)
C# 6 mit Visual Studio 2015: Das umfassende Handbuch: Spracheinführung,
Objektorientierung, Programmiertechniken. Rheinwerk Computing, 7. Auflage vom 26.
Oktober 2015

Theis T (2015)
Einstieg in C# mit Visual Studio 2015: Ideal für Programmieranfänger geeignet. Rheinwerk
Computing, 4. Auflage vom 28. September 2015

10. Danksagung

Ich möchte mich bei Prof. Dr. Werner Kloas für die freundliche und hilfsbereite Betreuung sowie die Möglichkeit zur Durchführung dieses Bachelorprojekts in seiner Arbeitsgruppe bedanken. Ebenso möchte ich ihm und meinem Zweitgutachter Dr. Klaus Knopf für ihre Zeit und Aufmerksamkeit bei der Bewertung meiner Abschlussarbeit danken. Meine Dankbarkeit geht auch an Dr. Daniela Baganz und Dr. Georg Staaks für ihre Freundlichkeit, Hilfsbereitschaft und technische Unterstützung während meiner Zeit am IGB. Besonderer Dank gebührt auch der IT-Abteilung und Verwaltung am IGB, welche mir meine zahlreichen Fragen zu Technik und Beruf immer verständlich und hilfreich beantwortet haben, und den Mitarbeitern der Firma autosoft, welche die im Projekt verwendete Datenbank bereitgestellt und entwickelt haben. Weiterhin möchte ich mich bei meiner Mutter für ihre Zeit und Aufmerksamkeit bei der Überarbeitung meiner Bachelorarbeit bedanken.

Selbstständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Berlin, den 17.12.2016

Marti Ritter