



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

## Bachelorarbeit

### Fakultät Life Sciences Studiendepartment Biotechnologie

Entwicklung einer grafischen Oberfläche und Datenbank zur Nutzeranmeldung und Dokumentation von Equipment einer BioProcess-Designer Software

Katharina Dahlmann- Matrikelnummer: 201 99 68

Hamburg- 19. Februar 2015

Gutachter: Prof. Dr. Boris Tolg (HAW Hamburg)

Gutachter: Prof. Dr. Ernst A. Sanders (HAW Hamburg)



## Eidesstaatliche Erklärung

Ich versichere, dass ich die vorliegende Seminararbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Alle Stellen, die dem Wortlaut oder dem Sinne nach anderen Texten entnommen sind, wurden unter Angabe der Quellen (einschließlich des World Wide Web und anderer elektronischer Text- und Datensammlungen) und nach den üblichen Regeln des wissenschaftlichen Zitierens nachgewiesen. Dies gilt auch für Zeichnungen, bildliche Darstellungen, Skizzen, Tabellen und dergleichen. Mir ist bewusst, dass wahrheitswidrige Angaben als Täuschungsversuch behandelt werden und dass bei einem Täuschungsverdacht sämtliche Verfahren der Plagiatserkennung angewandt werden können.

---

Ort, Datum

---

Unterschrift



## **Abstract**

In dieser Bachelorarbeit wurde die grafische Oberfläche und die Datenbank zur Nutzeranmeldung und Dokumentation von Equipment einer BioProcessDesigner Software entwickelt. Es wurde ein Menü generiert, um die Planung und Ausführung bioverfahrenstechnischer Prozesse zu ermöglichen. Durch die entwickelte Nutzerverwaltung ist ein Rechtesystem für die Software- und Projektstruktur entstanden, die eingeschränkte Rechte und Nutzungsvielfalt definiert. Durch die Registrierung werden neue Nutzer in der Datenbank angelegt. Durch die Charakterisierung eines Nutzers in der Datenbank werden die wichtigsten Informationen zum Kontaktieren gespeichert. Zudem werden doppelte Registrierungen vermieden. Beim Login wird berücksichtigt, dass nicht bestätigten Nutzern der Login verweigert wird. Nach dem Login wird der Zugriff auf Softwareteile an das Rechtesystem der Nutzerverwaltung angepasst. Als Beginn der Prozessplanung und -gestaltung wurde die Editierung und Kategorisierung von Equipment in der Datenbank und der Softwareoberfläche umgesetzt. Durch die Datenbankstruktur des Equipments wurde eine hohe Flexibilität ermöglicht.



# Inhaltsverzeichnis

1	Einführung in die Idee der BIOPROCESSDESIGNER(BPD) Software.....	3
1.1	Zielsetzung Oberfläche und Datenbankdesign für Nutzer- und Equipmentverwaltung....	5
2	Grundlagen zum Programmieren und zur Softwareentwicklung .....	6
2.1	JQuery- Das Framework des BPD.....	6
2.2	V- Modell XT- Ein Standardvorgehen zur Software- und Projektentwicklung.....	8
2.3	Sonstige verwendete Komponenten zur Entwicklung der Software .....	9
2.3.1	HTML-Die Grundlage des Programms.....	9
2.3.2	PHP als Grundlagenerweiterung .....	10
2.3.3	CSS.....	10
2.3.4	MySQL .....	10
2.3.4.1	Einführung Entity Relationship Modell .....	11
3	Aufbau des BPD.....	13
3.1	Nutzer- und Rechteverwaltung.....	13
3.2	Das Menü- Navigation durch den BPD.....	20
3.2.1	Login und Registrierung der Nutzer .....	21
3.2.2	Administration.....	22
3.2.3	BioProcessDesigner .....	23
3.2.4	Execution.....	23
3.2.5	Reporting.....	24
3.3	Der Equipment Manager – Die graphische Oberfläche .....	25
3.4	Datenbankdesign - Koordination von Nutzern und Equipment im virtuellen Raum .....	26
3.4.1	Das Datenbank Design der User.....	26
3.4.2	Das Datenbank Design des Equipment Managers .....	29
3.5	Funktionaler Verbund – Kommunikation zwischen graphischer Oberfläche und der Datenbank.....	34
3.5.1	Login .....	34

3.5.2	Registrierung.....	36
3.5.3	Neue Nutzer zulassen .....	39
3.5.4	Userclass eines Nutzers verändern .....	41
3.5.5	Equipment Manager.....	43
3.5.5.1	Anzeigen der Kategorien .....	43
3.5.5.2	Bearbeiten von Equipment Kategorien.....	45
3.5.5.3	Hinzufügen von Equipment Kategorien.....	45
3.5.5.4	Löschen von Equipment Kategorien.....	48
3.5.5.5	Anzeigen des angelegtem Equipments einer subcategory .....	50
4	Abschließende Betrachtungen .....	52
4.1	Zusammenfassung- Der BPD als Ganzes.....	52
4.2	Ausblick: Vorschläge zur Weiterentwicklung des BPD .....	52
4.3	Danksagung .....	53
5	Anhang.....	54
5.1	Literaturverzeichnis .....	54
5.2	Anhang A- Oberfläche des Equipment Managers .....	56
5.3	Anhang B- Programmcode.....	64

## 1 Einführung in die Idee der BIOPROCESSDESIGNER(BPD) Software

Ziel des BPD ist es einen Fermentationsprozess planen, durchführen, dokumentieren und einheitlich präsentieren zu können. Arbeitsschritte und Analysenmethoden können klar definiert werden und zur Gestaltung von neuen Prozessen oder zur Verbesserung vorhandener Prozesse genutzt werden.

Ein Prozess kann vorab am Computer detailliert geplant und vor der Umsetzung im Labor simuliert werden. Konflikte, die hier entstehen, werden identifiziert und beseitigt.

Anschließend kann der geplante Prozess durchgeführt werden. Um den Nutzungsraum zu erweitern, soll der Umstieg vom Desktop-PC oder Laptop auf ein Tablet oder Smartphone möglich sein. Der Prozessablauf wird darauf mit einem einheitlichen Schema dargestellt, auf welches während der Durchführung jederzeit zurückgegriffen werden kann. Die Software führt den Nutzer chronologisch durch die Arbeitsschritte. Vom Einwiegen der Chemikalien über das Kalibrieren der Pumpe bis zum Beenden des Prozesses wird der Nutzer geführt und begleitet. Während des Prozesses wird der Verlauf in regelmäßigen Zeit- oder Ereignisabständen dokumentiert. Mit einem Abfrageformular für Kommentare können über gezielte Fragen relevante Informationen festgehalten werden, deren Bedeutung im Nachhinein ausgewertet wird. Durch die einheitliche und übersichtliche Prozessdokumentation und Ergebnisdarstellung kann eine verbesserte Vergleichbarkeit zu ähnlichen Prozessen geschaffen werden. Änderungen oder Gemeinsamkeiten im Versuchsverlauf können schneller identifiziert werden. Es soll eine Messergebnisverwaltung ähnlich dem LIMS System<sup>1</sup> entstehen. Durch die einheitliche und genaue Dokumentation der Arbeitsschritte können GMP (Good manufacturing practice) Standards erreicht werden. Durch eine webbasierte Struktur ist die Software jederzeit zugänglich und erleichtert den Laboralltag. Die Idee des BPD stammt von Prof. Dr. Ernst Sanders, Labor für Bioverfahrenstechnik der HAW Hamburg. Angeregt durch Software wie CHEMCAD<sup>2</sup> und dem BIOPRO DESIGNER<sup>3</sup> entstand der Gedanke einer Software, die auf den Anforderungsbereich der eigenen bioverfahrenstechnischen Anwendungen angepasst ist. Die bisher genutzte Software spezialisiert sich mehr auf verfahrenstechnische Prozesse oder andere Anwendungsbereiche. Oft lässt sich Software zudem nur auf herstellereigene Hardware anwenden.

---

<sup>1</sup> Diese aus Kategorien bestehende Software befasst sich mit der Datenverarbeitung von Analytik Laboren (IMCOR GmbH 2014, 2014)

<sup>2</sup> Hersteller: Chemstations - Simulationssoftware für stationäre Prozesse der Verfahrenstechnik (CHEMSTATIONS EUROPE GMBH, 2015)

<sup>3</sup> Hersteller: Pace Report - Software zur erleichterten Kontrolle der betriebeigenen chemischen Mikroprozessoren zur Erfassung der Umgebung eines Chemischen Systems (Consortium, PACE, 2004-2008)

Der fertige BioProcessDesigner bietet im finalen Zustand viele Vorteile für Prozesse der Bioverfahrenstechnik und für den Laboralltag. Durch eine Nutzer- und Projekthierarchie werden Rechte klar definiert. So wird der Zugriff unbefugten Externen verweigert.

In Bezug auf das Equipment existiert immer eine aktuelle Inventarliste. Die Informationen zum Equipment liegen gesammelt an einem Ort. Alle Details sind leicht und komfortabel abrufbar. Durch Planung der Equipmentnutzung können Konflikte minimiert werden, dadurch entsteht Zeitersparnis.

Die Software ist auf Englisch programmiert. Um die Bezeichnungen in dieser Arbeit zu wiederzufinden, wurden viele englische Bezeichnungen übernommen.

Dies Software ist online abrufbar: <http://www.ls.haw-hamburg.de/~bpd/>

## 1.1 Zielsetzung Oberfläche und Datenbankdesign für Nutzer- und Equipmentverwaltung

Nach den Vorgaben durch Prof. Dr. Ernst Sanders und Prof. Dr. Boris Tolg sollen folgende Funktionen abgedeckt werden:

Die Grundlage der Software bildet das Menü. Die Datenbank ermöglicht zusammen mit der Softwareoberfläche den Login und Registrierung sowie die Verwaltung des Equipments und der Nutzer.

Durch das Registrieren neuer Nutzer und das Login kann sowohl die Rechteverwaltung als auch eine Sicherung der Projekthinhalte gegenüber Externen oder anderen Projektgruppen gewährleistet werden.

Mit dem Anlegen und Verwalten von Equipment kann dieses für die Prozessplanung genutzt werden. Auf die Equipmentstruktur baut die Weiterentwicklung der Software auf, die nicht Teil dieser Arbeit ist.

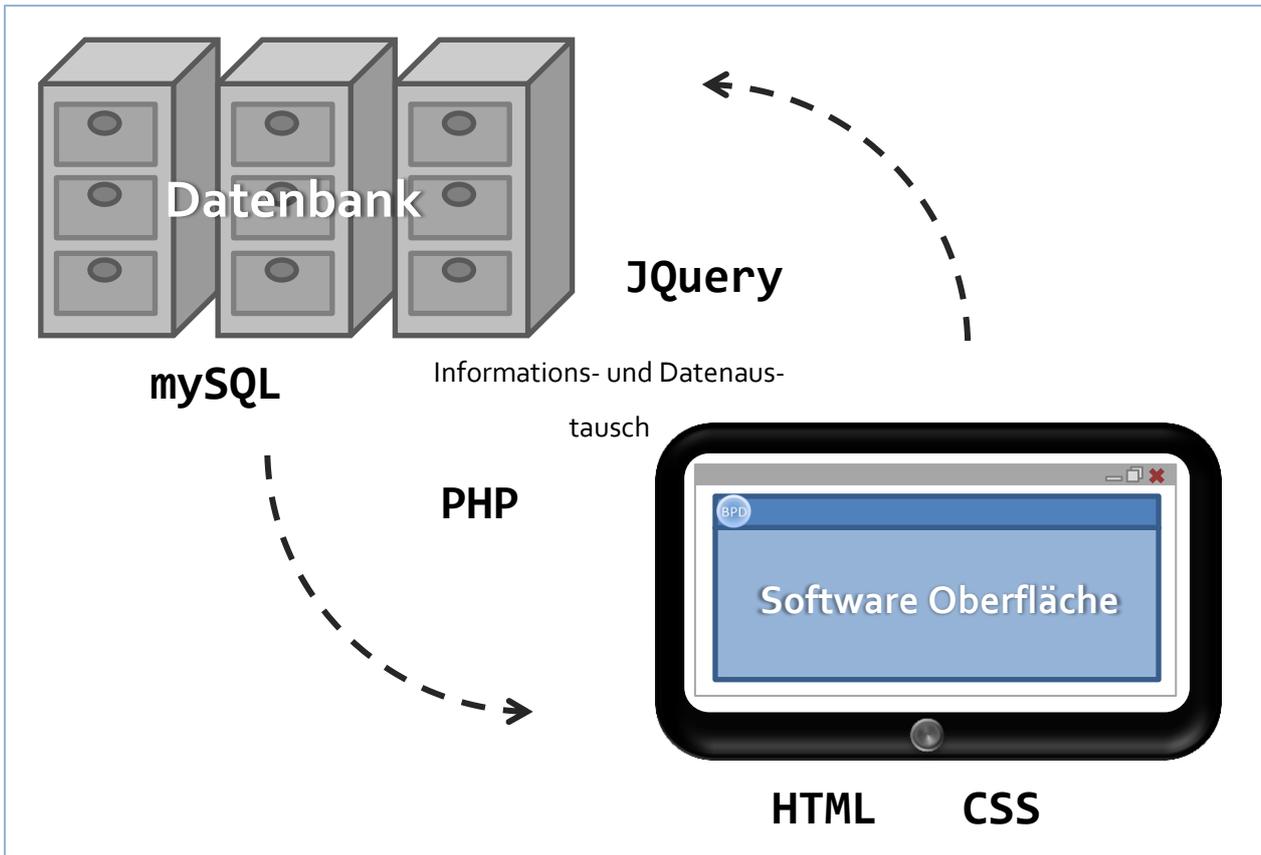
Für das Login kann sich der Nutzer mit seiner E-Mail-Adresse einloggen. Für die Registrierung neuer Nutzer sind bestimmte Pflichtfelder vorhanden. Jeder Nutzer darf sein eigenes Passwort unter vorgegebenen Sicherheitsbedingungen erstellen. Die Angaben werden in der Datenbank gespeichert, wobei das Passwort nicht im Klartext in der Datenbank zu finden ist.

Für den Equipment Manager wird eine einheitliche und intuitive Benutzeroberfläche geschaffen. Es soll neues Equipment hinzugefügt werden und vorhandenes Equipment geändert oder gelöscht werden können.

## 2 Grundlagen zum Programmieren und zur Softwareentwicklung

Die BPD-Software ist webbasiert. Die Programmiersprachen und Planungsmethoden zur Softwareentwicklung des BPD werden in diesem Kapitel kurz dargestellt.

In *Abb. 1* werden die Bestandteile des BioProcessDesigners dargestellt. Diese sollen im Folgenden näher erläutert werden.



*Abb. 1* Verbindungselemente der BPD Software

### 2.1 JQuery- Das Framework des BPD

Ein Framework ist eine Erweiterung zu einer Programmiersprache, die dem Anwendungsentwickler bestimmte sich wiederholende Aufgaben abnimmt (Vollendorf, et al., 2014).

Eine Analogie wäre das Nutzen eines Akkuschraubers statt eines Schraubendrehers, um mehrere Schrauben in ein Möbelstück zu drehen. Der Akkuschrauber übernimmt die immer wiederkehrende Drehbewegung für den Monteur. Ein Framework ist eine Art Werkzeugsammlung, welches dem Programmierer die Arbeit bei üblichen Handlungsschritten erleichtert oder sogar abnimmt. Der Vorteil eines Frameworks besteht darin, dass es einen kooperierenden Methodenverbund darstellt, der zusammenhängend arbeiten kann.

JQuery bietet umfangreiche Funktionen zur Navigation und Manipulation des Document Object Modell (DOM) welches als Schnittstelle für den Zugriff auf HTML Dokumente und objektorientierter Programmierung wie mit php darstellt.

JQuery hilft eine fortschrittliche Benutzerschnittstelle auf einer Webseite zu schaffen, die animiert interaktiv im Hintergrund läuft.

John Resig entwickelte 2006 dieses JavaScript Framework (jQuery Foundation, 2015). Es bietet eine Vereinfachung der Skriptsprache, welche ursprünglich dafür entwickelt wurde, dynamische HTML-Webbrowser interaktiv für Benutzer zu gestalten.

Es stellt dabei keine neue JavaScript-Funktionalität dar, sondern vereinfacht gerade die umfangreichen JavaScript-Methoden. Es gilt als Leitfaden, der den Zugang zu JavaScript erleichtert.

JQuery macht es beispielsweise möglich, die Inhalte einer HTML-Seite zu verarbeiten, ohne diese gleich neu laden zu müssen (Vollendorf, et al., 2014).

Dieser Vorteil wurde vor allem im Menü und in der Softwareoberfläche der Equipmentverwaltung genutzt. Für diese Arbeit wurde die Version 1.11.2 genutzt.

## 2.2 V- Modell XT- Ein Standardvorgehen zur Software- und Projektentwicklung

Wesentliches Ziel des Modells ist es, die Komplexität von Softwareprojekten zu überschauen und dadurch besser abarbeiten zu können. Das V- Modell XT 1.3 ist seit 2009 ein fester Bestandteil der Entwicklung von Software.

Das V-Modell definiert anstehende Arbeitsschritte, die als Bausteine schematisch dargestellt werden. Ihre Interaktionen werden über Verbindungspfeile veranschaulicht.

Im V-Modell wird das „Was“ mehr fokussiert als das „Wie“, somit ist die zeitliche Abfolge in den Hintergrund gedrängt und der Fokus liegt auf dem Produkt.

„Die Kernidee des V-Modells liegt in der Gegenüberstellung von Spezifikation und Zerlegung sowie Realisierung und Integration (...)" (Hahn, et al., 2013). Dies wird vorangetrieben durch das Verifizieren und Validieren der einzelnen Entwicklungsprodukte oder Entwicklungsteile, die durch die Staffelung zum Zielprodukt eine „V“- Form bekommen (vgl. Abb. 2).

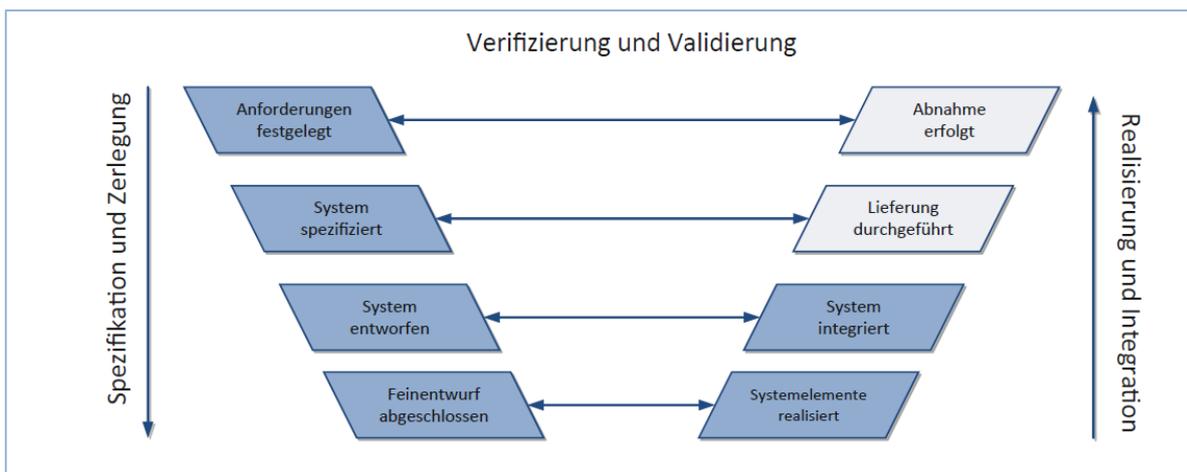


Abb. 2.: Struktur der Systemerstellung mit dem V-Modell XT 2009 (Hahn, et al., 2013)

Das V-Modell bietet Vorgehensbausteine, die an das konkrete Projekt angepasst werden. Dabei besteht es aus acht Bausteinen, die in Abb. 2 genannt sind. Sie können beliebig mit weiteren Zwischenstufen wie z. B. 'kaufmännisches Projektmanagement' oder 'Messung und Analyse' ergänzt werden können. Durch Messungen und Analysen von Projektkennzahlen wird das Projekt gesteuert und detaillierter definiert. Über die zusätzlichen Bausteine wird der Fokus des Organisators auf das entsprechende Projekt gelegt. Hier gilt es, die Detailausprägung des Projektes zu definieren, um an das Projektziel zu gelangen (Hahn, et al., 2013)

In Abb. 3 wird das V-Modell, wie es für die Softwareentwicklung des BPD umgesetzt wurde, dargestellt.

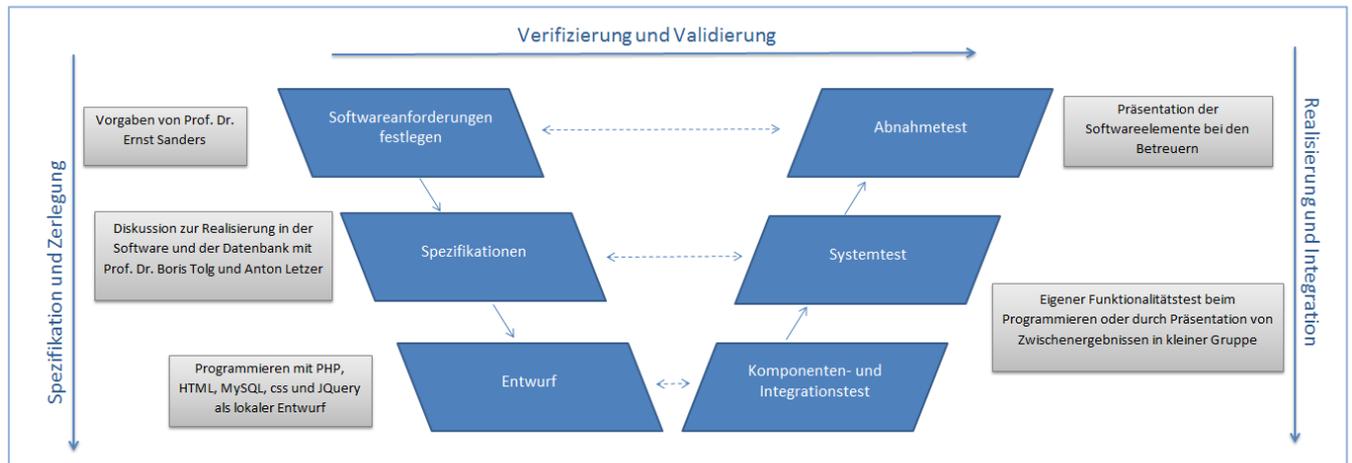


Abb. 3 V-Modell der Softwareentwicklung des BPD

Zunächst wurden die Softwarefunktionen und -anforderungen festgelegt.

