

Zweitklausur

Softwaretechnik II

Sommersemester 2023

Prof. Dr. Ralf H. Reussner

12.09.2023

Name: _____

Matrikelnummer: _____

Zur Klausur sind keine Hilfsmittel und kein eigenes Papier zugelassen. Die Bearbeitungszeit beträgt 90 Minuten. Die Klausur ist vollständig und geheftet abzugeben. Verwenden Sie ausschließlich dokumentenechte Stifte. Mit Bleistift oder roter Farbe geschriebene Angaben werden nicht bewertet. Lösen Sie die Aufgaben in leserlicher Schrift. Unlesbare Lösungen können naturgemäß nicht positiv bewertet werden.

Wenn das generische Maskulinum gewählt wurde, geschieht dies zur besseren Lesbarkeit und zum einfachen Verständnis der Aufgabenstellung. Sofern nicht anders angegeben, beziehen sich Angaben im Sinne der Gleichbehandlung auf Vertretende aller Geschlechter.

Aufgabe	1	2	3	4	5	6	Σ
Maximal	10	21	16	9	20	14	90
K1							
K2							
K3							

Name:

Matrikelnummer:

Aufgabe 1: Verschiedenes (10 Punkte)

a) Was versteht man unter dem Begriff Softwaretechnik? (1 Punkt)

b) Nennen Sie zwei Zwecke von Code Reviews. (1 Punkt)

c) Nennen Sie die vier Vorteile einer expliziten Softwarearchitektur aus der Vorlesung. (2 Punkte)

Name:

Matrikelnummer:

- d) Wie hängen Fehlerkosten (während der Entwicklung und Nutzung eines Systems) mit den Kosten der Anforderungsspezifikation zusammen? Wo befindet sich das Optimum? Sie dürfen entscheiden, ob Sie eine grafische oder textuelle Beschreibung wählen. Wenn Sie ein Schaubild zeichnen, denken Sie daran, die Achsen zu beschriften. (2 Punkte)



- e) Was versteht man unter dem Prinzip der Separation of Concerns? Nennen Sie einen daraus entstehenden Vorteil. (1 Punkt)



Name:

Matrikelnummer:

f) Nennen und erklären Sie drei der zehn Prinzipien der sicheren Softwareentwicklung. (3 Punkte)

Aufgabe 2: Anforderungserhebung (21 Punkte)

Im Folgenden betrachten wir die Entwicklung eines Projekts der Firma *PhischIT*.

Das WaterHealth-Tracking-System (WTS) von der Firma *PhischIT* soll lokalen Tauchern und Biologen die Möglichkeit bieten, die aktuelle Gesundheit von lokalen Seen zu dokumentieren und Missstände frühzeitig zu kommunizieren. Hierzu können sich Taucher und Biologen im System einloggen. Während der Dokumentation muss das Gewässer eindeutig markiert werden. Zusätzlich können Taucher Fotos von Unterwasser gesehenen Pflanzen (Flora) und Tieren (Fauna) hochladen. Um die lokale Umgebung der Pflanzen und Tiere unter Wasser zu erfassen, werden dabei Messewerte zur Wasserqualität eingetragen. Damit die Messungen einheitlich sind, erhalten alle Taucher gegen Hinterlegung einer Kautions ein Gerät der Firma *SeaSafe*. Beim Speichern der Aufzeichnung des Tauchgangs wird der Name des Tauchers mit allen Daten verknüpft und gespeichert. Die von Tauchern gemachten Aufzeichnungen können von Biologen analysiert werden, um Berichte über die Gesundheit eines Gewässers zu schreiben. Hierfür können Biologen ein bestimmtes Gewässer wählen und die Aufzeichnungen dazu aufrufen. Durch das Zusammenstellen verschiedener Aufzeichnungen können Biologen so Belege für ihren Bericht hinterlegen. Gemeinden können dann die Berichte für ihre Gewässer bei *PhischIT* erwerben.

a) Nennen Sie, neben Tauchern und Biologen, zwei weitere Stakeholder, für die die Entwicklung des Tracking-Systems relevant ist und begründen Sie kurz deren Interesse. (1 Punkt)

- Taucher
- Biologen

Name:

Matrikelnummer:

b) Nennen Sie drei Empfehlungen für gut geschriebene Anforderungen (*basic writing recommendations*). (1,5 Punkte)

c) In der Vorlesung haben Sie die Anforderungsklassifikation nach Glinz kennengelernt. Beschreiben Sie, wofür welche Anforderungsart (*functional, quality, constraint*) und Repräsentation (*operational, quantitative, qualitative, declarative*) genutzt wird. (3,5 Punkte)