Vertieft wurden diese durch die Spezifikationen, die durch die Realisierung beim Programmieren entstehen. Anschließend erfolgte ein Entwurf des Softwareteils.

Schon während der Programmierung wurden alle Komponenten auf ihre Funktionalität geprüft, bis am Ende der funktionale Verbund als Systems getestet wurde.

## 2.3 Sonstige verwendete Komponenten zur Entwicklung der Software

### 2.3.1 HTML-Die Grundlage des Programms

HTML ist eine Auszeichnungssprache, mit der das Aussehen von Dokumenten bestimmt werden kann. Grundlegende Idee ist es, auf vielen unterschiedlichen Anzeigegeräten ein Dokument entsprechend seiner Bedeutung durch eine logische Auszeichnung wiedergegeben zu können (Laborenz, 2006).

Mit HTML ist es möglich, die Syntax und die Anordnung von verankerten Anweisungen zur Darstellung des Dokumentinhaltes mit Text, Bildern und anderen Hilfsmedien zu organisieren. Mit ihr können über Hyperlinks Dokumente verbunden werden. Diese Funktion kann durch andere Programmiersprachen erweitert werden, z. B. durch PHP. (Musciano, et al., 1999)

HTML-Dokumente bestehen aus einer Ansammlung von ineinander geschachtelten Elementen, die in einer hierarchischen Struktur angeordnet und dargestellt werden. Es ist unterteilt in die Elemente HEAD und BODY, die jeweils weitere Elemente enthalten. HTML stellt das Fundament für den BioProcessDesigner.

### 2.3.2 PHP als Grundlagenerweiterung

PHP ist eine weitverbreitete Programmiersprache zur Entwicklung dynamischer Internetseiten. PHP steht für PHP Hypertext Pre-processor und wird in HTML eingebunden.

Dynamisch bedeutet, dass die Vorgänge und Verwendungen von Datentypen vorrangig zur Laufzeit eines Programms durchgeführt werden. Zudem können die Typen der Variablen wechseln. So kann eine Variable vom Typ *String* zu *Integer* wechseln. (Bergmann, 2005)

Eine dynamische Internetseite macht es im Vergleich zur statischen Internetseite möglich, direkt auf Aktionen des Benutzers reagieren zu können. Es unterstützt die einfache Auswertung von Formularen und die Zusammenarbeit mit verschiedenen Datenbanksystemen. (Thies, 2010)

PHP ist objektorientiert. Es beinhaltet PHP- Klassen, die dazu dienen, Daten und Operationen zu einer Einheit zusammenzufassen. Eine Klasse dient dabei als Bauplan zur Erzeugung eines Objektes während der Laufzeit des Programms. (Bergmann, 2005)

Durch PHP wird die Seite des BioProcessDesigners dynamisch, was z. B. beim Login oder der Registrierung genutzt wird.

### 2.3.3 CSS

CSS steht für Cascading Stylesheets. Sie koordinieren möglichst genau die Kontrolle über das Aussehen einer Webseite. Die Idee dahinter war, den HTML-Code einer Webseite komplett von Formatierungsbefehlen zu befreien und ihn in eine getrennte Datei, das Stylesheet, auszulagern. Das Stylesheet ähnelt insofern der Formatvorlage in Microsoft Word oder Open Office. Diese regelt z. B. die Schriftgröße und die Schriftart. (Laborenz, 2006)

Mit CSS wurde das Layout des BioProcessDesigners gestaltet.

### 2.3.4 MySQL

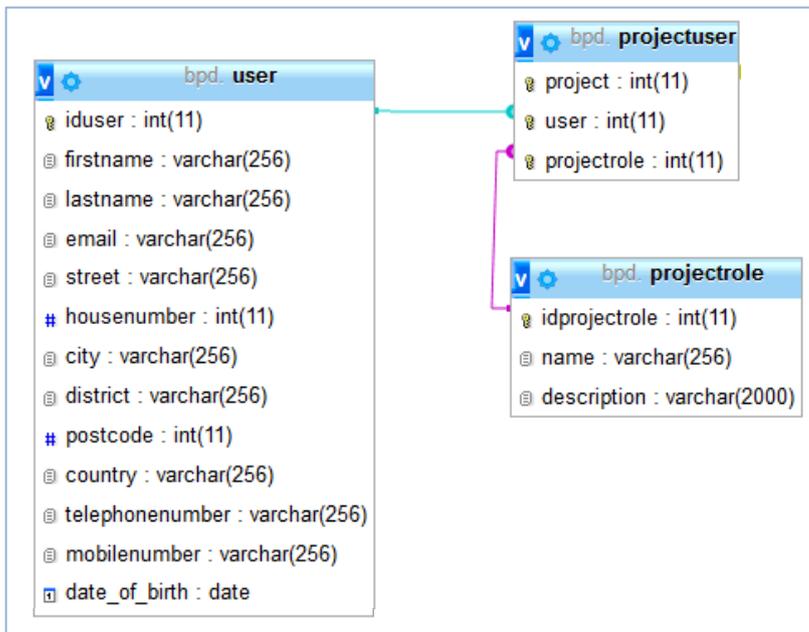
MySQL ist eine weit verbreitete Open Source Datenbank. Sie wird oft zusammen mit PHP verwendet. Über einen SQL- basierten Datenbankserver bietet es die Möglichkeit, Strukturen von Datenbanken und Tabellen zu erzeugen oder Datensätzen anzulegen, anzuzeigen, zu ändern oder zu löschen. PHP stellt hierbei die komfortable Schnittstelle zum Bearbeiten der Datensätzen aus der MySQL-Datenbank. (Thies, 2010)

Die Datenbank im Hintergrund vom BioProcessDesigner ist eine MySQL Datenbank.

Durch die Verwendung einer Datenbank entsteht Redundanzfreiheit, die Daten sind logisch unabhängig, man hat eine hohe Flexibilität, sie kann von mehreren Benutzern genutzt werden und es wird automatisch ein Zugriffsschutz für den bearbeiteten Datensatz eingerichtet.

### 2.3.4.1 Einführung Entity Relationship Modell

Im Verlauf dieser Arbeit wird des Öfteren mit Entity Relationship Modellen gearbeitet, um die Zusammenhänge in der Datenbank darzustellen. In *Abb. 4* ist ein Teil des Entity Relationship Modells des BPD dargestellt.



**Abb. 4** Teilmodell zur Erläuterung eines Entity Relationship Modells

Jeder Kasten stellt eine Tabelle der Datenbank *bpd* dar. Diese steht für eine Entity, auf Deutsch ein Objekt. Der Name der jeweiligen Tabelle steht im Kopf des Kastens. Er wird im weiteren Verlauf **fettgedruckt und in dieser Schriftart** gekennzeichnet. Die darunter angegebenen Elemente sind die Spalten der Tabelle, sie werden im weiteren Verlauf grau und

in dieser Schriftart gekennzeichnet.

Hinter dem Doppelpunkt ist der Typ der Variable angegeben. Er legt fest, ob der Inhalt der Spalte z. B. als Text (varchar) oder als Zahl (int) interpretiert werden soll. Die Zahl in Klammern hinter dem Typ legt die maximale Zeichenlänge des Eintrags fest. So hat beispielsweise die Tabelle **projectuser** (rechts oben in *Abb. 4*) der Datenbank *bpd* die Spalten **project**, **user** und **projectrole**. Alle drei Spalten sind vom Typ Zahl und haben eine maximale Zeichenlänge von 11 Zeichen (vgl. *Abb. 4*).

<b>bpd. projectuser</b>		
<i>project</i>	<i>user</i>	<i>projectrole</i>
bpd project 1	Max Muster	project member
bpd project 2	Michael Mustermann	project manager

**Abb. 5** Analoge Tabelle **projectuser** mit Beispielinträgen

*Die hier gezeigten Zeileninhalte entsprechen nicht den Inhalten der im BPD realisierten Datenbank, da sie den Variablentyp „int“ der Übersicht halber ignorieren.*

Diese Spalten benennen alle relevanten Merkmale, die eine Entity beschreiben. So wird z. B. ein **user** durch die Merkmale **firstname**, **lastname**, **email**, **street**, usw. beschrieben. Einige dieser Merkmale sind so prägnant, dass sie zur direkten Identifikation eines Objektes genutzt werden können. Diese Eigenschaften werden als **primary key**

in der Tabelle angelegt. Die Spalten haben einen kleinen Schlüssel als Aufzählungspunkt (  ) (vgl. *Abb. 4.*). In der Tabelle **user** ist der primary key die iduser.

Einige Einträge werden dazu genutzt, auf eine Vielzahl von Charakteristika zu verweisen. So verweist die Spalte **user** aus der Tabelle **projectuser** auf die **userid** der Tabelle **user**. Dies hat den Vorteil, dass nicht alle Charakteristika der **user** in der Tabelle **projectuser** mit aufgelistet werden müssen. Bei Bedarf kann in der Tabelle **user** nachgeschaut werden. Ein solcher Verweis nennt sich **foreign key**. Sie werden in *Abb. 4* durch die Verbindungslinie dargestellt.

Die in der Tabelle angegebenen Daten sind auch oft nicht wie in *Abb. 5* die ausgeschriebenen Bezeichnungen, sondern werden durch die Zahlenwerte der **ids** ersetzt, die diese Bezeichnung in der entsprechenden Tabelle hat. Dadurch enthalten manche Tabellen keinen Klartext mehr.

Die Tabelle **projectuser** ist eine **Koppeltabelle**. Sie verbindet Einträge aus der Tabelle **user** mit den Einträgen der Tabelle **project** und **projectrole**. Hier wird festgelegt, welcher **user** in welchem **project** mit welchen Rechten beteiligt ist.



## Project

---



Project XY

Der Verbund aus einzelnen Labormitarbeitern zeigt eine **project group** in einem **project**. Diese besteht aus mindestens zwei Personen.

Es besteht die Möglichkeit, dass eine Person an mehreren Projekten gleichzeitig beteiligt ist.

## Head of Laboratory- Laborleitung

---



Bei ihm liegt die Verantwortung für die Mitarbeiter. Er koordiniert seine Mitarbeiter und besitzt komplette Weisungsbefugnis.

Anzumerken ist an dieser Stelle, dass diese Hierarchieebene nur der Vollständigkeit halber aufgelistet wird, sie wurde in der Software nicht realisiert.

## Project Manager

---



Eine **project group** wird durch den Projektleiter delegiert. Er ist weisungsbefugt für seine Projektmitarbeiter, die **project member**. Er stellt seine **project group** aus dem vorhandenen Laborpersonal zusammen. Beim Erstellen der **project group** mit der Software bestimmt er

Projekteigenschaften, die ein **project guest** sehen kann.

Auf Softwareebene darf er Prozesse seines Projektes editieren. Das *Editieren* beinhaltet dabei das Anlegen, Bearbeiten und löschen von Daten.

Er kann Prozesse, die außerhalb des Projektes erstellt wurden, importieren und einpflegen. Diese Prozesse oder Prozessschritte können von anderen Nutzern jeder Hierarchiestufe stammen.

### Attribute:



## Project Member

---



Ist dem [project manager](#) und dem 'Head of Laboratory' unterstellt. Er erhält Anweisungen von beiden und muss diese ausführen.

Innerhalb der Projekte darf er Daten eingeben und seine eigenen Daten überschreiben oder löschen. Er darf diese Daten drucken, speichern oder plotten (grafische Weiterverarbeitung der Rohdaten).

Er führt alle Arbeiten aus, die in einem Prozess anfallen, und kann Prozesse starten oder beenden.

### Attribute:



## Project Guest

---



Ist stiller Beobachter des Geschehens eines Projekts. Sein Blick beschränkt sich auf einzelne Projektteile, die durch den [project manager](#) definiert wurden. Der [project guest](#) kann zu einem späteren Zeitpunkt in die bisher beschriebene Hierarchie eingegliedert werden.

### Attribute:



## Überlappender Projektteilnehmer

---



Diese Person arbeitet sowohl an [project A](#) also auch an [project B](#). In [project A](#) ist er in diesem Beispiel [project manager](#) in [project B](#) hingegen [project member](#). Dadurch entstehen unterschiedliche Befugnisse und Aufgaben für jedes Projekt.

Die Softwarehierarchie wird in der Datenbank über die **userclass** realisiert, im Folgenden in grün und durch diese Schriftart hervorgehoben. Sie gliedert sich wie folgt (vgl. Abb. 7):

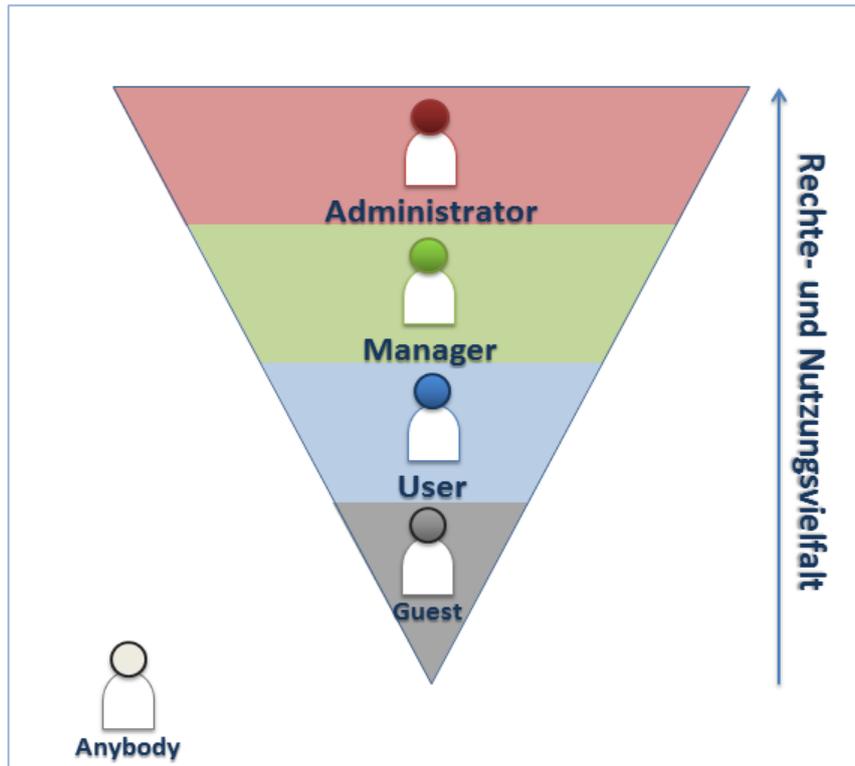


Abb. 7 userclass- Softwarehierarchie der BPD

Die Nutzer der **userclasses** haben folgende Rechte- und Nutzungsräume:

### Administrator



Hat absolute Handlungsfreiheiten, deren Grenzen allein durch die Software gesetzt sind. Der **administrator** darf in jedem Bereich wie z. B. project, user, Equipment oder Prozesse frei editieren. Der Administrator darf als Einziger Daten jeder Art löschen.

Er kann einen neu registrierten Nutzer einer **userclass** zu weisen, ihn sogar als einen weiteren Administrator zulassen.

#### Attribute:



## Manager

---

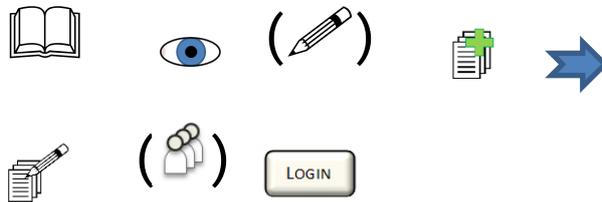


Darf Projekte erstellen und legt den **project manager** des Projekts fest, dabei muss er selber nicht Teil des Projekts sein.

Er darf ähnlich wie der **administrator** die neu registrierten Nutzer bis auf eine Hierarchiestufe unter sich – also bis auf **user** Ebene- anheben. Er kann also keine weiteren **manager** zulassen.

Der **manager** darf neue Elemente wie Equipment einpflegen und editieren, jedoch im beschränkten Rahmen, der zu diesem Zeitpunkt noch nicht genau festgelegt ist.

### Attribute:



## User

---



Darf seine eigenen Daten bearbeiten. Er kann außerhalb von Projekten seine eigenen Prozessabläufe erstellen und editieren um Möglichkeiten von Parametereinstellungen o. ä. auszuprobieren und dadurch die Software kennen zu lernen. Diese Prozesse oder Prozessfragmente können zu späteren Zeitpunkten durch den **project manager** in einen Prozessablauf importiert werden.

### Attribute:



## Guest

---



Nachdem sich ein Nutzer registriert hat, kann ihm der **administrator** oder **manager** zulassen und ihm das Einzuloggen freischalten. An diesem Punkt hat er den Nutzerstatus **guest** erlangt.

Als **guest** kann der Nutzer das Menü sehen. Er kann nicht sehen, welche Projekte es gibt.

Durch einen **project manager** kann er als **project guest** zugelassen werden, jedoch kein **project member** werden.

**Attribute:**



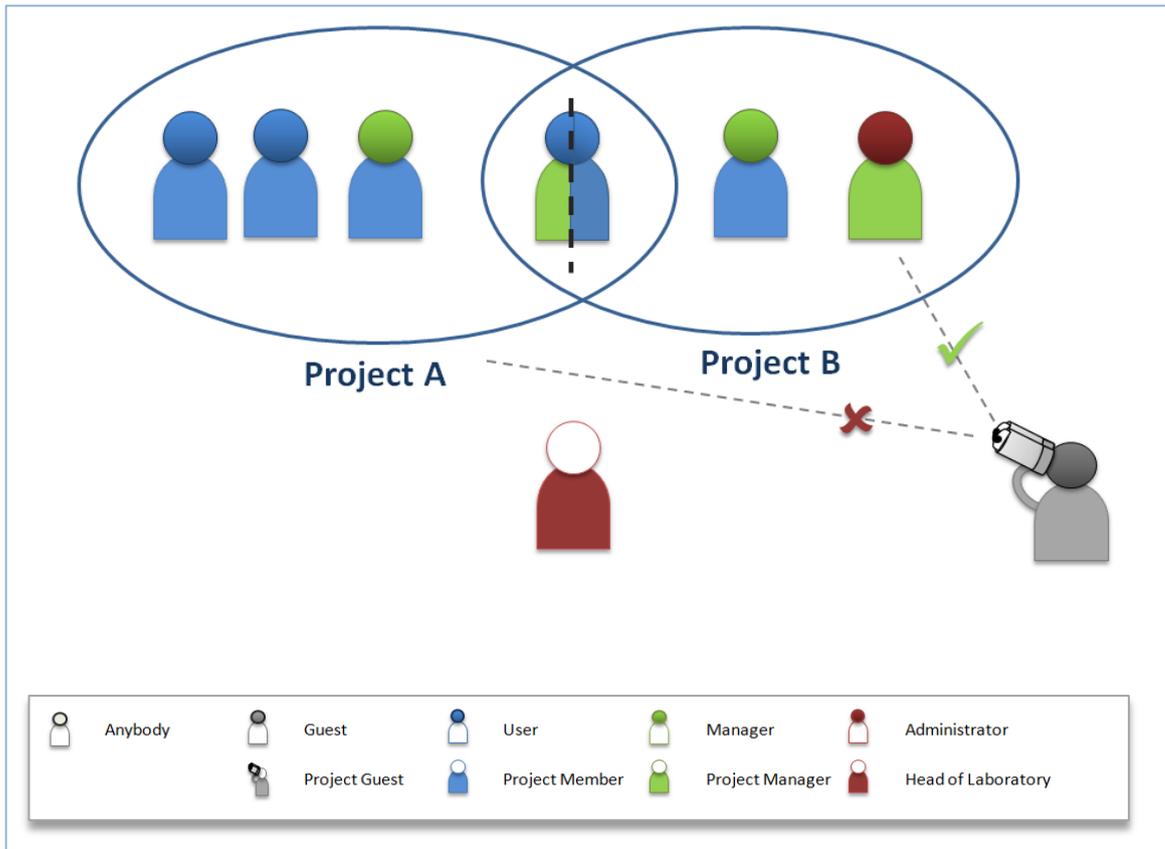
## Anybody

---



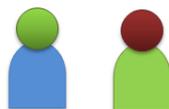
Direkt nach der Registrierung bekommt ein Nutzer diese **userclass**-Status. Der Nutzer ist gezwungen, auf das Anheben der Hierarchiestufe durch den **administrator** oder **manager** zu warten, da er sich in dieser Stufe nicht einloggen kann.

**Attribute:** -



**Abb. 8** Überlappung der Software- und der Laborhierarchie

In der Software überlappen sich die bisher beschriebenen Hierarchien teilweise. Dadurch können Situationen entstehen, die in der folgenden Darstellung in *Abb. 8* veranschaulicht werden.



Der linke Nutzer ist in Bezug auf die Softwarenutzung ein **manager** in **project B** ist er ein **project member**

Der rechte Nutzer ist in Bezug auf die Softwarenutzung ein **administrator** und für das **project B** der **project manager**.



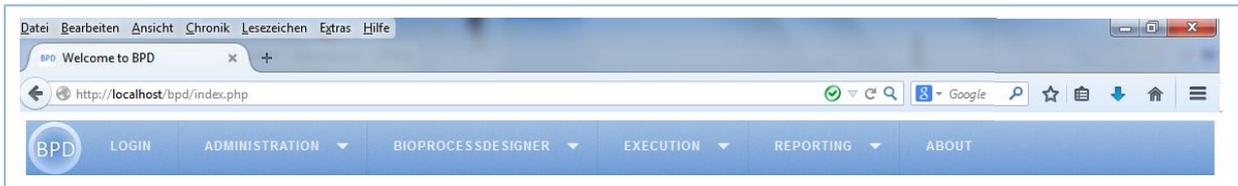
Dieser Nutzer ist sowohl an **project A** also auch an **project B** beteiligt. In der Softwarehierarchie ist er ein **user**. In **project A** übernimmt er die Rolle des **project managers**, in **project B** hingegen ist er ein **project member**. Ein **user** hat oft weniger Erfahrung im Laboralltag als ein **manager**. Daher soll ein Prozessablauf vor Prozessstart in einer Situation wie dieser von einem **manager** bestätigt werden müssen.

### 3.2 Das Menü- Navigation durch den BPD

Das Menü des BPD ist unterteilt in die Punkte LOGIN, ADMINISTRATION, BIOPROCESSDESIGNER, EXECUTION, REPORTING und ABOUT( vgl. Abb. 9).<sup>4</sup>

Die Breite des Menüs passt sich automatisch der Fensterbreite an, es wurde *responsive* gestaltet.

Im Weiteren werden sie durch die Verwendung von KAPITÄLCHEN hervorgehoben.



**Abb. 9.** Das Menüband als Ausschnitt aus der Arbeitsoberfläche des BPD.

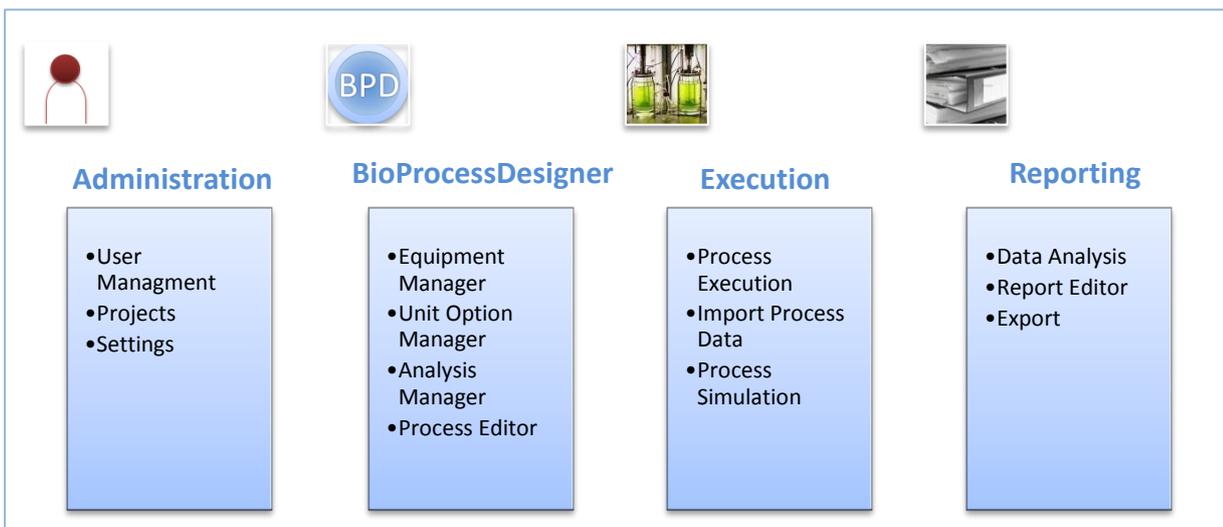
Das Menü baut auf die Struktur in Abb. 10 auf.

Es wurde für die Software um die Punkte LOGIN und ABOUT ergänzt, um die Oberflächen den üblichen Bedienungsanforderungen einer Software anzupassen.

Da im Weiteren wird nicht auf den Unterpunkt ABOUT, HOME und IMPRESSUM eingegangen wird, folgt hier eine kurze Erläuterung. Unter dem Menüpunkt werden die Project Manager der Softwareentwicklung des BPD mit ihren Kontaktdaten aufgeführt, sowie eine Aufzählung der Developer.

Die HOME Seite ist durch Klicken auf das Logo des BPD zu erreichen, sie ist gleichzeitig die Startseite beim Aufrufen der Software.

Das IMPRESSUM der Seite ist zu jedem Zeitpunkt am linken Rand der Softwareoberfläche zu finden.



**Abb. 10** Aktuelle geplante Aufbau Struktur des BPD (Bildquelle: (Decker), (Zoo15))

<sup>4</sup> Abgebildet wird die Software hier mit dem Mozilla Firefox 34.0.5(x86 de) Browser 1287 x398 Pixel

Die Bezeichnungen der Menüpunkte und deren Unterpunkte sind derzeit schon festgelegt, die meisten aber noch nicht detailliert geplant. Daher kann im Folgenden nur eine grobe Skizze der späteren Gestaltung und deren Funktionen gegeben werden.

Viele Inhalte der geplanten Funktionen werden hier in Anlehnung an das Studienprojekt von Peter Maul beschrieben (Maul, 20.12.2013).

### 3.2.1 Login und Registrierung der Nutzer



Abb. 11. Login Oberfläche für den Nutzer

Wie bereits erwähnt, muss sich ein Nutzer zunächst registrieren.

Dies kann er, indem er im Menüpunkt LOGIN auf den Link zur [registration](#) klickt (vgl. Abb. 11).

Anschließend leitet die Software auf die Seite mit dem Registrierungsformular in

Abb. 12. Für die Registrierung muss der neue Nutzer mindestens seinen Vor- und Nachnamen, seine E-Mail- Adresse und sein Geburtsdatum zusammen mit einem Passwort und der Passwortwiederholung angeben. Über den Vor- und Nachnamen sowie das Geburtsdatum sollen die Nutzer im Notfall auch außerhalb des Laborbetriebes eindeutig identifizierbar sein. Eine gültige E-Mail- Adresse ermöglicht das Kontaktieren des Nutzers.

Die oben genannten Angaben sind im Formular mit einem (\*) markiert und müssen eingegeben, um sich erfolgreich zu registrieren.

Abb. 12. Oberfläche bei der Registrierung neuer Nutzer

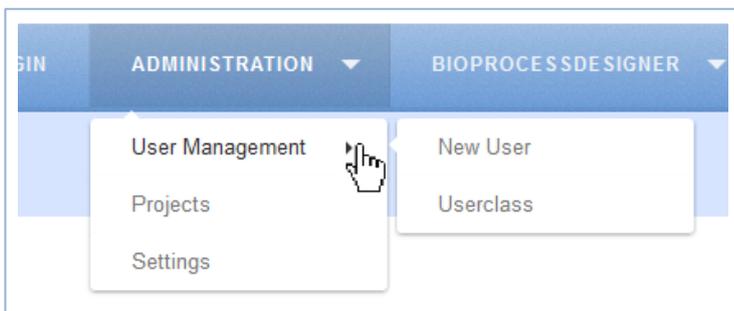
Ein gültiges Passwort muss den Passwortanforderungen genügen, diese werden dem Nutzer beim Wischen mit dem Mauszeiger über ⓘ angezeigt. Es muss mindestens acht Zeichen, eine Ziffer, einen Groß- und einen Kleinbuchstaben und ein Sonderzeichen enthalten.

Der Nutzer wird mit allen eingetragenen Daten in die Datenbank eingelesen. Beim Registrieren wird geprüft, ob die eingegebene E-Mail-Adresse schon in der Datenbank enthalten ist.

Für den LOGIN muss der Nutzer seine E-Mail-Adresse mit dem dazugehörigen Passwort eingeben. Nachdem der Nutzer sich erfolgreich eingeloggt hat, wechselt der Menüpunkt LOGIN zu LOGOUT. Der Nutzer kann sich jetzt, innerhalb des Handlungsraums seiner **userclass**, durch die Software bewegen.

Beim Klicken auf den Menüpunkt LOGOUT ist der Nutzer wieder abgemeldet.

### 3.2.2 Administration



**Abb. 13.** Unterpunkte des Menüpunktes ADMINISTRATION

Der Menü Punkt ADMINISTRATION enthält die Menü Unterpunkte USER MANAGEMENT, PROJECTS und SETTINGS (vgl. Abb. 13).

Obwohl die Unterpunkte des USER MANAGEMENT jederzeit angezeigt werden, sind sie nur für Nutzer mit dem **userclass** Status

**manager** oder **administrator** zugänglich. Versucht ein **user** auf die Seite des USER MANagements zu kommen, wird er automatisch auf die HOME Seite weitergeleitet. Über NEW USER können Nutzer der **userclass anybody** bestätigt werden. Über USERCLASS kann der **userclass** Status der Nutzer geändert werden. Nähere Beschreibung der Realisierung dieser Funktionen dieser Menüteile in Kapitel 3.5.3 - Neue Nutzer zulassen - Seite 39 und Kapitel 3.5.4 - Userclass eines Nutzers verändern - Seite 41.

PROJECTS soll dem Nutzer eine Übersicht über alle bisher in der Datenbank vorhandenen Projekte geben. Von hier aus hat der Nutzer die Möglichkeit, seine Projekte zu verwalten und zu erreichen.

Über SETTINGS sollen Formatierungen oder Grundeinstellungen der Seite einstellbar sein.

### 3.2.3 BioProcessDesigner



Abb. 14 Unterpunkte des Menüpunktes  
BIOPROCESSDESIGNER

Dieser Menü Punkt enthält die Unterpunkte EQUIPMENT MANAGER, UNIT OPERATION MANAGER, ANALYSIS MANAGER und PROCESS EDITOR (vgl. Abb. 14). Er beinhaltet das Herzstück der Planung der späteren Prozesse. Über den Unterpunkt EQUIPMENT MANAGER gelangt der Nutzer zur Verwaltung des Equipments. Er wird in

Kapitel 3.3 - Der Equipment Manager – Die graphische Oberfläche Seite 25 näher erläutert.

Im Menüunterpunkt des UNIT OPERATION MANAGERS können Tätigkeiten in Prozessen bis in die kleinste unabhängige Tätigkeit definiert werden. Sie können anschließend zu 'Process steps' zusammengefasst werden, um im PROCESS EDITOR verwendet zu werden.

Im ANALYSIS MANAGER werden die Analysenverfahren verwaltet. Sie können hier erstellt und bearbeitet werden.

Mit dem PROCESS EDITOR kann der Ablauf des Prozess geplant und gespeichert werden. Hier wird die Abfolge der Prozessschritte definiert und das darin verwendete Equipment eingeplant.

### 3.2.4 Execution



Abb. 15 Unterpunkte des Menüpunktes  
EXECUTION

EXECUTION beinhaltet die Menü Punkt PROCESS EXECUTION, IMPORT PROCESS DATA und PROCESS SIMULATION (vgl. Abb. 15). Über den PROCESS EXECUTION kann ein vorher geplanter Prozess ausgeführt und gestartet werden. Hier werden die gespeichert Prozesse aus dem PROCESS EDITOR des Menüpunktes BIOPROCESSDESIGNER genutzt.

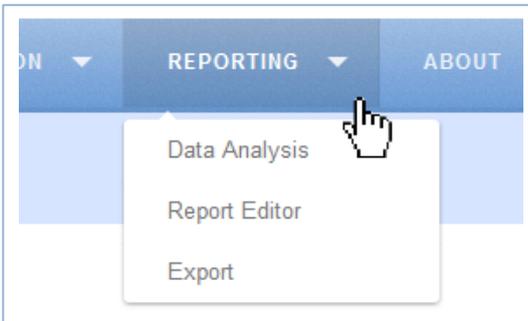
Während des Prozesses können Notizen zum Verlauf gemacht werden. Der BPD soll es ermöglichen, jedes handgeschriebene Laborjournal zu ersetzen.

Zu gegebenem Anlass soll es möglich sein, Parameter des Prozess während des Ablaufes zu ändern, ohne den Prozess neu starten zu müssen.

Nach der Prozess Erstellung kann der Ablauf des Prozesses über PROCESS SIMULATION graphisch simuliert werden.

Über IMPORT PROCESS DATA können extern gespeicherte Prozess Daten in die Software importiert werden und weiterverarbeitet werden.

### 3.2.5 Reporting



**Abb. 16** Unterpunkte des Menüpunktes REPORTING

REPORTING beinhaltet die Menü Punkt DATA ANALYSIS, REPORT EDITOR und EXPORT (vgl. Abb. 16). Ist ein Prozess beendet, können die Daten über den Menüpunkt DATA ANALYSIS den Auswertungsanforderungen angepasst werden. Die Funktionen werden dazu genutzt, aussagekräftige Fazits aus dem Prozess ziehen zu können, zum Beispiel die Bestimmung des

Umrechnungskoeffizienten von der optischen Dichte zur Biotrockenmasse oder die maximale spezifische Wachstumsrate  $\mu_{max}$ . Hierfür können Standardformeln genutzt werden oder neue hinzugefügt werden. Der Nutzer lädt hier die Prozessdaten, markiert den Teil der Prozessdaten, die weiterverwendet werden sollen, und wählt eine vorhandene oder neue Methode aus. Das Ausgabeformat kann eine Tabelle oder ein Diagramm sein.

Im REPORT EDITOR kann Inhalt und Layout des Berichts angepasst werden. Hier können mehrere Daten aus dem DATA ANALYSIS kombiniert aufgeführt und gespeichert werden. Es besteht dabei die Möglichkeit, den Report nach angelegten Vorlagen, sogenannten Report Templates, zu gestalten.

Über EXPORT können die Daten der Prozesse in andere Dateiformate exportiert werden, um sie für andere Programme zugänglich zu machen. Hier können auch die erstellten Berichte aus dem REPORT EDITOR gedruckt oder beispielsweise als PDF-Datei gespeichert werden.

Ein Teil zu diesem Softwarebereich wurde bereits von Jan Bautsch in (Runge, et al., 2014) programmiert.

### 3.3 Der Equipment Manager – Die graphische Oberfläche

Der Equipment Manager soll dafür genutzt werden, neues Equipment einzupflegen (*add*), vorhandenes Equipment zu bearbeiten (*modify*) oder zu löschen (*delete*).

Dabei unterteilt sich das Equipment in Kategorien entsprechend ihres Nutzungsbereiches, ihrer Heterogenität und ihres Typs in *group*, *category* und *subcategory*. In einer *subcategory* befinden sich die *items*. Im Weiteren werden diese Kategorisierungen in *dieser Farbe und Schriftart* hervorgehoben.

Es wurde versucht, durch einen gleichen Aufbau der Funktionen zum Modifizieren, Hinzufügen und Löschen, die allgemein als Editier-Option bezeichnet werden, eine intuitive Nutzung der Equipmentverwaltung zu ermöglichen. Eine detailliert Beschreibung der Oberfläche und der Nutzung der Oberfläche befindet sich in *Anhang A- Oberfläche des Equipment Managers - Seite 56*.

### 3.4 Datenbankdesign - Koordination von Nutzern und Equipment im virtuellen Raum

Das entworfene Datenbankmodell enthält Tabellen zur Verwaltung der Nutzer, der Projekte und des Equipments. Die Verbindung und der Aufbau dieser Tabellen sollen im Weiteren näher erläutert werden. Der Arbeit liegt ein Entity Relationship Modell der gesamten Datenbank bei.

#### 3.4.1 Das Datenbank Design der User

Die bisher erläuterten Strukturen wurden wie folgt im Datenbankmodell umgesetzt:

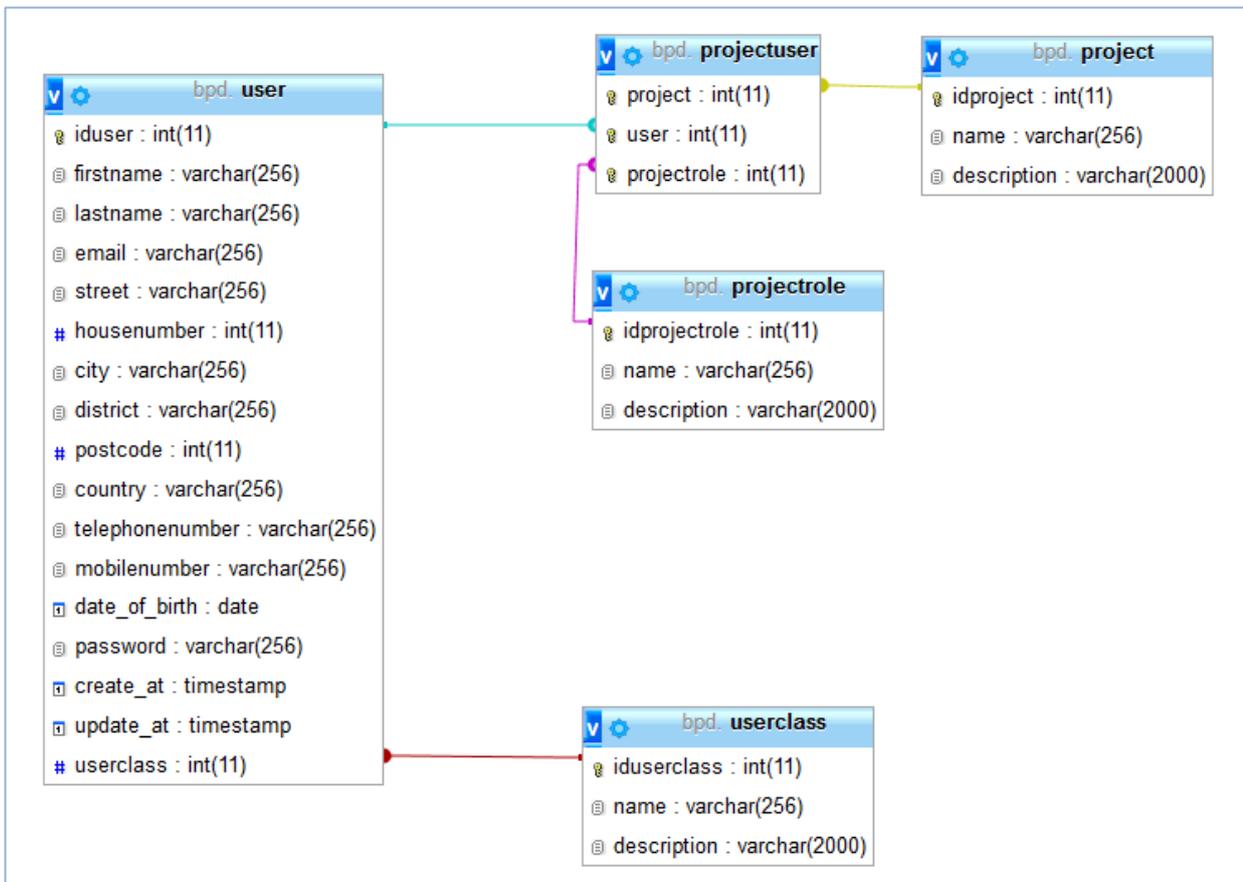


Abb. 17 Entity Relationship Model der Datenbankstruktur der **userclass** und **projectrole**

#### User

Die Tabelle **user** (in Abb. 17 links) besteht aus den Charakteristika iduser, firstname, lastname, email, street, housenumber, city, district, postcode, country, telephonenumber, mobilenummer, date of birth, password, create\_at, updated\_at und userclass.

Bis auf iduser, create\_at, update\_at und userclass kommen alle Informationen aus dem Registrierungsformular der Anmeldung (vgl. Kapitel 3.2.1, Seite 21).

In create\_at wird der Zeitpunkt der Registrierung gespeichert. In update\_at wird der Zeitpunkt der letzten Änderung der Daten gespeichert. Zum Zeitpunkt der Registrierung ist dieser gleich dem Zeitpunkt von create\_at. Geplant ist, dass, wenn der Nutzer oder ein Administrator eine Einstel-

lung bei dem Nutzer verändert, der `update_at` mit dem Aktualisierungszeitpunkt überschrieben wird.

Die `iduser` des **users** wird automatisch beim Erstellen des Eintrags als Integer von der Datenbank generiert. Wird ein Nutzer gelöscht, wird seine `iduser` nicht überschrieben, es wird von der gelöschten `iduser` weiter hoch gezählt. Wie erwähnt, ist die `iduser` der primary key der **users** Tabelle. Er repräsentiert den Nutzer in den Tabellen **projectrole** und der **project** und kann seinen Nutzerdaten zugewiesen werden. Es wurde sich für einen primary key auf der `iduser` entschieden und nicht auf der E-Mail-Adresse, um es den Nutzern zu ermöglichen, ihre E-Mail-Adresse zu ändern.

### Userclass

Die Logik der **userclass** wurde ausführlich in *Kapitel 3.1 - Nutzer- und Rechteverwaltung - Seite 13* erläutert, in *Abb. 17* befindet sich die Tabelle der Datenbank unten rechts.

In der Datenbank charakterisiert sich die Tabelle der **userclass** aus den Einträgen `iduserclass`, `name` und `description`.

Die Umsetzung in der Datenbank erfolgt über die Zuordnung der `iduserclass` zu den oben erläuterten Hierarchieebenen.

**Tabelle 1** Zugehörigkeit der **userclass** Bezeichnung zu der **iduserclass**

name der <b>Userclass</b>	<code>iduserclass</code> in der Datenbank
Administrator	1
Manager	2
User	3
Guest	4

In der Spalte `description` kann ein kurzer beschreibender Text festgehalten werden, der den Handlungsbereich aus *Kapitel 3.1 Nutzer- und Rechteverwaltung* beschreibt.

Die Tabelle **userclass** ist über die Spalte `iduserclass` mit der Spalte `userclass` aus der Tabelle **user** über einen Foreign key verbunden. Dies ist in *Abb. 17* als rote Linie zu sehen. Somit ist an dieser Stelle einsehbar, was der Zahleneintrag der `userclass` aus der Tabelle **user** bedeutet.

### Project

**Project** (in *Abb. 17* oben rechts) besteht aus den Spalten `idproject`, `name` und `description`.

Die `idproject` wird, wie bei der Tabelle **user**, automatisch beim Erstellen des Eintrags generiert.

In der Spalte `name` wird der Name des Projektes eingetragen und kann mit einer kurzen Beschreibung in der Spalte `description` ergänzt werden.