Anforderungsart	Beschreibung
functional	
quality	
constraint	

Repräsentation	Beschreibung
operational	
quantitative	
qualitative	
declarative	

Name:

Matrikelnummer:

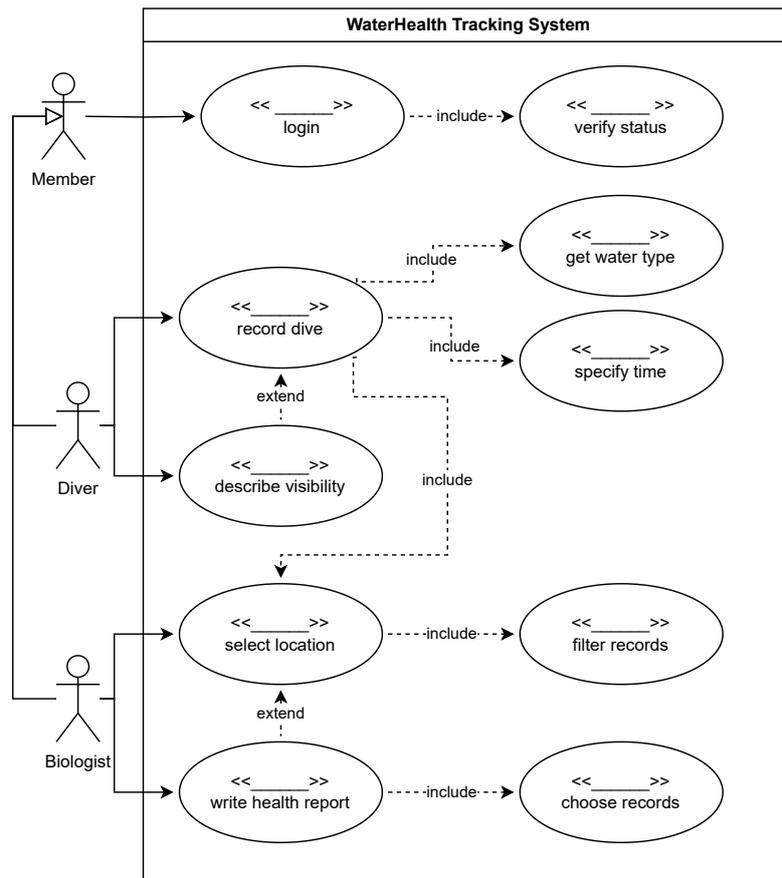
- d) Ergänzen Sie in den nachfolgenden Tabellen für die gegebene Anforderungen (R1 bis R6) sinnvolle Anforderungsbeschreibungen sowie passende Unterkategorien der Anforderungen. Beachten Sie, dass Ihre Lösung die Klassifikation von Glinz sowie die Empfehlungen aus Teilaufgabe b) erfüllt. *Hinweis: Wenn Sie zum Formulieren von Anforderungen mehr Details brauchen, als im Einleitungstext gegeben sind, dürfen Sie sich diese Details ausdenken.* (15 Punkte)

Nr.	Anforderung	Art	Unterkategorie	Repräsentation
R1	Datenschutz	constraint		declarative
R2	Speicherplatz	constraint		quantitative
R3	Erstellen von Aufzeichnungen	functional		qualitative
R4	Importieren von Daten	functional		operational
R5	Bedienbarkeit	quality		qualitative
R6	Erweiterbarkeit	constraint		qualitative

Nr.	Anforderungsbeschreibung
R1	
R2	
R3	
R4	
R5	
R6	

Aufgabe 3: Use-Cases (16 Punkte)

Basierend auf den Anforderungen plant *PhischIT* erst einmal einen Prototypen für das WaterHealth-Tracking-System (WTS) und entwirft daher ein erstes Anwendungsfalldiagramm. Dieses enthält registrierte Nutzer (*members*), Biologen (*biologist*) und Taucher (*diver*) des WTS, sowie grundlegende Anwendungsfälle. Diese sind im folgenden Anwendungsfalldiagramm zusammengefasst und in der nachfolgenden Tabelle kurz erläutert.