Inhalte dieser Tabelle werden später einmal die Übersicht der Projekte im Menü Unterpunkt PROJECTS der ADMINISTRATION füllen.

### Projectrole

Die Logik der **projectrole** wurde schon in *Kapitel 3.1- Nutzer- und Rechteverwaltung - Seite 13* erläutert.

Die **projectrole** Tabelle setzt sich aus den Spalten `idprojectrole`, `name` und `description` zusammen.

Analog zur **userclass** (vgl. *Seite 26*) lässt sich hier folgende Zuordnungstabelle erstellen:

Tabelle 2 Zugehörigkeit der **projectrole** Bezeichnung zu der `idprojectrole`

name der <b>projectrole</b>	<code>idprojectrole</code> in der Datenbank
Project Manager	1
Project Member	2
Project Guest	3

Auch hier kann in die Spalte `description` dazu genutzt werden, eine kurze Beschreibung der einzelnen Hierarchieebenen kurz zu erläutern.

### Projectuser

Der **projectuser** (*Abb. 17* oben Mitte) besteht aus den Spalten `project user` und `projectrole`. In dieser Tabelle werden die Einträge der Tabelle **project** mit den Einträgen der Tabelle **user** in Form einer Koppeltabelle verbunden. Hier wird festgehalten, welcher Nutzer (über seine `iduser`) in welchem Projekt (über die `idproject`) mit welchen Rechten (über die `idprojectrole`) arbeitet. Eine Verdeutlichung ist in *Abb. 18* zu finden. Der Nutzer hat zwei Einträge für jedes Projekt in dem er arbeitet.

Die Verbindungen der Tabellen ist durch die foreign keys die in *Abb. 17* in den Farben türkis, gelb und pink dargestellt. Es kann es zu jedem Nutzer mehrere Einträge geben.

Hier ein Beispiel:

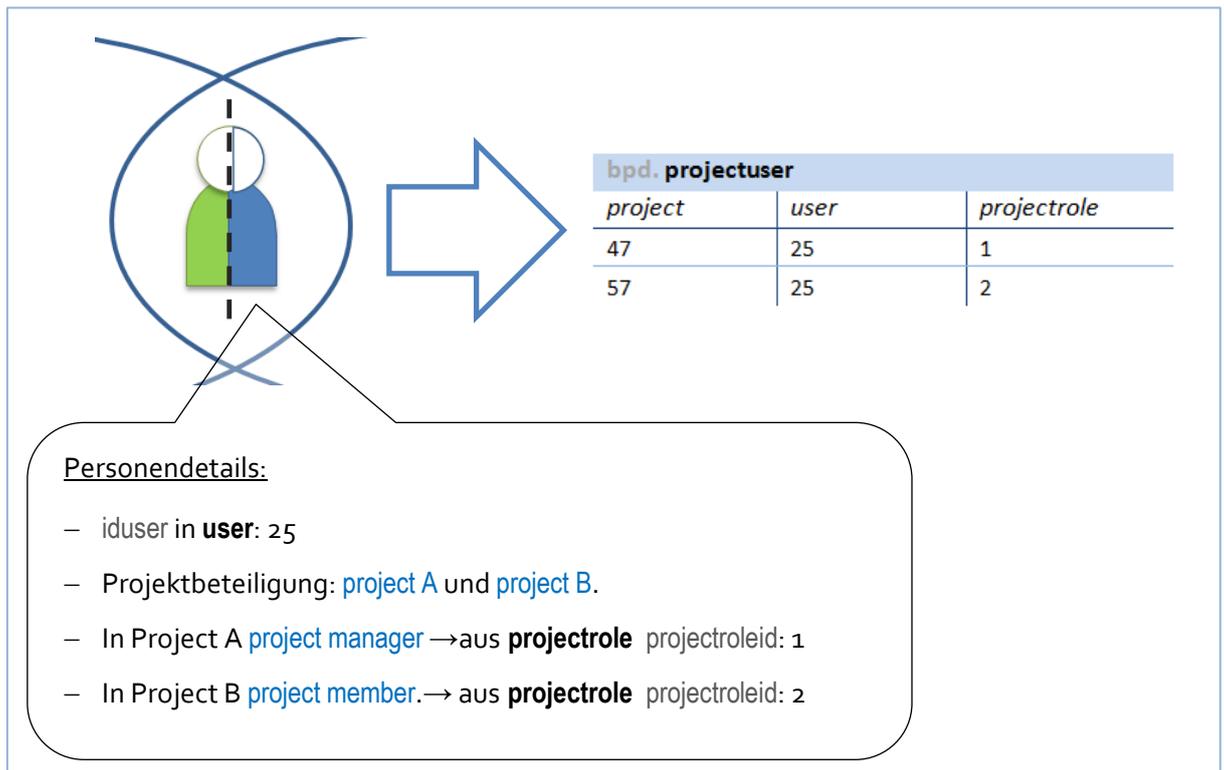


Abb. 18 Beispielseintrag in der Tabelle **projectuser** der Datenbank BPD

### 3.4.2 Das Datenbank Design des Equipment Managers

Für den Equipment Manager essenziell ist die Datenbanktabelle **collection**. Zusammen mit den Tabellen **basedata** und **collectioncontent** bieten sie dem Nutzer der Software eine große Vielfalt im Umgang mit dem Equipment.

Die **collection** Tabelle besteht aus den Spalten idcollection, name, parent und description (vgl. Abb.

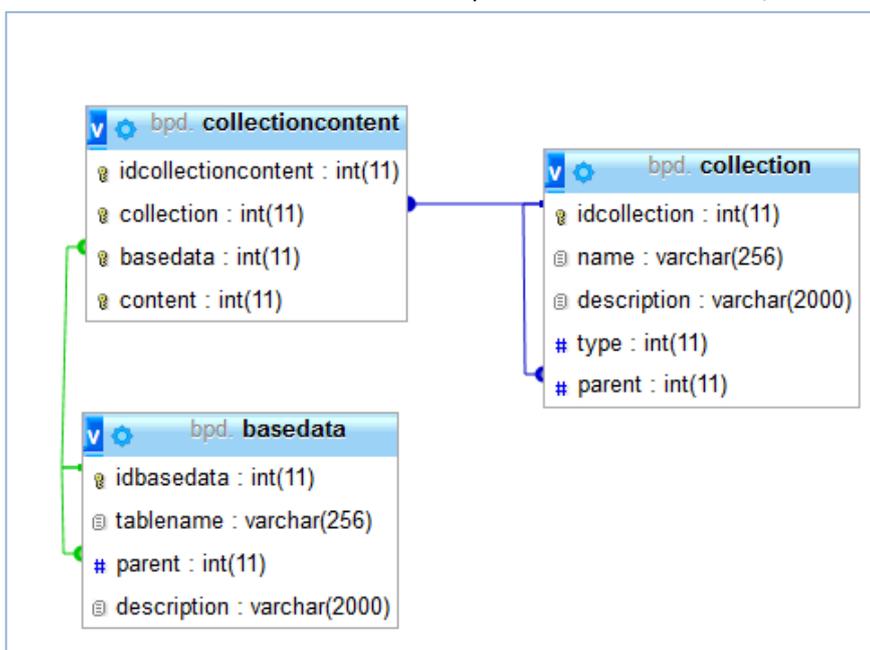


Abb. 19 **collection** Tabelle der Datenbank BPD im Entity Relationship Modell

19).

Die idcollection weist jedem neuen Eintrag automatisch eine fortlaufende Nummer zu. Die idcollection gelöschter Elemente wird nicht neu vergeben, es wird von der gelöschten idcollection weiter hochgezählt. Die Tabelle **basedata**

besteht aus der idbasedata, tablename, parent und description. Die Tabelle **collectioncontent** aus idcollectioncontent, collection, basedata und content.

Die **collection** Tabelle dient als ein Verzeichnis der Kategorisierung. Die Kategorisierung entsteht durch die Nutzungsanforderungen des Equipments, eine Übersicht befindet sich in *Abb. 22 - Seite 33*. Die **groups** unterscheiden sich stark voneinander. Bsp.: Chemicals/ Biologicals und Documents. Innerhalb der **group** ist unter den **categories** noch eine deutliche Heterogenität zu erkennen, z. B.: **group**: Devices **categories**: Centrifuge und Pump. Die **subcategory** listet die Typen der **category** auf, z. B.: **Category**: Centrifuge; **subcategories**: Cooling Centrifuge, Desk Centrifuge, Decanter. Diese Kategorisierung lässt sich auch in der graphischen Oberfläche wiederfinden, vgl. *Kapitel 3.3 - Der Equipment Manager – Die graphische Oberfläche – Seite 25*.

Die Kategorisierung wird in der Datenbank über die Spalte des parent umgesetzt. Die Namen der **groups** und der **categories** werden in der description eingetragen. Jeder Eintrag hat eine idcollection. Bei den **categories** wird in der Spalte des parent die idcollection der Zugehörigkeit **group** eingetragen. Der parent verweist auf die **group** der **category**. Die Namen der **subcategory**, die in *Abb. 22 - Seite 33* angegebenen sind, stehen nicht bei description, sondern in der Spalte name.

Die Spalte des parent zeigt auf die Tabelleneigene Spalte idcollection, dieser foreign key stellt eine **Rekursion** dar, abgebildet in *Abb. 19* durch die grüne Linie und in *Abb. 20* durch die Pfeile. Die Rekursion stellt einen sachlogischen Zusammenhang zwischen einer Entity und einem Entity-Typen dar.

Das Beispiel in *Abb. 20* zeigt: die **group** Devices hat die idcollection 4. Die **category** Centrifuge wird

bpd. collection			
idcollection	name	parent	description
2		NULL	Disposables
3		NULL	Documents
4		NULL	Devices
...	...	...	...
17		4	Centrifuge
18		4	Columns
19		2	Filter
20		3	Report
...	...	...	...
24	Cooling Centrifuge	17	.....
25	Capsule	19	.....
26	HIC	18	.....

**Abb. 20** Veranschaulichung der Beziehungen in der **basedata** Tabelle und der dazugehörigen Rekursion

über den Eintrag der 4 in die Spalte parent der **group** Devices zu geordnet. Die **category** Centrifuge bekommt die idcollection 17. Das gleiche Prinzip lässt sich auch bei der **subcategory** Cooling Centrifuge wiederfinden. Über die 17 in der Spalte parent wird sie der **category** Cooling Centrifuge zugeordnet. Die **subcategory** Cooling Centrifuge bekommt die idcollection 24.

Um die Eigenschaften einer **subcategory** festgelegt und die Einträge der **items** zu generieren, werden Einträge in die Tabelle **collectioncontent** gemacht. Diese Tabelle ist eine Koppeltabelle für die Einträge aus den Tabellen der **collection**, **basedata** und Spalten der Tabellen auf die die **basedata** Tabelle verweist.

Die **basedata** Tabelle wird als Tabellenverzeichnis genutzt. Sie dokumentiert alle Tabellen, die angelegt werden, um die Eigenschaften von Equipment zu beschreiben. Die beschreibenden Tabellen können genau wie die Tabelle der **collection** kategorisiert werden. Die **basedata** Tabelle enthält auch eine Rekursion, die über die Spalten idbasedata und parent dargestellt wird. Bisher

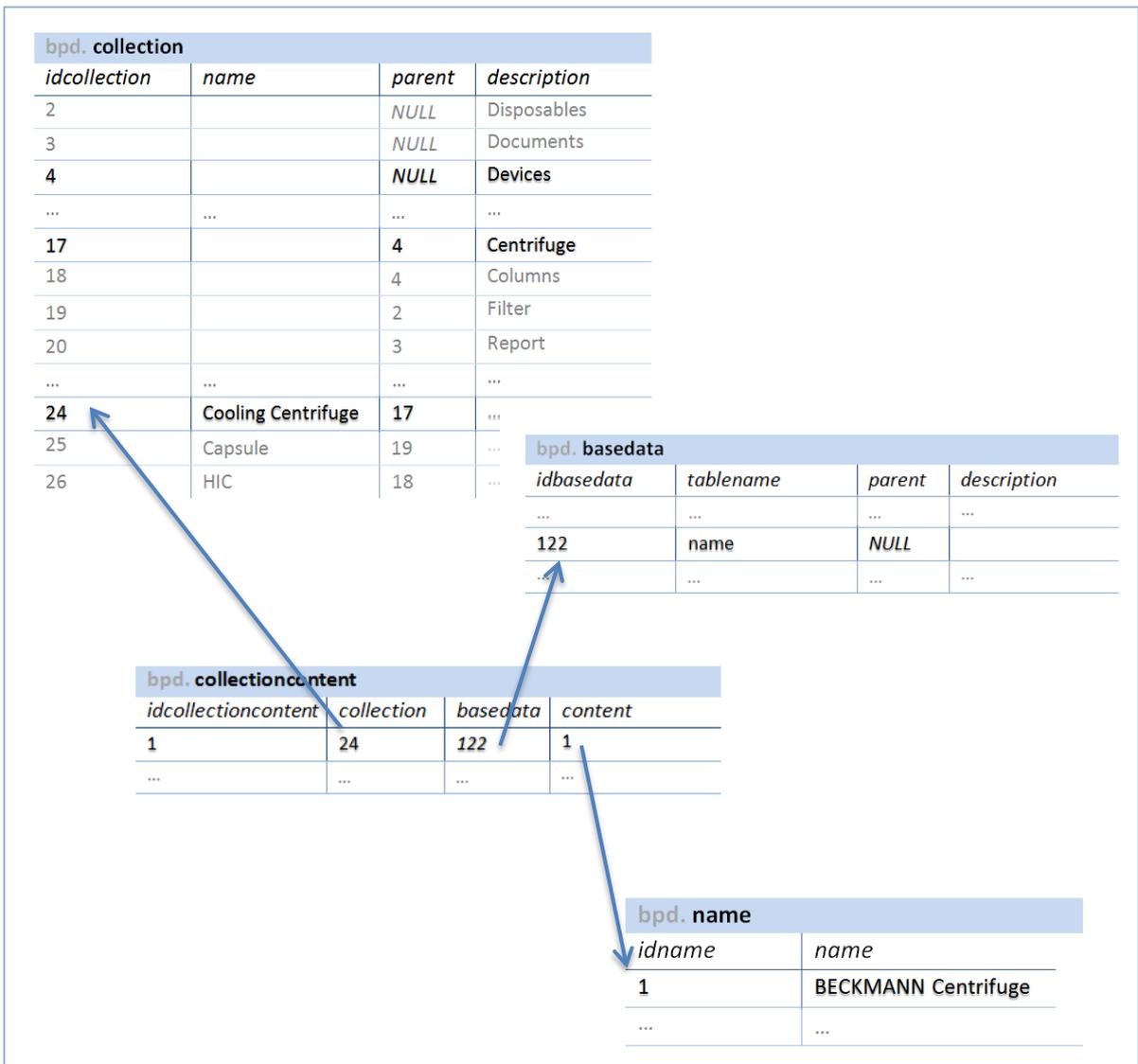


Abb. 21 Beispiel Verdeutlichung der Funktion der Tabelle **collectioncontent**

sind hier noch keine weiteren Kategorisierungen festgelegt worden.

Das Beispiel von oben in *Abb. 21* sollen nun um den **collectioncontent** erweitert werden. Das **item**, dass sich der **subcategory** **Cooling Centrifuge** zuordnen lässt, hat einen Namen: **BECKMANN Centrifuge**. Dieser steht in der Tabelle **name**. Der Eintrag hat die **idname** **1**. Der Name der Tabelle wird in die **basedata** Tabelle eingetragen und ist dort mit der **idbasedata** **122** zu finden. Sie besitzt keinen **parent**. Den Namen dem **item** zu zuordnen passiert wie erwähnt über die Tabelle **collectioncontent**. Die **24** zeigt auf die **idcollection** der **subcategory** **Cooling Centrifuge**. Die **122** zeigt auf die oben erwähnte Tabelle **name**. Die **1** aus der Spalte **content** verweist auf die **idname** der Tabelle **name**, die den Eintrag **BECKMANN Centrifuge** zuweist.

Durch die Koppeltabellen und die Rekursionen der Datenbanktabellen lässt sich die Equipmentunterteilung und -beschreibung uneingeschränkt erweitern.

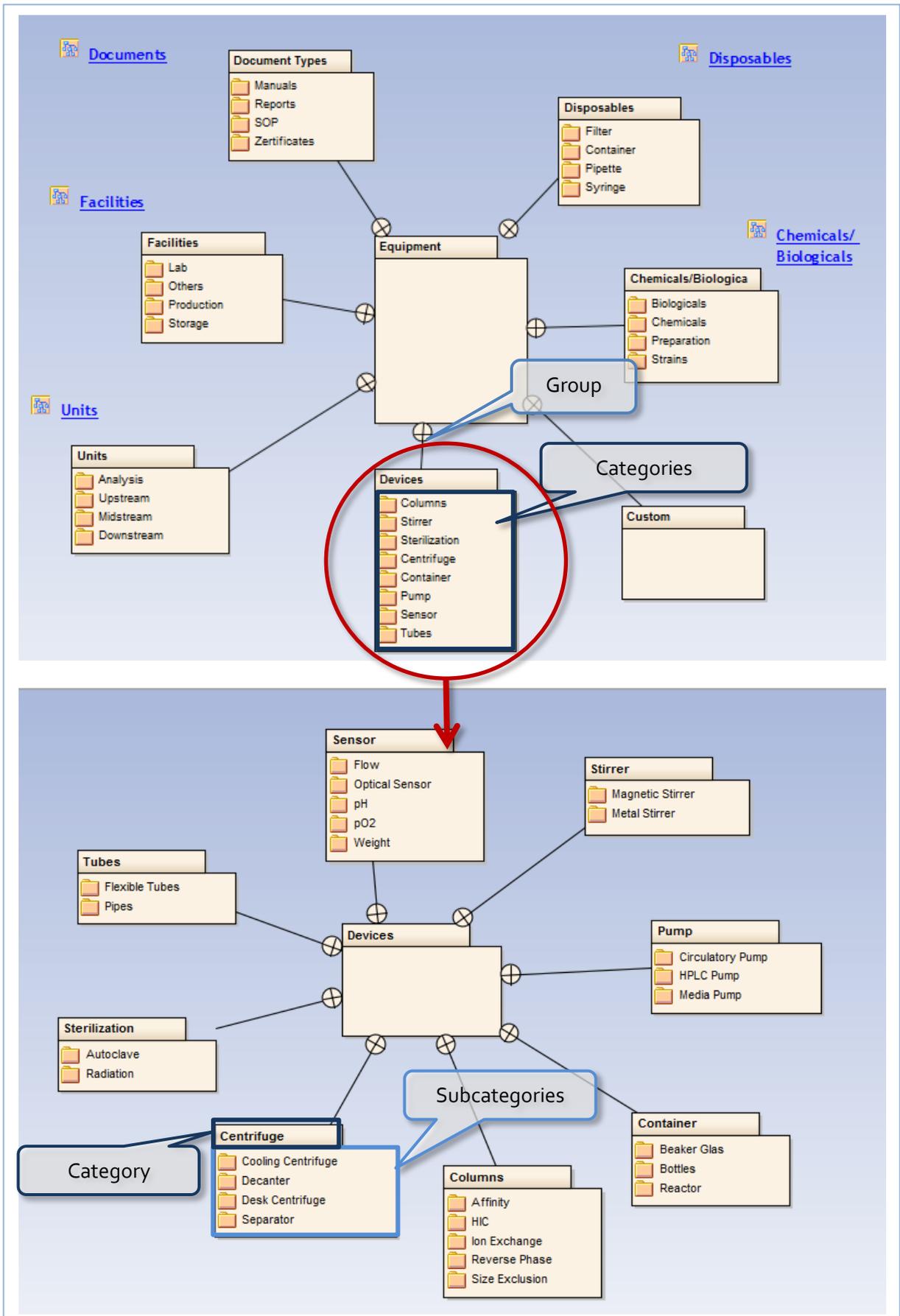


Abb. 22 Equipmентаufteilung (Maul, 20.12.2013)

## 3.5 Funktionaler Verbund – Kommunikation zwischen graphischer Oberfläche und der Datenbank

Die Softwareoberfläche sendet aufgrund der Eingaben des Nutzers Daten an die Datenbank, teilweise werden Informationen aus der Datenbank angefordert und im Hintergrund weiterverarbeitet. Aus diesen verarbeiteten Daten ergibt sich die Reaktion der Software auf die Eingabe des Nutzers. Die Abfragen und die Kommunikationen im Hintergrund sollen in diesem Kapitel näher erläutert werden. Der Quellcode liegt in *Anhang B- Programmcode- Seite 64* auf CD bei.

### 3.5.1 Login

Zum Login-Vorgang gibt es in *Abb. 23 - Seite 35* einen Verlauf des Entscheidungsdiagramms.

Zum Einloggen registrierter Nutzer werden die Textfelder des Loginformulars über POST-Variablen ausgelesen. Sind die Felder leer, wird der Nutzer über die Nachricht "Please fill in alle details" dazu aufgefordert, seine Daten einzugeben. Wurden die Daten des Nutzers eingegeben, werden die Details des Nutzers aus der Datenbank gesucht. Als Suchkriterium wird die E-Mail-Adresse genutzt. Gibt es einen Eintrag in der Datenbank zu der eingegebenen E-Mail-Adresse, wird überprüft, ob der Nutzer mindestens den **userclass** Status **guest** hat. Ist der Nutzer auf dem Status **anybody**, kann er sich nicht einloggen und bekommt die Nachricht: "You are not allowed to login". Hat er den Status **guest** oder höher, werden die Einträge zu seiner E-Mail-Adresse in das Array `$row` gespeichert und können hier abgerufen werden.

Das Passwort des Nutzers liegt chiffriert in der Datenbank vor. Dieses wird aus dem Array `$row` in die Variable `$dbpassword` gespeichert und über die `password_verify`-Funktion von PHP mit der Eingabe des Nutzers verglichen. Diese Funktion prüft, ob das chiffrierte (durch einen einmaligen Algorithmus verschlüsselte Passwort) zum Passwort des Nutzers passt. Die Gültigkeit wird über den hash und den salt geprüft (PHP-Dokumentationsgruppe, 2015). Details zum Vorgang der Passwortverschlüsselung siehe *Kapitel 3.5.2- Registrierung - Seite 36*.

Passen Passwort und Eingabe zusammen, wird die E-Mail und die **userclass** in das Array `$_SESSION` eingelesen. Dies ermöglicht ab diesem Zeitpunkt bis zum Zeitpunkt des LOGOUT das Prüfen der **userclass** und somit die Prüfung auf den Zugriff der Software. Beim LOGOUT wird das Array über die PHP-Funktion `session_destroy()` gelöscht.

Passen die Eingaben nicht mit den Angaben aus der Datenbank, bekommt der Nutzer die Nachricht „Sorry the user does not exist or password failure“. Anschließend wird die Seite neu geladen und der Nutzer kann sich wieder versuchen einzuloggen oder zu registrieren.

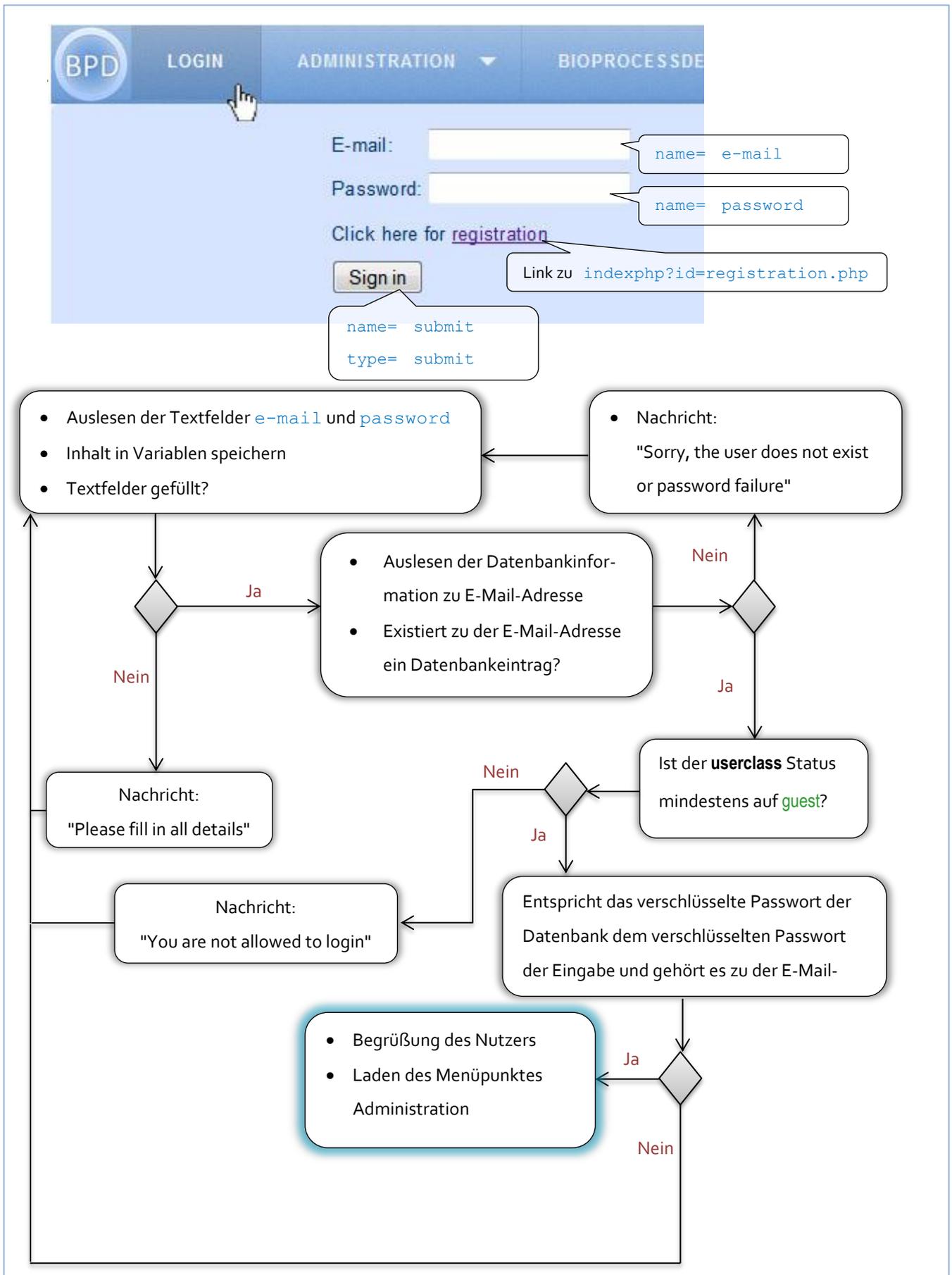


Abb. 23 Entscheidungsdiagramm zum Loginvorgang

### 3.5.2 Registrierung

Zum Registrierungs-Vorgang gibt es in *Abb. 24 - Seite 38* einen Verlauf des Entscheidungsdiagramms.

Die Felder aus dem Registrierungsformular werden mit dem Klicken des Submit- Buttons über POST- Variablen ausgelesen. Dabei wird das Geburtsdatum an das Datumsformat in der Datenbank angepasst (von DD-MM-YYYY zu YYYY-MM-DD). Mit der PHP Time Stamp- Funktion `date("Y-m-d H:i:s")` wird der aktuelle Zeitpunkt für die Variablen `create_at` und `update_at` erzeugt. Diese Funktion speichert das aktuelle Datum im folgenden Format:

Jahr (vierstellig)- Monat (mit füllender Null 01 bis 12)- Tag (mit füllender Null 01 bis 31)  
Stunde (im 24Stundenformat mit führender Null 00 bis 09):Minuten (mit führenden Nullen 00 bis 59) : Sekunden (mit führenden Nullen 00 bis 59)

Sind die Variablen der Pflichtfelder nicht gefüllt, wird eine Nachricht an den Nutzer ausgegeben. Sind die Pflichtfelder ausgefüllt, wird zunächst geprüft, ob das eingegeben Geburtsdatum mit einem Datumsformat übereinstimmt. Dafür wird geprüft, ob der Wert des Tages im Intervall [1, 31] liegt, der Wert des Monats im Intervall [1, 12] liegt und der Wert des Jahres im Intervall [1940,2000] liegt. Sind alle Bedingungen erfüllt, wird die `$dateOfBirthCheck`- Variable gleich 1 gesetzt.

Anschließend wird das Passwort auf seine Gültigkeit geprüft. Dafür wird zunächst das Passwort in einzelne Ziffern zerlegt. Über eine `for`- Schleife und mehrere `if`- Schleifen wird die Länge des Passwortes gezählt (soll  $\geq 8$ ), die Anzahl der Groß- und Kleinbuchstaben bestimmt, die Anzahl der Zahlen und die Anzahl der Sonderzeichen bestimmt (alle mindestens 1). Entsprechen sie den Anforderungen, wird die `$passwordcontrol1`- Variable gleich 1 gesetzt.

Als letztes wird die E-Mail-Adresse auf typische Eigenschaften geprüft. Zunächst wird geprüft, ob die Eingabe ein @-Zeichen enthält. Es wird die Länge betrachtet, ob eine Domain vorhanden ist, wo sie liegt, wie lang sie ist und ob sie Sonderzeichen enthält. Der Ort und die Anzahl von Punkten wird bestimmt und ausgewertet. Sind Sonderzeichen enthalten oder zu viele Punkte in der Eingabe, existiert kein @-Zeichen oder ist die Domain zu lange oder mit Sonderzeichen versehen, wird die E-Mail-Adresse als nicht gültig betrachtet, die `$regok` wird auf `false` gesetzt.

Eine als gültige beurteilte E-Mail-Adresse, wird in der Datenbank gesucht, um zu vermeiden, dass E-Mail-Adressen zweimal vorkommen. Die Anzahl der Suchergebnisse wird in die `$count`- Variable gespeichert (entweder 0 oder 1).

Entspricht das Geburtsdatum den Anforderungen eines Datums, das Passwort den Passwortkriterien, ist die E-Mail gültig und existiert noch keine Registrierung zu der angegeben

E-Mail-Adresse wird das Passwort über die PHP– Funktion `password_hash` chiffriert. Über diese Funktion wird das Passwort mit einem Algorithmus chiffriert, das Ergebnis ist nicht reproduzierbar und in der Datenbank für keinen als Klartext zu sehen.

Alle Einträge inklusive dem chiffrierten Passwort werden in die Datenbank eingelesen. Die **userclass** wird auf 5 gesetzt, was dem **userclass** Status `anybody` entspricht.

Der Nutzer bekommt alle seine eingegeben Daten noch einmal angezeigt, zusammen mit der Aussage, dass sein Registrierung erfolgreich war. Gleichzeitig wird eine E-Mail an den Administrator geschickt, dass es einen neuen Nutzer gibt, der bestätigt werden muss.

Ist das Geburtsdatum, das Passwort oder die E- Mail Adresse nicht in Ordnung oder existiert die E-Mail-Adresse schon in der Datenbank, wird dem Nutzer angezeigt, dass seine Angaben fehlerhaft sind oder der Nutzer schon registriert ist. Das Formular lädt neu für einen erneuten Versuch.

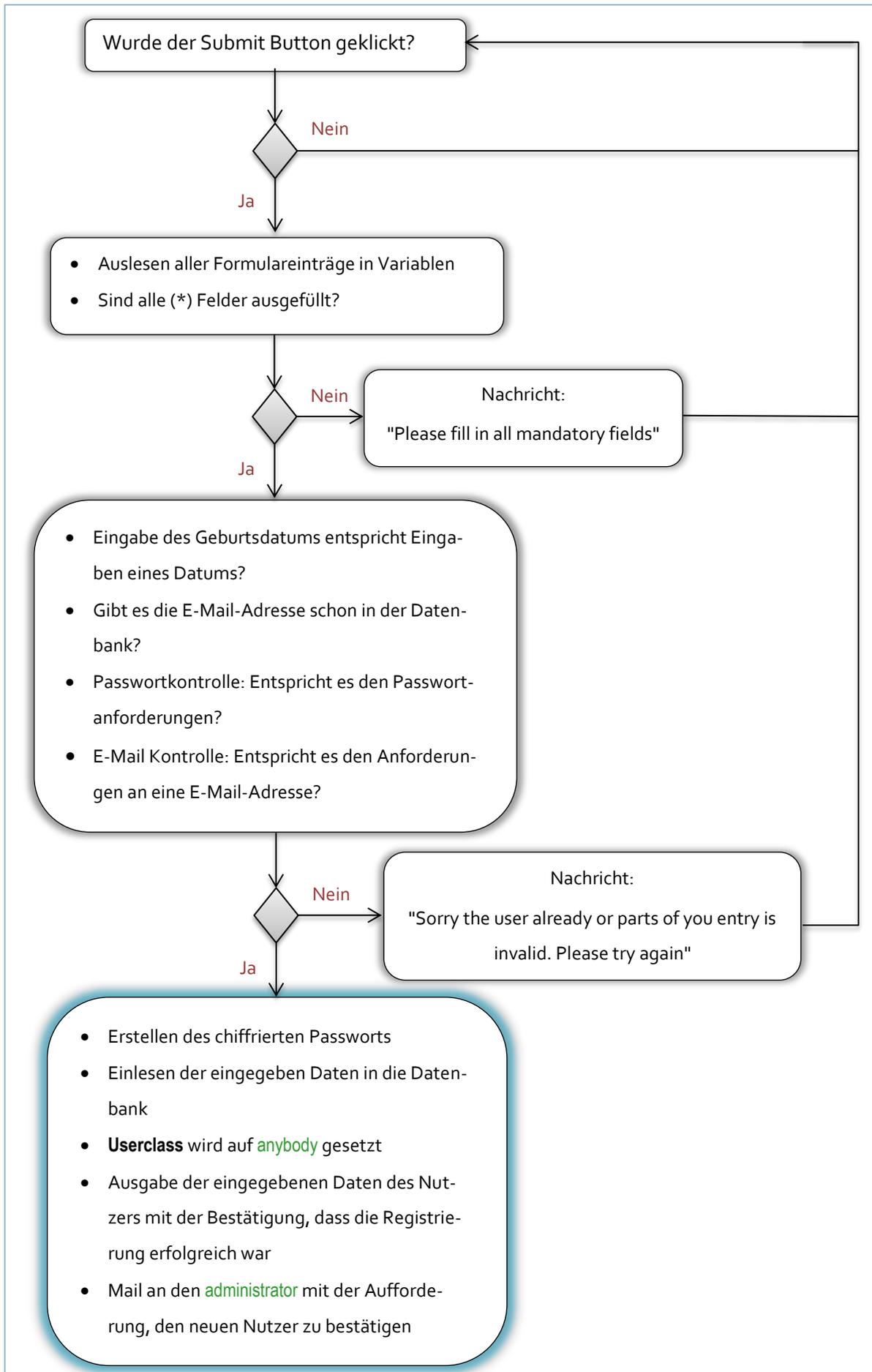


Abb. 24

Entscheidungsdiagramm bei der Registrierung neuer Nutzer

### 3.5.3 Neue Nutzer zulassen

Zur Nutzerzulassung gibt es in *Abb. 25- Seite 40* ein Entscheidungsdiagramm.

Beim Klicken auf den Menüpunkt NEW USER wird zunächst geprüft, ob der Nutzer eingeloggt ist und sein **userclass** Status gleich oder höher dem des **managers** ist.

Hat der Nutzer die Anforderungen nicht, wird er auf die HOME Seite weitergeleitet.

Ein **manager** oder **administrator** bekommt nun eine Tabelle mit den neu registrierten Nutzern angezeigt. Gibt es keine neuen Nutzer, bekommt er dies über eine Nachricht mitgeteilt. Die Wortwahl oberhalb der Tabelle mit der Information, wie viele neue Nutzer es gibt, passt sich der Anzahl der Nutzer und der Wortwahl im Singular oder Plural an.

Zum Ausgeben der Tabelle wird in der Datenbank nach allen Nutzern gesucht, deren **userclass** Status **anybody** ist, diese werden dann in ein Array eingelesen und über eine **while**- Schleife in der Tabellen mit den Informationen zum Zeitpunkt der Registrierung, dem Vor- und Nachnamen, der E-Mail-Adresse und dem Geburtstagdatum und einer Checkbox dargestellt. Der **value** der Checkbox entspricht der **iduser** des aufgelisteten Nutzers. Nach der Auswahl durch das Klicken der Checkboxen, können die Nutzer entweder über den **delete**-Button aus der Datenbank gelöscht werden oder über den **confirm**- Button den **userclass** Status **guest** bekommen. Die Auswertung der Checkboxen wird über eine **foreach** Schleife mit der **iduser** in dem Array **newuser[]** gespeichert und dann entsprechend der Buttonauswahl weiterverarbeitet. Wird die Checkbox zu einem Nutzer nicht ausgewählt, bleibt sein **userclass** Status unverändert.

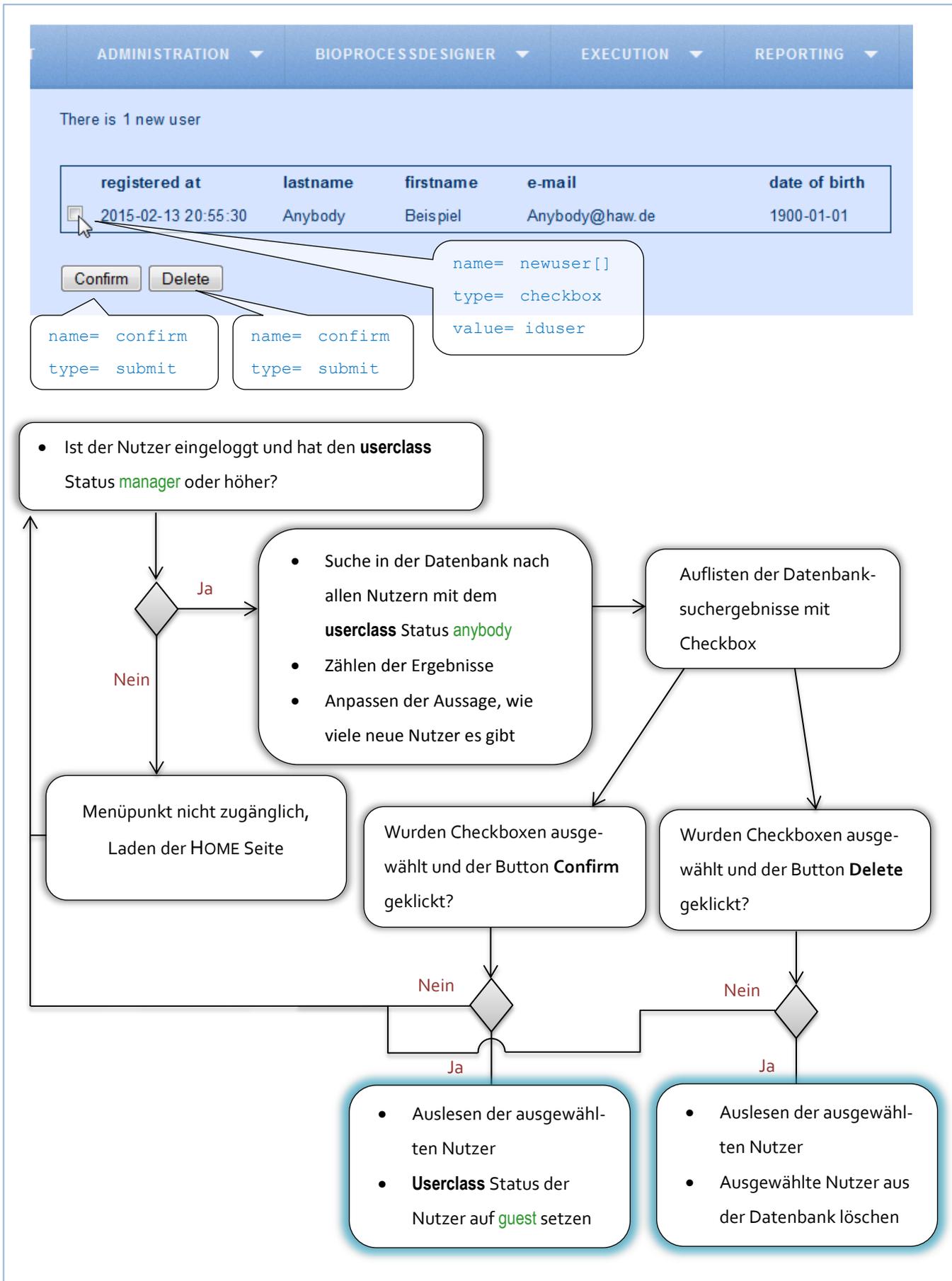


Abb. 25 Entscheidungsdiagramm zum Zulassen neuer Nutzer

### 3-5.4 Userclass eines Nutzers verändern

Zur Änderung des **userclass** Status gibt es in *Abb. 26* *Abb. 25, Seite 42* ein Entscheidungsdiagramm.

Beim Klicken auf den Menüpunkt NEW USER wird zunächst geprüft, ob der Nutzer eingeloggt ist und sein **userclass** Status gleich oder höher dem des **managers** ist.

Hat der Nutzer die Anforderungen nicht, wird er auf die HOME Seite weitergeleitet.

Ein **manager** oder **administrator** bekommt ein Dropdownmenü mit allen registrierten Nutzern angezeigt. Die Einträge enthalten den Vor- und Nachnamen sowie den **userclass** Status. Die Einträge des Dropdownmenüs werden über eine **while**- Schleife aus einem Array gelesen, welches die Ergebnisse einer Datenbankabfrage mit Auflistung aller Nutzer enthält. Die Radiobuttons darunter enthalten die **iduserclass** entsprechend des Status der in Klartext ausgeschriebenen **userclass** als **value**.

Wurde ein Nutzer ausgewählt, ein Radiobutton für den neuen **userclass** Status selektiert und der save-Button geklickt, wird geprüft, ob der aktive Nutzer den **userclass** Status **manager** hat. Will er einen Nutzer zum **manager** oder **administrator** machen, wird ihm mitgeteilt, dass er diesen Nutzer diese **userclass** nicht zuordnen kann. Entspricht der **userclass** Status des aktiven Nutzers einem **administrator** oder will ein **manager** einen **user** ernennen, wird die **userclass** des ausgewählten Nutzers auf die neue **userclass** gesetzt. Dafür wird die Auswahl des Dropdownmenüs und des Radiobuttons über POST-Variablen gespeichert und in der Datenbank aktualisiert.

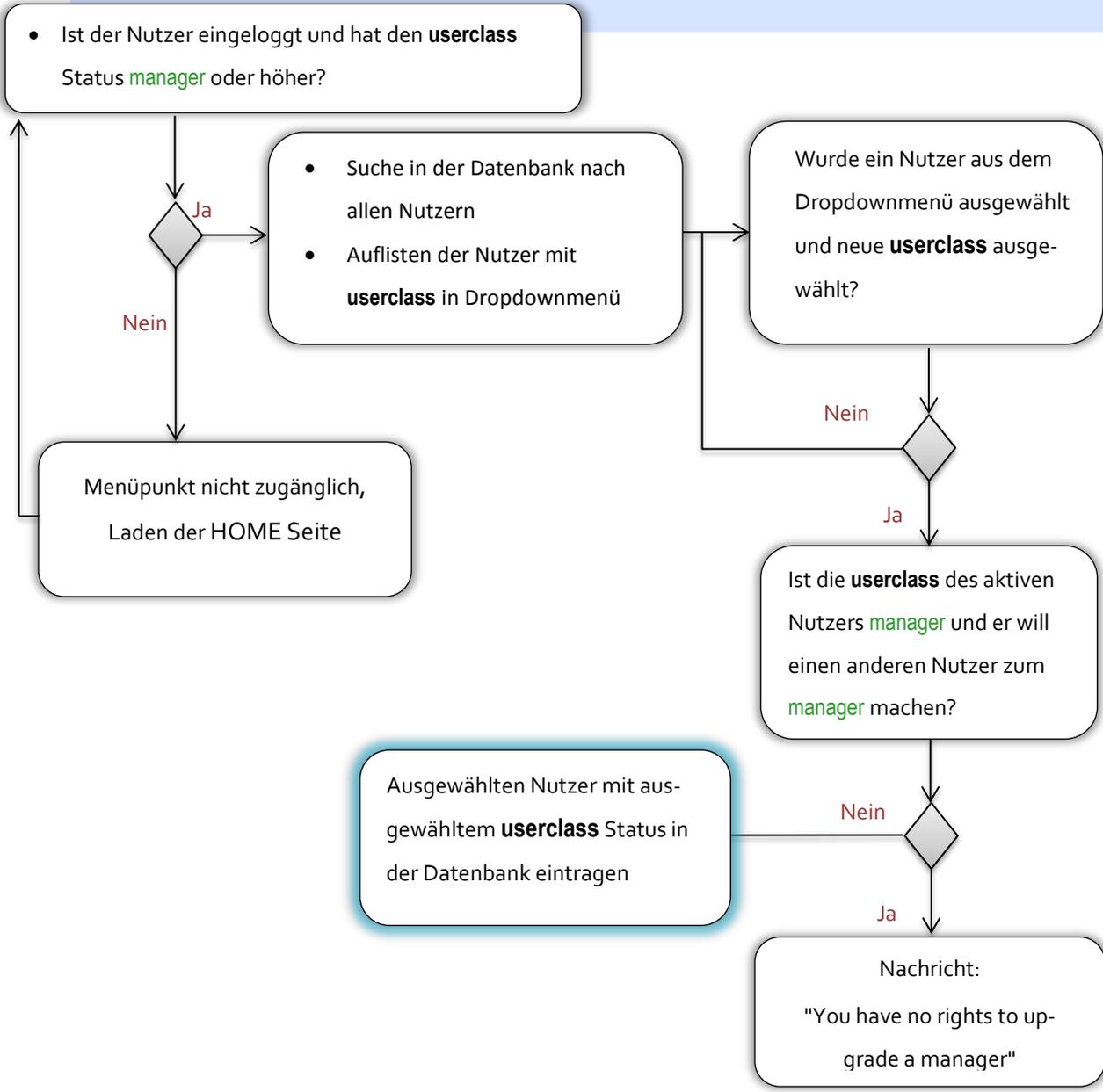


Abb. 26 Entscheidungsdiagramm zum Ändern der userclass Nutzer

### 3-5-5 Equipment Manager

Der Aufgabenbereich des Equipment Managers lässt sich in einzelne Funktionalitäten unterteilen. Erklärt werden hier nur die der `group`, die Funktionen der `category`, der `subcategory` und `items` weichen kaum von denen der `group` ab.

#### 3-5-5.1 Anzeigen der Kategorien

Zum Verlauf der Anzeige der Kategorien gibt es in *Abb. 27 - Seite 44* ein Entscheidungsdiagramm.

Durch die Kategorisierung des Equipments legt die Auswahl der `group` die Anzeige der `category` und diese die Anzeige der `subcategory` fest. Aus der Datenbank werden alle Einträge gesucht, die keinen `parent` haben, alphabetisch sortiert und in das Array `$group` gespeichert. Über eine `while`-Schleife wird der Inhalt des Arrays in einem Dropdownmenü dargestellt werden. Dieses wird über den HTML-Befehl `<select>` generiert. Jeder Eintrag hat einen `value`, der die `idcollection` der im Klartext angegeben `group` ab. In jedem Dropdownmenü des Equipment Managers entspricht der `value` des Eintrags der `idcollection` der `group`, `category` oder `subcategory`. Durch diese dynamische Gestaltung zeigt das Dropdownmenü jederzeit den aktuellen Stand der Datenbank und basiert nicht auf statischen `group`-Einträgen. Statische Bestandteile sind die Editieroptionen `Modify`, `Add` und `Delete` und die Aufforderung: "Please select ...".

Wird ein Eintrag ausgewählt, wird die JQuery Funktion `change (function)` aufgerufen. Sie ist die Selektierfunktion und speichert die `idcollection` der ausgewählten `group` in die Variable `$groupId`. Sobald eine `group` ausgewählt wurde, werden die dazugehörigen Einträge der `category` in einem weiteren Dropdownmenü dargestellt. Es werden alle Datenbankeinträge im Dropdownmenü angezeigt, die die `idcollection` der `group` als `parent` haben. Im Dropdownmenü der `category` ist festgelegt, dass bei einer Änderung der ausgewählte `value` des Eintrags an die Funktion `getSubCategories` geschickt wird (`onchange= "getSubCategories(this.value)"`). Auch diese Funktion ist eine Selektierfunktion und speichert die `idcollection` der `category` in die Variable `$catId`. Sobald diese Variable einen Wert zugewiesen bekommen hat, wird das Dropdownmenü der `subcategory` angezeigt. Es werden alle Datenbank Einträge im Dropdownmenü angezeigt, die die `idcollection` der `category` als `parent` haben. Das Senden des Ausgewählten Eintrags ist identisch mit dem Vergehen der `category`, nur dass die `idcollection` der `subcategory` in der Variable `$subcatId` gespeichert wird. Der beschriebene Vorgang wiederholt sich bei der Auswahl der `subcategory`.

Wird eine der Editier-Optionen ausgewählt, wird der `Value` der Auswahloption in die Variable `editGrAction/ editCatAction/ editSubcatAction` gespeichert. Die entsprechende Editier-Option wird neben dem Dropdownmenü dargestellt.

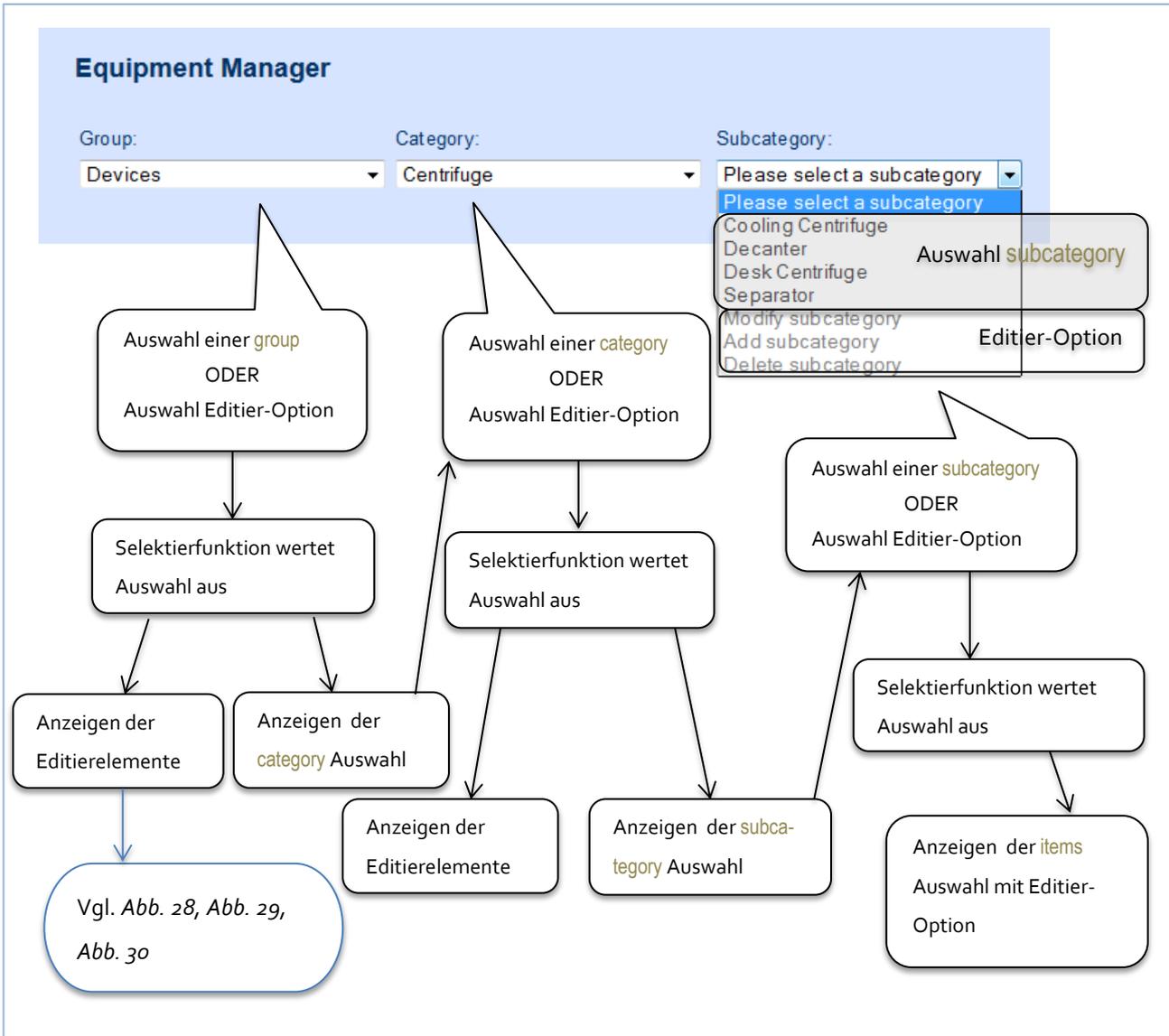


Abb. 27 Vorgehen zur Anzeige der Kategorien des Equipment Managers

### 3.5.5.2 Bearbeiten von Equipment Kategorien

Zum Verlauf der Bearbeitung einer der Kategorien gibt es in *Abb. 28 - Seite 46* ein Entscheidungsdiagramm.

Wird in der Editier- Option *-Modify...*- ausgewählt, kann eine *group*, *category* oder *subcategory* bearbeitet werden. Zum Bearbeiten einer *group* muss zunächst über das Dropdownmenü die zu bearbeitende *group* ausgewählt werden. Die Anzeige und das Auswerten des ausgewählten Eintrags dieses Dropdownmenüs funktioniert genauso wie oben erläutert. Diese Auswahl ( $\triangleq$  der *idgroup* des ausgewählten Eintrags) wird in ein verstecktes Textfeld (*type= hidden*) mit der *id listSelModGroupname* zwischengespeichert. Der neue Name der *group* wird in dem Textfeld darunter eingetragen. Wurde dieses Textfeld gefüllt und der Save-Button gedrückt, werden die Textfeldinhalte in Variablen gespeichert. Der geänderte Name wird mit den Namen der *groups* in der Datenbank verglichen, sodass keine *group* doppelt in der Datenbank existieren kann. Gibt es den Namen schon, wird der Nutzer über eine Nachricht darüber informiert. Ist der Eintrag noch nicht vorhanden, wird der Name der *group* überschrieben.

### 3.5.5.3 Hinzufügen von Equipment Kategorien

Zum Verlauf des Hinzufügens einer der Kategorien gibt es in *Abb. 29 - Seite 47* ein Entscheidungsdiagramm.

Wird in der Editier- Option *-Add...*- ausgewählt kann eine *group*, *category* oder *subcategory* hinzugefügt werden. In das angezeigte Textfeld wird der neue Name der *group* eingegeben. Anschließend wird in der Datenbank überprüft, ob es schon einen Eintrag mit diesem Namen gibt. Ist der Name noch nicht vorhanden, wird ein neuer Eintrag in die Datenbank angelegt. Beim Hinzufügen einer *category* oder *subcategory* wird die *idcollection* der *group* bzw. *category* als *parent* mit eingetragen. Die Auswahl des *group/category*- Dropdownmenüs wird beim Auslesen des Textfeldes mit dem Namen der neuen *category/subcategory* ausgelesen und mit übergeben.

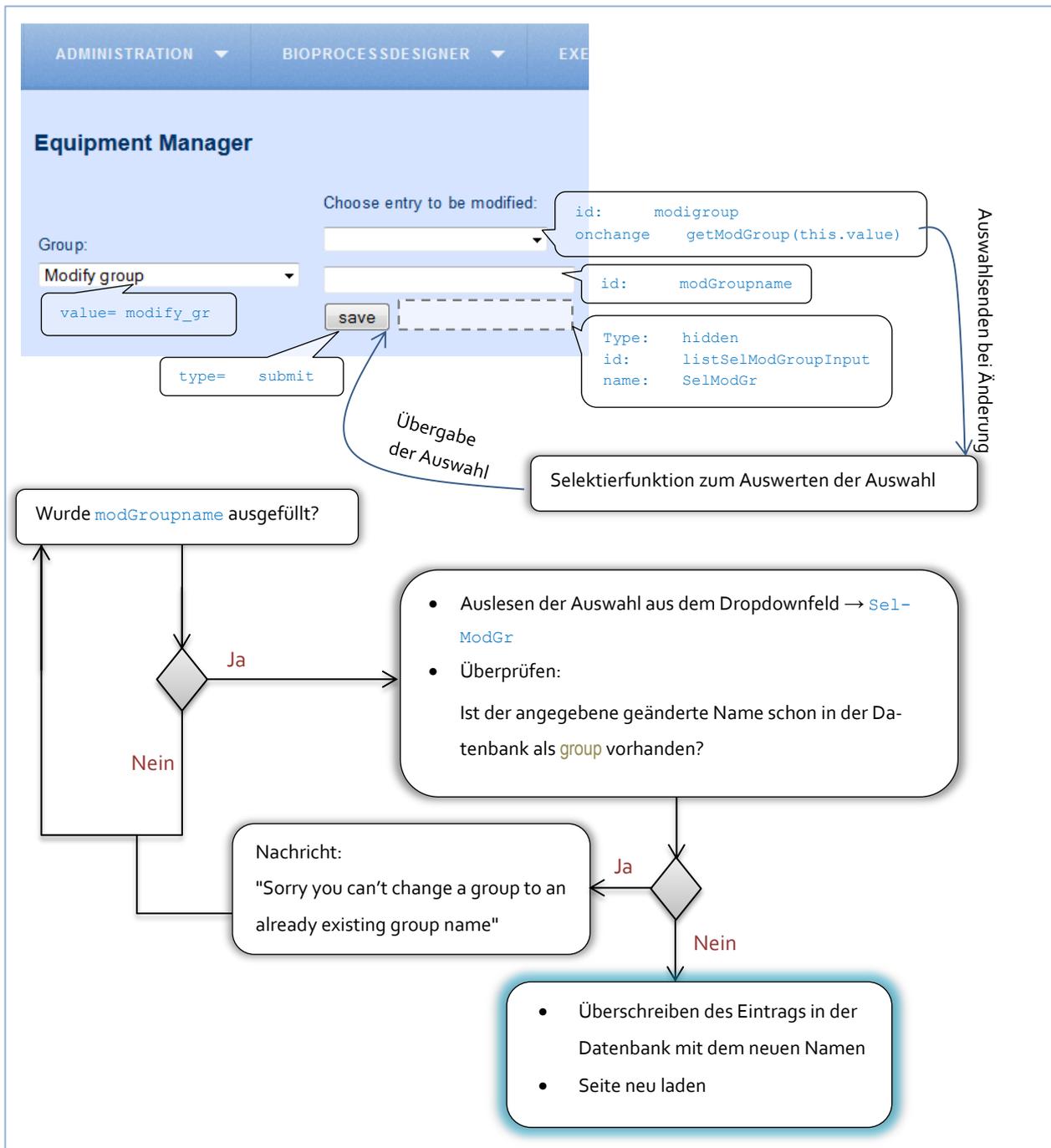


Abb. 28 Entscheidungsdiagramm zum Modifizieren einer group

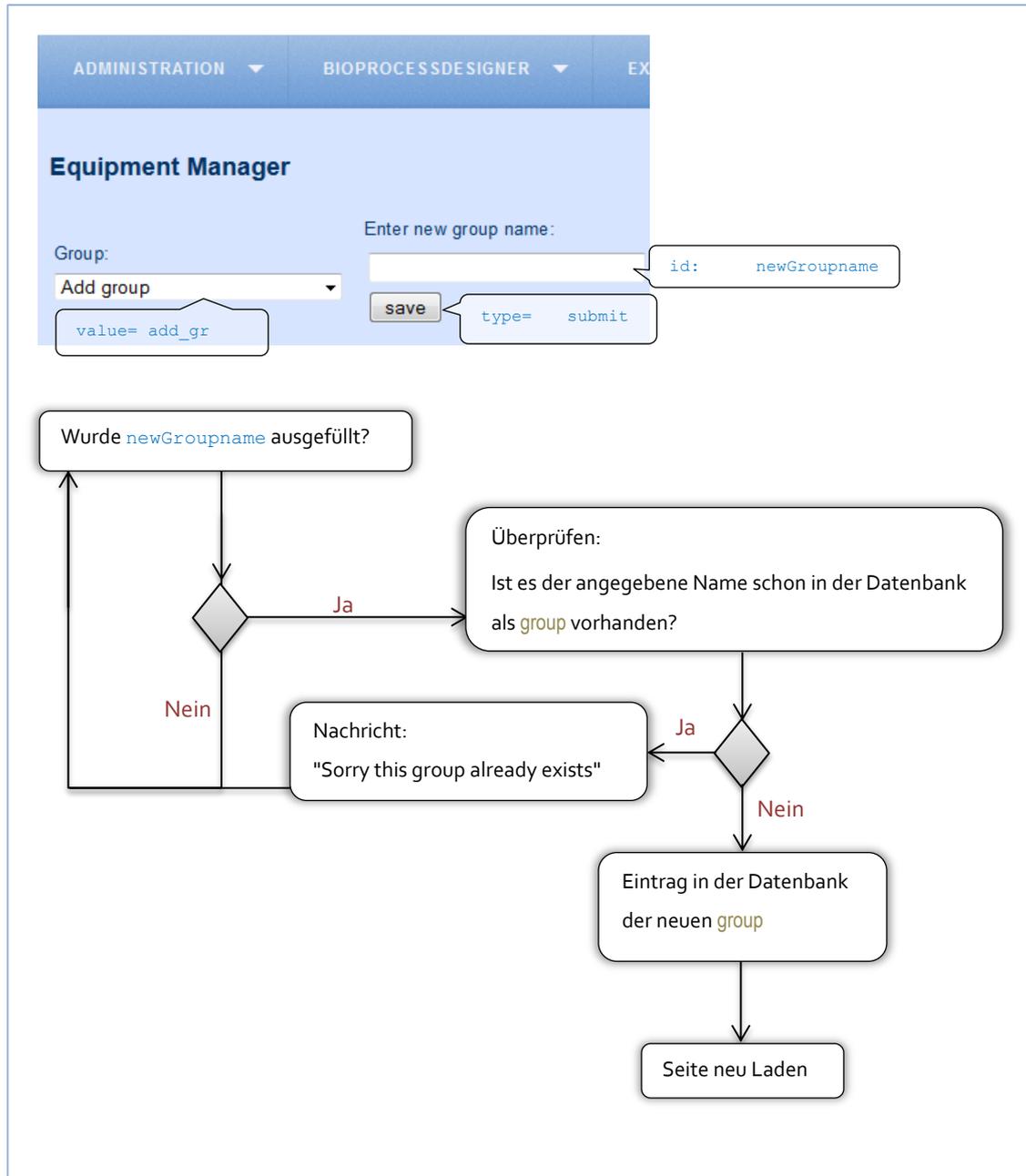


Abb. 29 Entscheidungsdiagramm zu Add group

#### 3.5.5.4 Löschen von Equipment Kategorien

Zum Verlauf des Löschens einer der Kategorien gibt es in *Abb. 30, Seite 49* ein Entscheidungsdiagramm.

Wird in der Editier- Option *-Delete...* ausgewählt kann eine *group*, *category* oder *subcategory* gelöscht werden. Zum Löschen einer *group* muss zunächst über das Dropdownmenü die zu löschende *group* ausgewählt werden. Die Anzeige und das Auswerten des ausgewählten Eintrags dieses Dropdownmenüs funktioniert wie oben erläutert. Die Auswahl wird in einem versteckten Textfeld (*type= hidden*) mit der id *listSelDelGroupName* zwischen gespeichert. Anschließend wird geprüft, ob die ausgewählte *group* irgendwelche *categories* hat. Wenn es keine gibt, wird die *group* gelöscht. Es kann keine *group* mit *categories*, keine *category* mit *subcategories* bzw. *subcategory* mit Equipment gelöscht werden. Bevor dies möglich ist, müssen alle darunter liegenden Kategorien gelöscht werden. Der Nutzer wird mit einer Nachricht darauf aufmerksam gemacht.

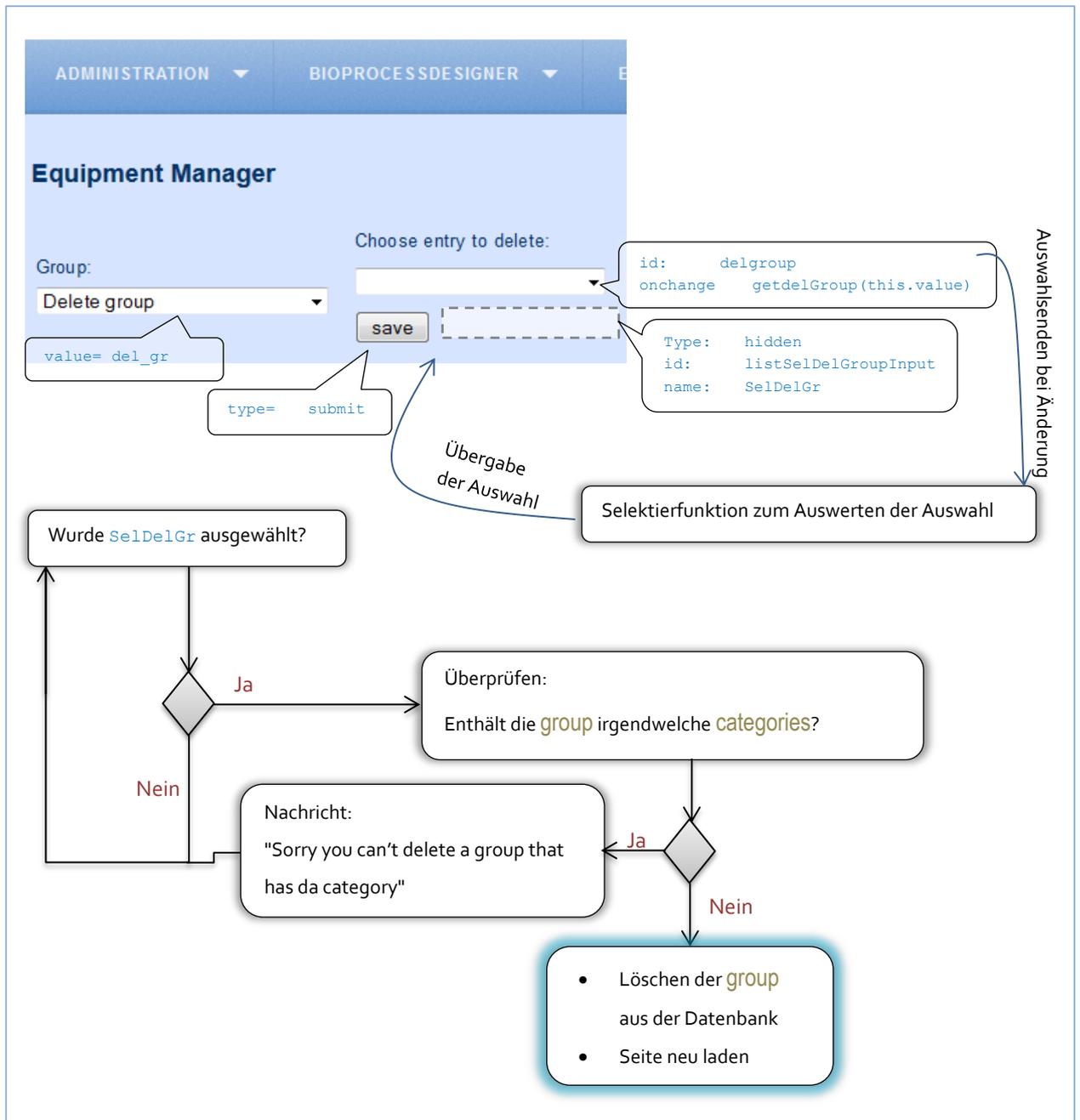


Abb. 30 Entscheidungsprozess zu löschen einer group

### 3.5.5.5 Anzeigen des angelegtem Equipments einer subcategory

Zum Verlauf des Anzeigens von Equipment gibt es in Abb. 31, Seite 51 ein Entscheidungsdiagramm.

Zum Anzeigen der **items** einer **subcategory** wird diese aus dem Dropdownmenü gelesen.

Die **idcollection** der **subcategory** wird in der Datenbanktabelle **collectioncontent** gesucht. Hier sind die vorab definierten Eigenschaften des Equipments gespeichert. Die Tabellen, die die Eigenschaften repräsentieren, werden aus der **basedata** Tabelle gesucht. Über den **content** des Eintrags der **collectioncontent** Tabelle werden die **ids** der Eigenschaftentabellen gespeichert und mit einer Tabelle über eine **while** Schleife mit einer Positionsnummer dargestellt.

Im Kopf der Tabelle mit der aufgelisteten **items** befindet sich das Dropdownmenü mit den Editier-Optionen. Diese funktionieren wie oben beschrieben.

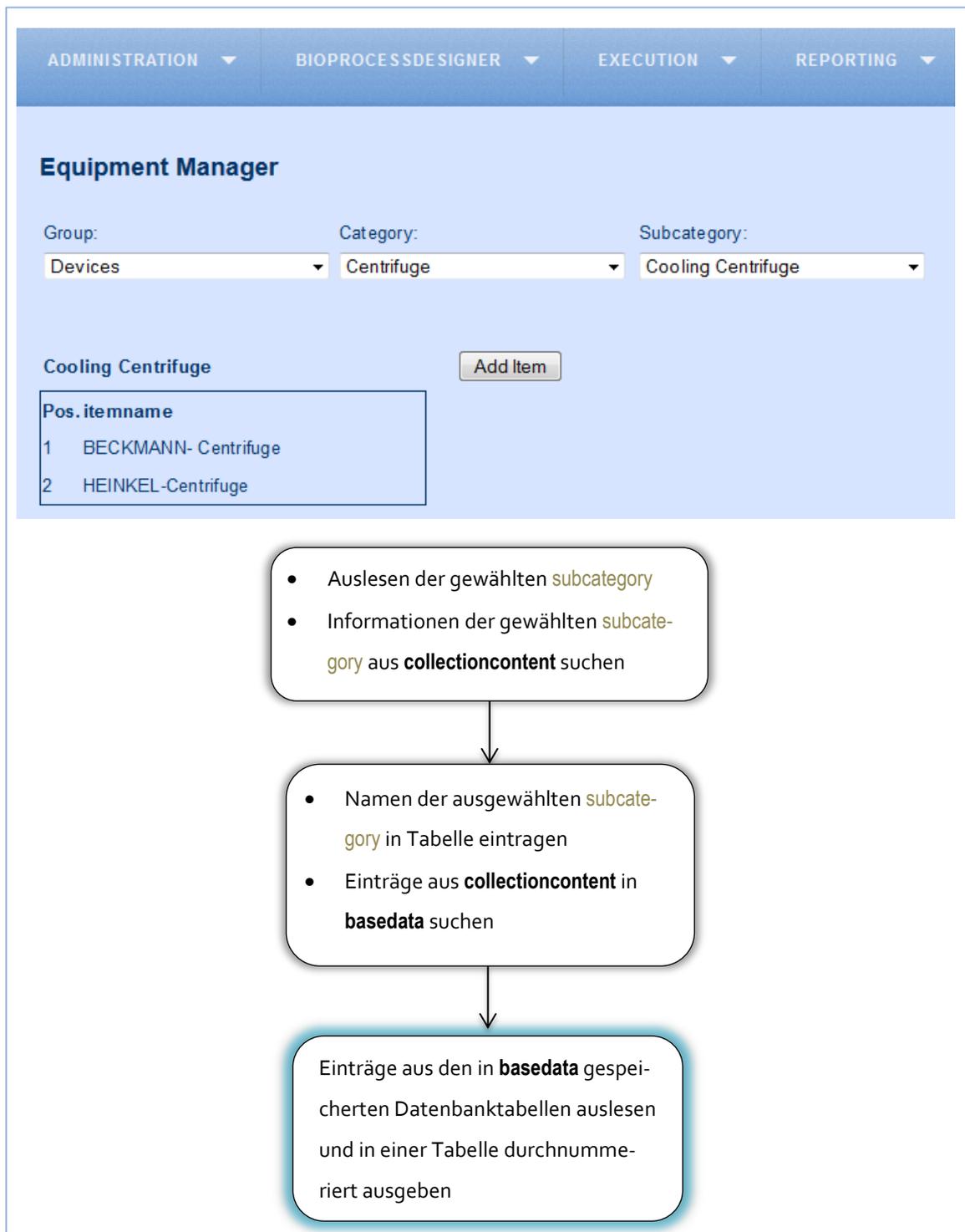


Abb. 31 Entscheidungsprozess zu Anzeigen vom (in der Datenbank) angelegten Equipment

## 4 Abschließende Betrachtungen

### 4.1 Zusammenfassung- Der BPD als Ganzes

In dieser Bachelorarbeit wurde die grafische Oberfläche und die Datenbank zur Nutzeranmeldung und Dokumentation von Equipment einer BioProcessDesigner Software entwickelt. Es wurde ein Menü generiert, um die Planung und Ausführung bioverfahrenstechnischer Prozesse zu ermöglichen. Durch die entwickelte Nutzerverwaltung ist ein Rechtesystem für die Software- und Projektstruktur entstanden, die eingeschränkte Rechte und Nutzungsvielfalt definiert. Durch die Registrierung werden neue Nutzer in der Datenbank angelegt. Durch die Charakterisierung eines Nutzers in der Datenbank werden die wichtigsten Informationen zum Kontaktieren gespeichert. Zudem werden doppelte Registrierungen vermieden. Beim Login wird berücksichtigt, dass nicht bestätigten Nutzern der Login verweigert wird. Nach dem Login wird der Zugriff auf Softwareteile an das Rechtesystem der Nutzerverwaltung angepasst. Als Beginn der Prozessplanung und -gestaltung wurde die Editierung und Kategorisierung von Equipment in der Datenbank und der Softwareoberfläche umgesetzt. Durch die Datenbankstruktur des Equipments wurde eine hohe Flexibilität ermöglicht.

### 4.2 Ausblick: Vorschläge zur Weiterentwicklung des BPD

Der Grundgedanke zur Vereinfachung des Laboralltags der Bioverfahrenstechnik wurde in dieser Arbeit grob skizziert. Beim Entwickeln sind immer wieder kleinere oder größere Probleme aufgetreten. Nicht immer ist es gelungen, diese zu beherrschen. Zunächst ist es Ziel der Software, einen beschränkten Zugriff für externe zu gewährleisten. Durch einen LOGIN Formular ohne Menü würde dies noch weiter ausgebaut werden. Hat ein Nutzer sein Passwort vergessen, kann ihm derzeit nicht geholfen werden. Er müsste sich neu registrieren, um sich wieder einloggen zu können. Bisher gibt es auch keine Möglichkeit sein Passwort zu ändern.

Die Benennung der Menüunterpunkte kann derzeit noch für Verwechslung mit Aufgabenbereichen der **userclass** erzeugen. Hier wird des Öfteren der Begriff des 'MANAGERS' genutzt. Ein Austausch gegen den Begriff 'Management' wäre zu überdenken, um nicht irrtümlich davon auszugehen, dass diese Menüteile nur für den **manager** zugänglich sind.

Die Verwaltung von Equipment bietet schon jetzt eine hohe Flexibilität. Durch die Planung des genauen Datenbankdesigns der **basedata** Tabelle und deren Darstellungsform können dann Eigenschaften des Equipments angezeigt, hinzugefügt und bearbeitet werden. Des Weiteren funktioniert das Ein- und Ausblenden der Editier- Optionen des Equipment Managers noch nicht nutzerfreundlich.

Mit der aktuellen Oberfläche kann der **userclass** Status der Nutzer angepasst werden, Nutzer die einen **userclass** Status höher als **anybody** haben aber nicht gelöscht werden. Ein Nutzer kann, nachdem er zugelassen wurde, auch nicht wieder auf den userclass Status **anybody** gesetzt werden.

### 4.3 Danksagung

Diese Arbeit wurde mit Unterstützung der Hochschule für Angewandte Wissenschaften Hamburg –Fakultät Life Sciences angefertigt.

Herrn Prof. Dr. Boris Tolg danke ich für viele fruchtbare Diskussionen, wertvolle Hinweise und Ratschläge und eine ganz allgemein vorbildliche Betreuung.

Herrn Prof. Dr. Ernst Sanders danke ich für vertiefende Diskussionen und eine allgemein vorbildliche Betreuung.

Für die Unterstützung zum Programmieren und der Software Entwicklung danke ich Paul Evert.

Zu gleich danke ich Herrn Dr. Michael Riebandt für die Beratung und die kritische Durchsicht des Manuskripts.

## 5 Anhang

### 5.1 Literaturverzeichnis

- **Bergmann, Sebastuan. 2005.** Professionelle Softwareentwicklung mit PHP 5. Heidelberg : dpunkt.verlag GmbH, 2005, S. 3-5.
- **CHEMSTATIONS EUROPE GMBH. 2015.** Chemstations. [Online] Chemstations Europe GmbH, 2015. [Zitat vom: 5. Februar 2015.]  
<http://www.chemstations.eu/produkte/chemcad-steady-state/>.
- **Consortium, PACE. 2004-2008.** PACE REPORT. [Online] PACE Consortium, 2004-2008. [Zitat vom: 5. Februar 2015.]  
[http://www.istpace.org/Web\\_Final\\_Report/WP\\_6\\_microfluidic\\_complementation/index.html](http://www.istpace.org/Web_Final_Report/WP_6_microfluidic_complementation/index.html).
- **Decker, Eva.** Wikipedia. [Online] [Zitat vom: 5. Februar 2015.]  
[http://upload.wikimedia.org/wikipedia/commons/b/b1/Bioreaktor\\_quer2.jpg](http://upload.wikimedia.org/wikipedia/commons/b/b1/Bioreaktor_quer2.jpg).
- **Hahn, Axel, Häusler, Stefan und groß Austing, Stephan. 2013.** Quantitatives Entwicklungsmanagement. Berlin Heidelberg : Springer Vieweg, 2013, 1, S. 15-16.
- **IMCOR GmbH 2014. 2014.** LIMS.DE. [Online] IMCPR GmbH, 2014. [Zitat vom: 5. Februar 2015.] <http://www.lims.de/grundlagen.htm>.
- **JQuery Foundation. 2015.** JQuery. [Online] 2015. [Zitat vom: 16. Februar 2015.]  
<https://jquery.org/history/>.
- **Laborenz, Kai. 2006.** CSS- Praixs. Bonn : Galileo Press, 2006, S. 17-25.
- **Maul, Peter. 20.12.2013.** *Definition von Anforderungen an eine Datenbank für einen BioProcessDesigner mit dem Softwaremodellierungswerkzeug Enterprise Architect.* Hamburg : -, 20.12.2013.
- **Musciano, Chuck und Bill, Kennedy. 1999.** HTML Das umfassende Referenzwerk. Köln : O'Reilly Verlag GmbH & Co. KG, 1999, S. 8.
- **204-2008.** PACE REPORT. [Online] PACE Consortium, 204-2008. [Zitat vom: 5. Februar 2015.]  
[http://www.istpace.org/Web\\_Final\\_Report/WP\\_6\\_microfluidic\\_complementation/index.html](http://www.istpace.org/Web_Final_Report/WP_6_microfluidic_complementation/index.html).
- **Patrick. 2007.** Ajaxschmiede.de. [Online] 29. November 2007. [Zitat vom: 08. Februar 2015.] <http://www.ajaxschmiede.de/jquery/jquery-ein-maechtiges-und-effizientes-werkzeug/>.
- **PHP-Dokumentationsgruppe.** php. [Online] <http://php.net/manual/de/>.

- **Runge, Nils und Bautsch, Jan. 2014.** *Untersuchung der Induktionsphase in einem HCDC Kultivierungsprozess mit E. coli BL21 pGLO und Entwicklung einer Webanwendung zur Archivierung und Präsentation von Versuchsdaten.* Hamburg : s.n., 2014.
- **Thies, Thomas. 2010.** Einstieg in PHP 5.3 & MySQL 5.4. Bonn : Galileo Press, 2010, S. 15-17, 162, 192-194.
- **Vollendorf, Maximilian und Bongers, Frank. 2014.** jQuery Das umfassende Handbuch. Bonn : Galileo Press, 2014, S. 17-28.
- Zoonar. [Online] [Zitat vom: 5. Februar 2015.]  
[http://static.zoonar.de/img/www\\_repository1/cb/49/44/10\\_d01957a05cfdfdc238290e0ofdef3631.jpg](http://static.zoonar.de/img/www_repository1/cb/49/44/10_d01957a05cfdfdc238290e0ofdef3631.jpg).

## 5.2 Anhang A- Oberfläche des Equipment Managers

Der Equipment Manager soll dafür genutzt werden, neues Equipment einzupflegen (add), vorhandenes Equipment zu bearbeiten (modify) oder zu löschen (delete).

In Abb. 32 wird die Startansicht des Equipment Managers gezeigt. Das Equipment wird in die Kategorien **group**, **category** und **subcategory** unterteilt.

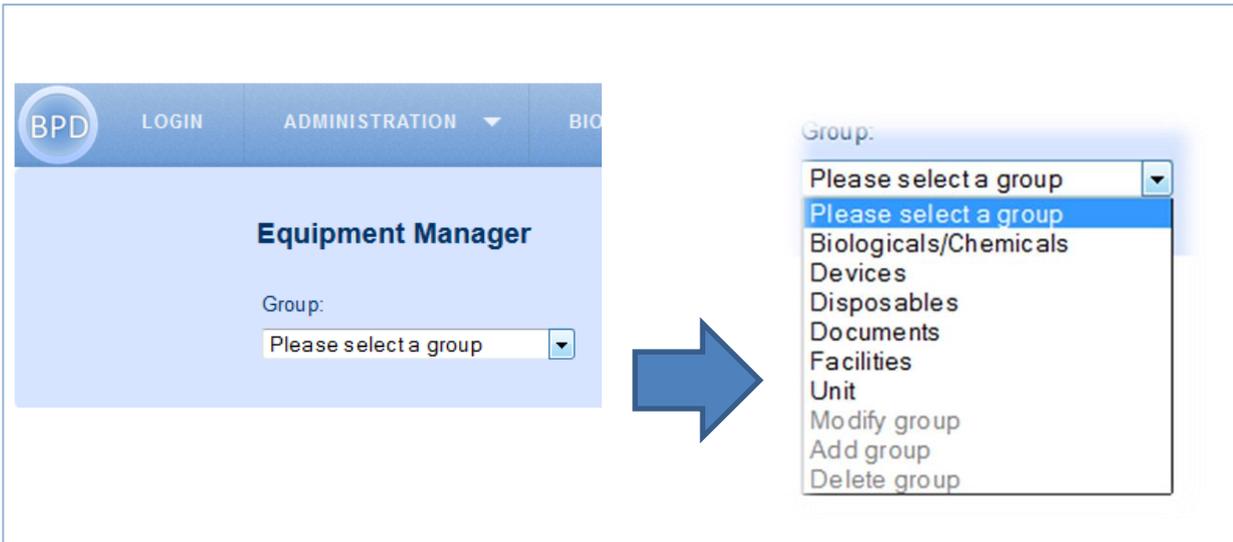


Abb. 32 Startansicht des Equipment Managers

Die **group** fasst alle unter ihr liegenden **categories** zusammen. Diese wiederum fasst alle unter ihr liegenden **subcategories** zusammen. Durch das Datenbank Design können alle Kategorien jederzeit beliebig erweitert werden (vgl. Kapitel 3.4.2 - Das Datenbank Design des Equipment Managers - Seite 29).

Über ein Dropdown Menü lässt sich eine **group** auswählen (vgl. Abb. 32).

Durch das Auswählen wird automatisch ein Dropdownmenü angezeigt, welches die entsprechenden **categories** der **group** enthält, siehe Abb. 33

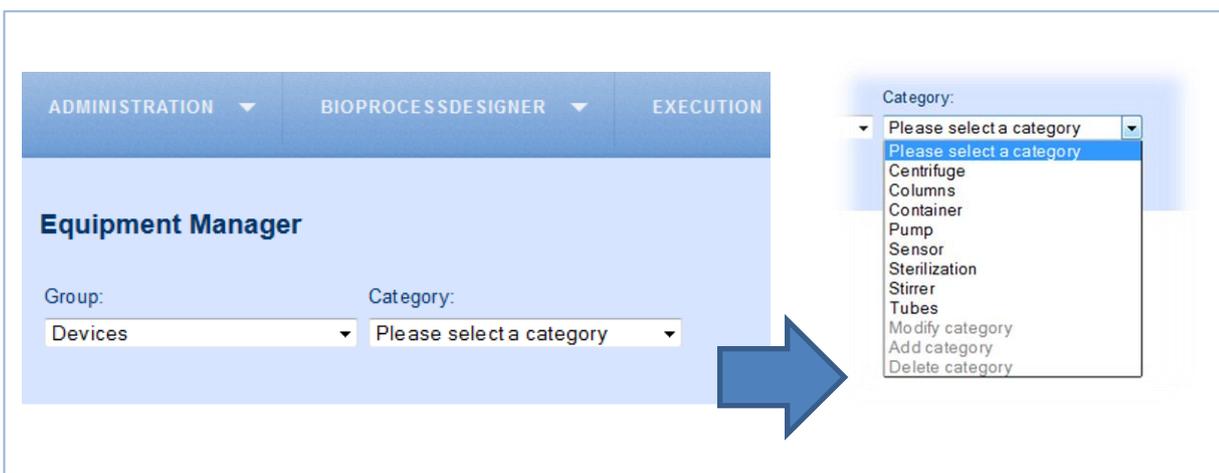


Abb. 33 Ansicht nach Auswahl der **group** Unit, nun werden die entsprechenden **categories** angezeigt

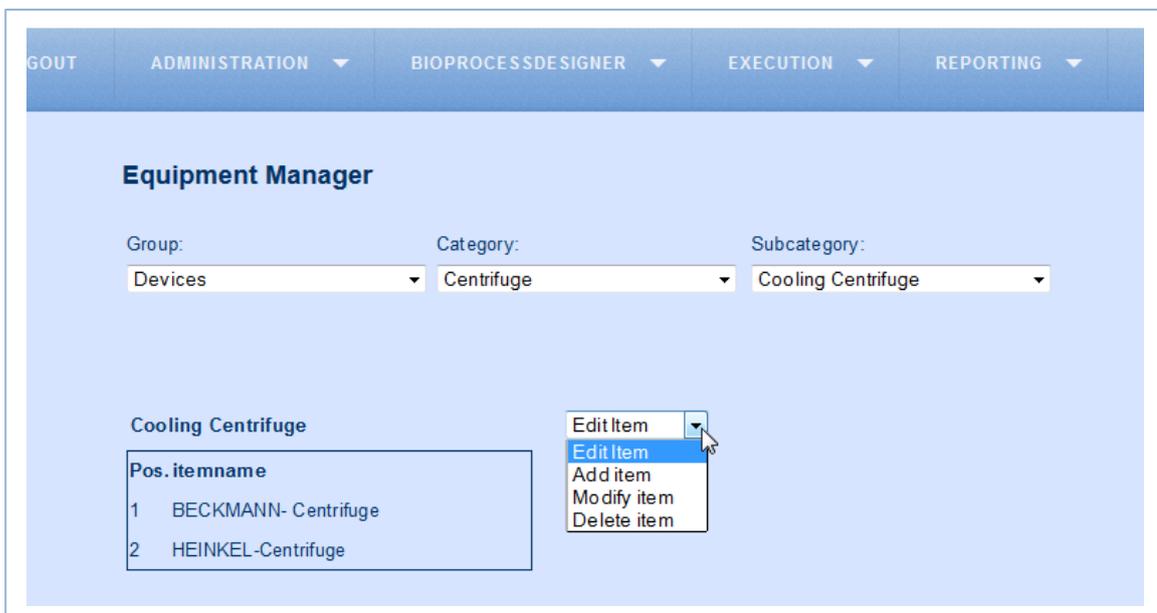
Wählt man eine **category** aus, erscheint automatisch ein Dropdownmenü mit den dazugehörigen **subcategories** (vgl. Abb. 34).

Die aufgelisteten Einträge der **group**, **category** und **subcategory** werden alphabetisch sortiert angezeigt.



**Abb. 34** Ansicht bei der **group** Auswahl Unit, mit der **category**: Devices und den dazugehörigen **subcategories**

Alle Dropdownmenü enthalten standardmäßig die Auswahlpunkt **Modify**, **Add** und **Delete**; angepasst auf den entsprechenden Inhalt des Dropdownmenüs z. B. **Modify subcategory**, **Add subcategory** und **Delete Subcategory** (vgl. Abb. 34).



**Abb. 35** Anzeige bei der aufgelisteten **Equipment items** einer **subcategory**

Wird eine **subcategory** ausgewählt, werden darunter die **items** der **subcategory** als Tabelle mit ihren Eigenschaften aufgelistet. Im Dropdownmenü neben der Tabelle können die Editier-Optionen **Add item**, **Modify item** und **Delete item** ausgewählt werden (vgl. Abb. 36).

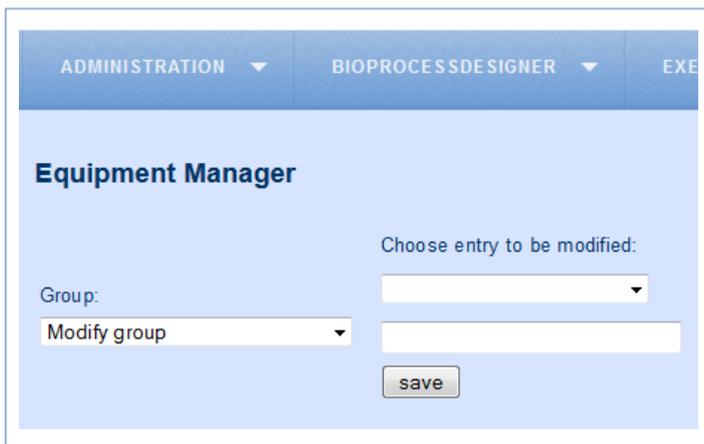


Abb. 36 Anzeige bei der Dropdown Auswahl  
Modify group

Wird der Menüpunkt Modify group ausgewählt, erscheint ein Dropdownmenü, ein Eingabefeld und ein save-Button (vgl. Abb. 36). Im Dropdownmenü kann die group ausgewählt werden, bei der der Name geändert werden soll. Der neue Name wird in dem Eingabefeld darunter eingegeben. Über den save- Button wird die

Eingabe und die Änderung bestätigt. Anschließend lädt die Seite neu und die modifizierte group ist im Dropdownmenü von group zu sehen. Die categories und subcategories findet man nun unter dem geänderten Namen der group.

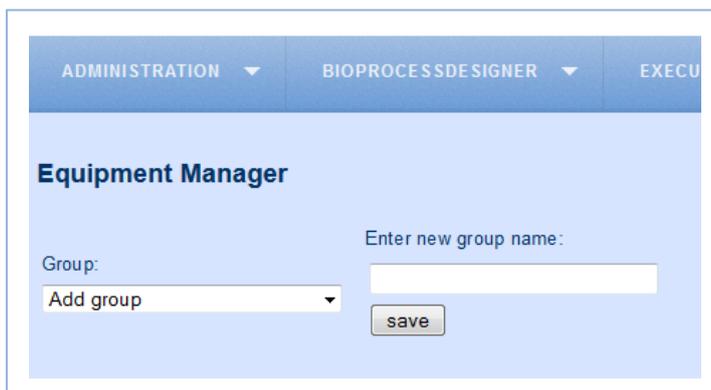


Abb. 37 Anzeige bei der Dropdown Auswahl  
Add group

Wird der Menüpunkt Add group ausgewählt, erscheint ein Eingabefeld und ein save-Button (vgl. Abb. 37).

Der Name der neuen group wird in das Eingabefeld eingegeben. Über den save- Button wird die Eingabe bestätigt. Anschließend lädt die Seite neu und die neue group ist im

Dropdownmenü von group zu sehen.

Man kann keine group mit einem schon bestehenden Namen erstellen.

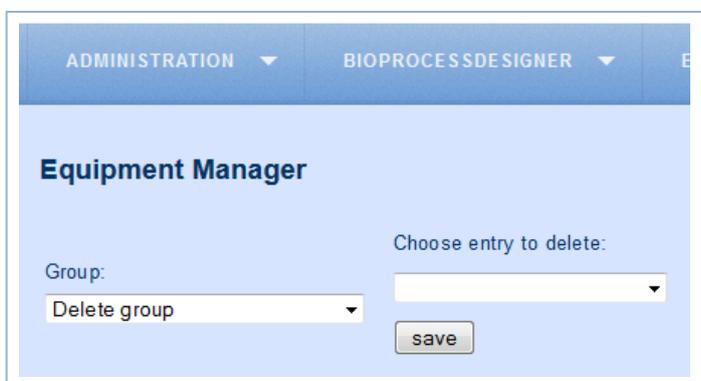


Abb. 38 Anzeige bei der Dropdown Auswahl  
Delete group

Wählt man die neue group aus, enthält sie keine category oder subcategory. Diese können über die Dropdownmenüpunkte Add category und Add subcategory hinzugefügt werden.

Wird der Menüpunkt Delete group ausgewählt, erscheint ein Dropdownmenü und ein save-

Button (vgl. Abb. 38).

Die **group**, die gelöscht werden soll, kann über das Dropdownmenü ausgewählt werden. Über den save- Button wird die Auswahl bestätigt.

Eine **group** kann nur gelöscht werden, wenn sie keine **category** oder **subcategory** enthält.

Sind diese Bedingungen erfüllt, wird die **group** gelöscht, die Seite lädt neu und die gelöschte **group** ist im Dropdownmenü nicht mehr vorhanden.

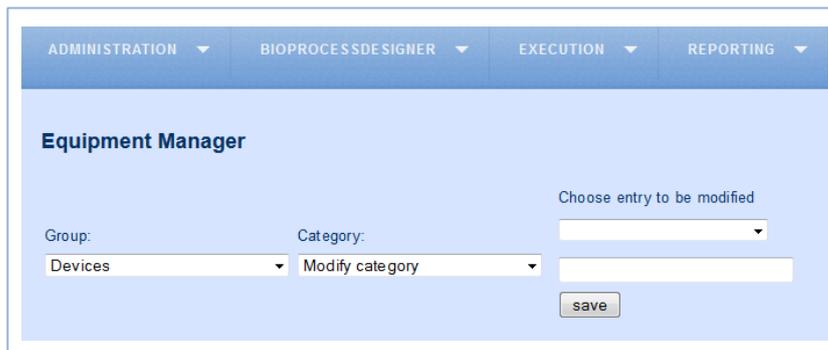
The screenshot shows the 'Equipment Manager' interface. At the top, there are four navigation tabs: 'ADMINISTRATION', 'BIOPROCESSDESIGNER', 'EXECUTION', and 'REPORTING'. Below the tabs, the title 'Equipment Manager' is displayed. The main content area contains a form with two dropdown menus: 'Group:' and 'Category:'. The 'Group:' dropdown is set to 'Devices'. The 'Category:' dropdown is set to 'Modify category'. To the right of the 'Category:' dropdown is a label 'Choose entry to be modified' and a small dropdown menu. Below the 'Category:' dropdown is an input field for the new category name. A 'save' button is located at the bottom right of the form.

Abb. 39 Anzeige bei der Dropdown Auswahl Modify category

Wir der Menüpunkt Modify category ausgewählt, erscheint auch hier ein Dropdownmenü, ein Eingabefeld und ein save-Button (vgl. Abb. 39).

Im Dropdownmenü kann die **category** ausgewählt werden, bei der der Name geändert werden soll.

Der neue Name wird in dem Eingabefeld darunter eingegeben. Über den save- Button wird die Eingabe und die Änderung bestätigt. Anschließend lädt die Seite neu und die modifizierte **category** ist im Dropdownmenü von **category** bei Auswahl der entsprechenden **group** zu sehen. Die **subcategories** findet man nun unter dem geänderten Namen der **category**.

Im Dropdownmenü kann

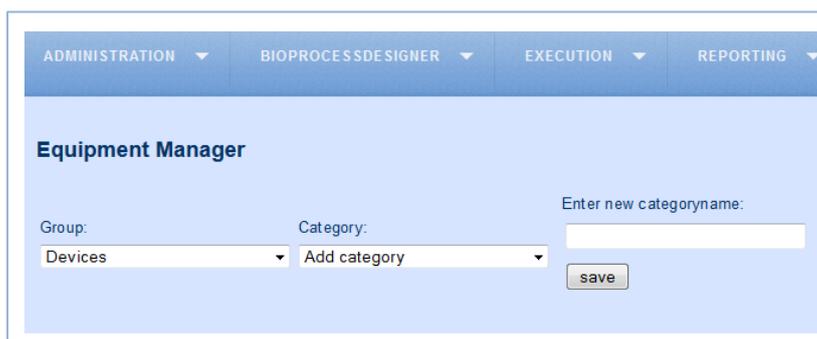
The screenshot shows the 'Equipment Manager' interface. At the top, there are four navigation tabs: 'ADMINISTRATION', 'BIOPROCESSDESIGNER', 'EXECUTION', and 'REPORTING'. Below the tabs, the title 'Equipment Manager' is displayed. The main content area contains a form with two dropdown menus: 'Group:' and 'Category:'. The 'Group:' dropdown is set to 'Devices'. The 'Category:' dropdown is set to 'Add category'. To the right of the 'Category:' dropdown is a label 'Enter new categoryname:' and an input field. A 'save' button is located at the bottom right of the form.

Abb. 40 Anzeige bei der Dropdown Auswahl Add category

Wird der Menüpunkt Add category ausgewählt, erscheint ein Eingabefeld und ein save-Button (vgl. Abb. 40).

Der Name der neuen **category** wird in das Eingabefeld eingegeben.

Über den save- Button wird die Eingabe bestätigt. Anschließend lädt die Seite neu und die neue **category** ist im Dropdownmenü von **category** bei Auswahl der entsprechenden **group** zu sehen.

Man kann keine **category** mit einem schon bestehenden Namen erstellen.

Wählt man die neue **category** aus, enthält sie keine **subcategory**. Diese können über die Dropdownmenüpunkte Add subcategory hinzugefügt werden.



Abb. 41 Anzeige bei der Dropdown Auswahl Delete category

Wird der Menüpunkt Delete category ausgewählt, erscheint ein Dropdownmenü und ein save-Button (vgl. Abb. 41). Die category, die gelöscht werden soll, kann über das Dropdownmenü

ausgewählt werden. Über den save- Button wird die Auswahl bestätigt.

Eine category kann nur gelöscht werden, wenn sie keine subcategory enthält.

Ist diese Bedingung erfüllt, wird die category gelöscht, die Seite lädt neu und die gelöschte category ist im Dropdownmenü nicht mehr zu vorhanden.

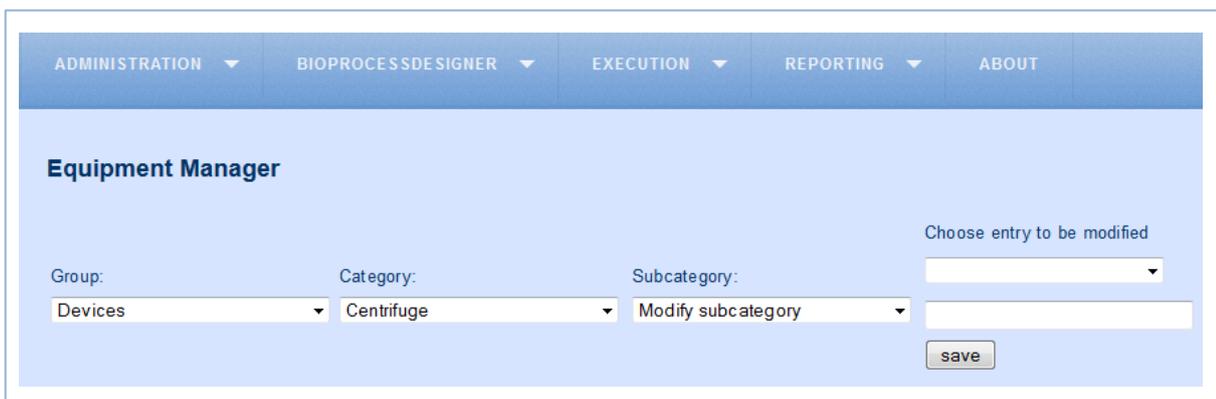


Abb. 42 Anzeige bei der Dropdown Auswahl Modify subcategory

Wir der Menüpunkt Modify subcategory ausgewählt, erscheint auch hier ein Dropdownmenü, ein Eingabefeld und ein save-Button (vgl. Abb. 42).

Im Dropdownmenü kann die subcategory ausgewählt werden bei der, der Name geändert werden soll.

Der neue Name wird in dem Eingabefeld darunter eingegeben. Über den save- Button wird die Eingabe und die Änderung bestätigt. Anschließend lädt die Seite neu und die modifizierte subcategory ist im Dropdownmenü von subcategory bei Auswahl der entsprechenden category zu sehen. Die items findet man nun unter dem geänderten Namen der subcategory.



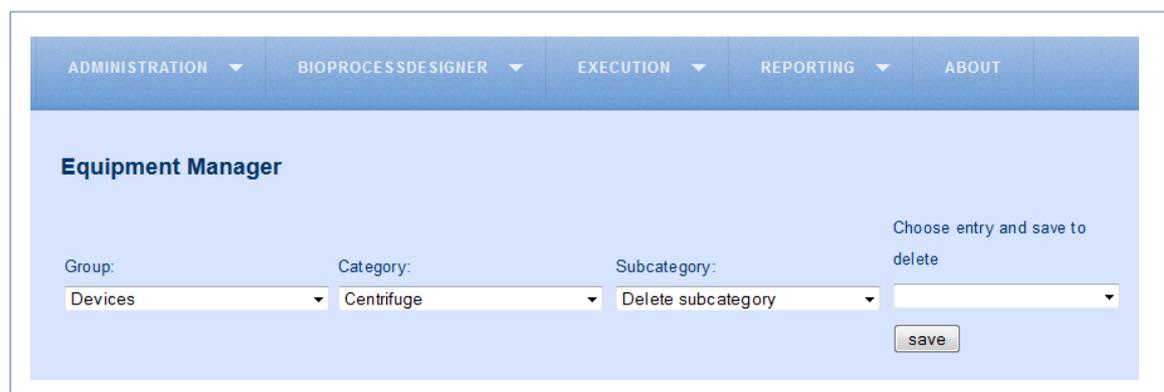
**Abb. 43** Anzeige bei der Dropdown Auswahl Add subcategory

Wird der Menüpunkt Add subcategory ausgewählt, erscheint ein Eingabefeld und ein save-Button (vgl. Abb. 43).

Der Name der neuen subcategory wird in das Eingabefeld eingegeben. Über den save-Button wird die Eingabe bestätigt. Anschließend lädt die Seite neu und die neue subcategory ist im Dropdownmenü von subcategory bei Auswahl der entsprechenden category zu sehen.

Man kann keine subcategory mit einem schon bestehenden Namen erstellen.

Wählt man die neue subcategory aus, enthält sie keine items. Diese können über die Dropdownmenüpunkte Add Item hinzugefügt werden.



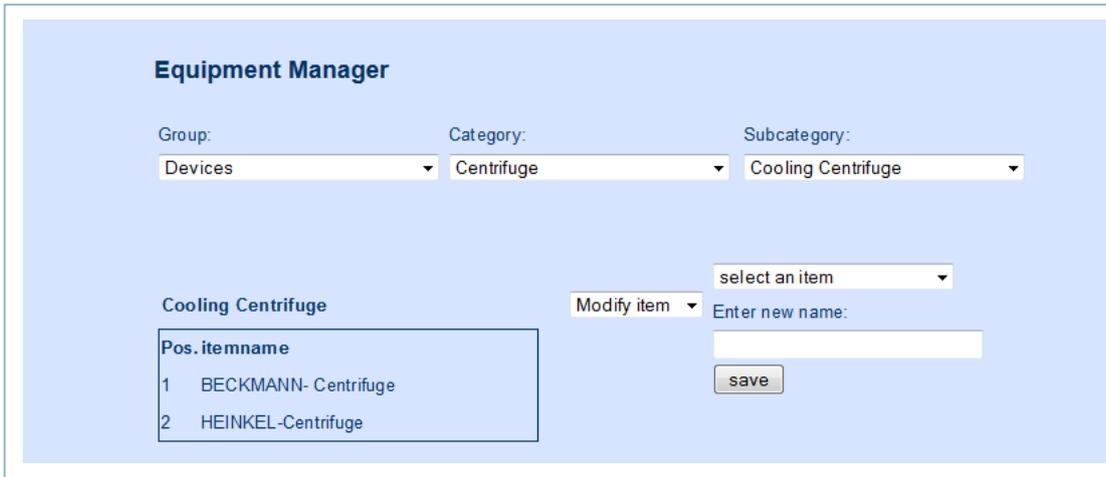
**Abb. 44** Anzeige bei der Dropdown Auswahl Delete subcategory

Wird der Menüpunkt Delete subcategory ausgewählt, erscheint ein Dropdownmenü und ein save-Button (vgl. Abb. 44).

Die subcategory, die gelöscht werden soll, kann über das Dropdownmenü ausgewählt werden. Über den save-Button wird die Auswahl bestätigt.

Eine subcategory kann nur gelöscht werden, wenn sie keine items enthält.

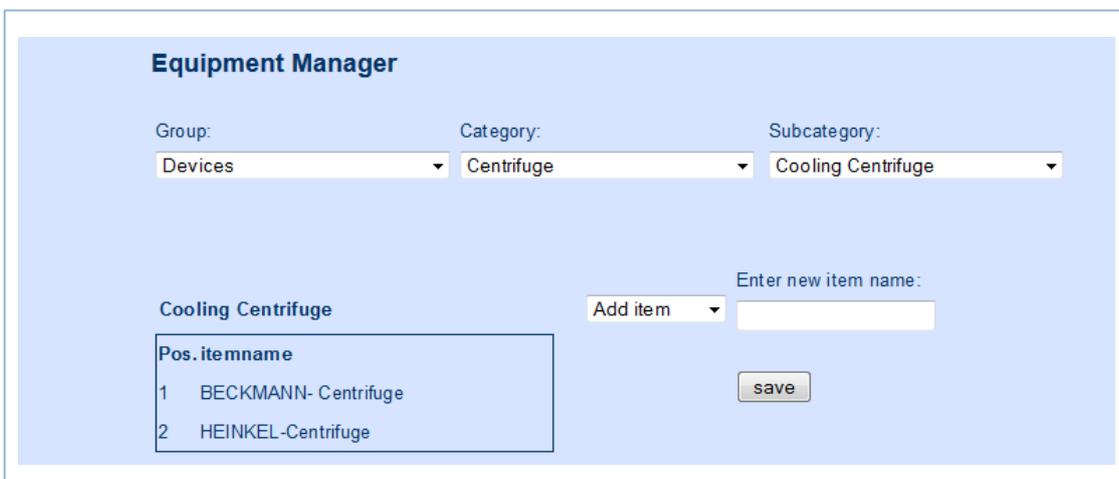
Ist diese Bedingung erfüllt, wird die subcategory gelöscht, die Seite lädt neu und die gelöschte subcategory ist im Dropdownmenü nicht mehr vorhanden.



**Abb. 45** Anzeige bei der Dropdown Auswahl Modify item

Wird der Menüpunkt **Modify item** ausgewählt, erscheint auch hier ein Dropdownmenü, ein Eingabefeld und ein save-Button (vgl. Abb. 45).

Im Dropdownmenü kann das **item** ausgewählt werden bei der, der Name geändert werden soll. Der neue Name wird in dem Eingabefeld darunter eingegeben. Über den save- Button wird die Eingabe und die Änderung bestätigt. Anschließend lädt die Seite neu und das modifizierte **item** ist in der Auflistungstabelle zu finden.



**Abb. 46** Anzeige bei der Dropdown Auswahl Add item

Wird der Menüpunkt **Add item** ausgewählt, erscheint ein Eingabefeld und ein save-Button (vgl. Abb. 46).

Der Name des neuen **items** wird in das Eingabefeld eingegeben. Über den save- Button wird die Eingabe bestätigt. Anschließend lädt die Seite neu und das **item** ist in der Auflistungstabelle zu finden.

The screenshot shows the 'Equipment Manager' interface. At the top, there are three dropdown menus for 'Group' (set to 'Devices'), 'Category' (set to 'Centrifuge'), and 'Subcategory' (set to 'Cooling Centrifuge'). Below these, the title 'Cooling Centrifuge' is displayed. To the right, there is a 'Delete item' dropdown menu currently showing 'select an item' and a 'Delete' button. Below the title, a table lists items:

Pos.	itemname
1	BECKMANN- Centrifuge
2	HEINKEL-Centrifuge

**Abb. 47** Anzeige bei der Dropdown Auswahl Delete item

Wird der Menüpunkt Delete item ausgewählt, erscheint ein Dropdownmenü und ein save-Button (vgl. Abb. 47).

Das **item**, das gelöscht werden soll, kann über das Dropdownmenü ausgewählt werden. Über den save-Button wird die Auswahl bestätigt. Anschließend ist es nicht mehr Teil der Auflistungstabelle.

## 5.3 Anhang B- Programmcode