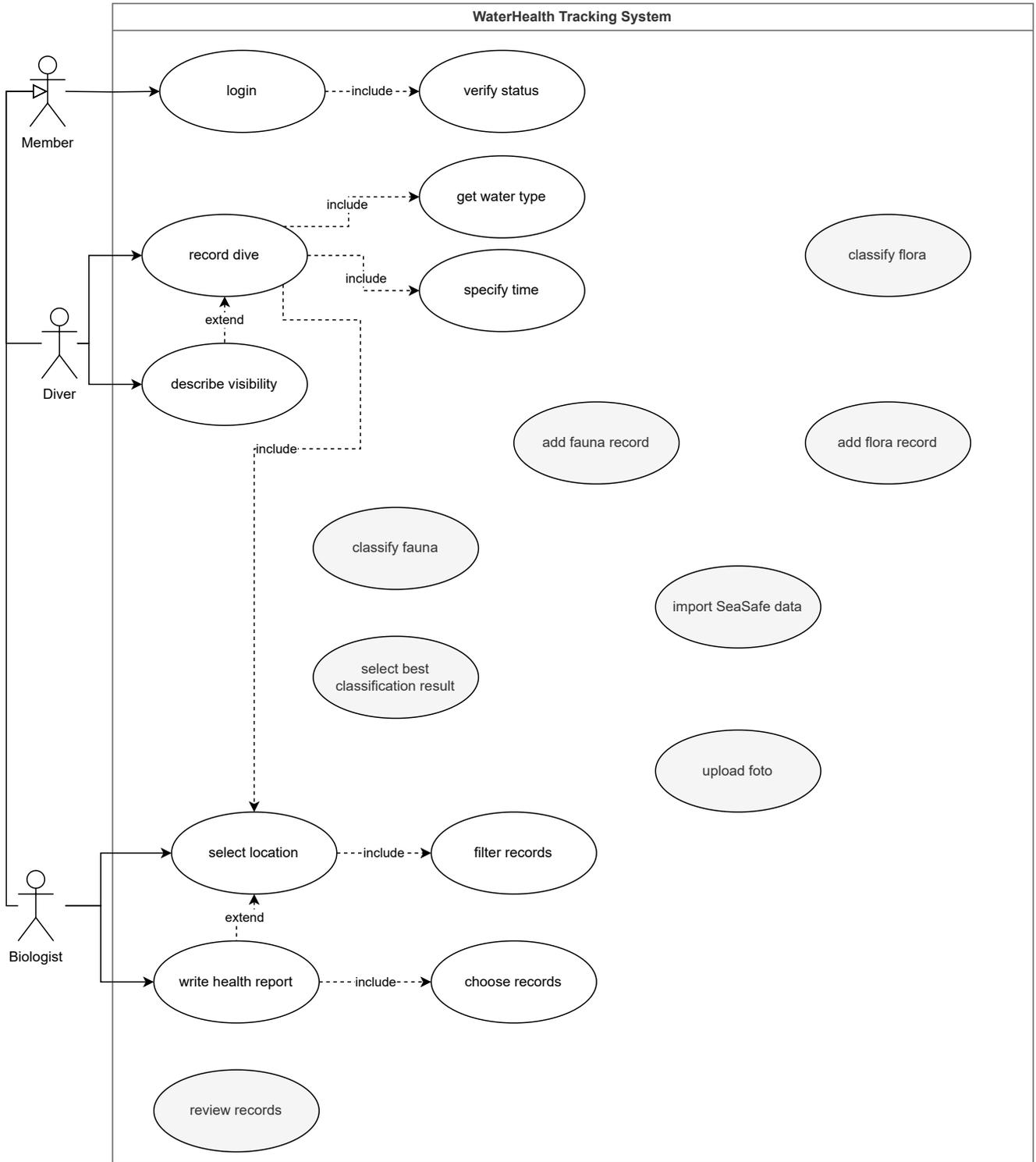
Anwendungsfall	Beschreibung
<i>login</i>	Anmelden des Nutzers im Tracking System
<i>verify status</i>	Automatisches Verifizieren des Status eines eingeloggten Nutzers
<i>record dive</i>	Anlegen einer Tauchgang-Aufzeichnung für ein spezifisches Gewässer
<i>get water type</i>	Eintragen des Wassertyps basierend auf dem spezifizierten Gewässer
<i>specify time</i>	Eintragen des Zeitpunkts vom Tauchgang durch den Nutzer
<i>describe visibility</i>	Beschreibung der Sichtweite Unterwasser
<i>select location</i>	Auswahl eines Gewässers durch den Nutzer
<i>filter records</i>	Ausfiltern der zum Gewässer zugehörigen Aufzeichnungen
<i>write health report</i>	Schreiben eines Berichts zur Gesundheit des Gewässers
<i>choose records</i>	Auswählen der Belege zum Gesundheitsbericht durch den Nutzer

- a) Klassifizieren Sie jeden Anwendungsfall im obigen Anwendungsfalldiagramm (use case diagram) entweder als Nutzerziel (*user goal*) mit U, als Teilfunktion (*subfunction*) mit S oder als Operation (*operation*) mit O. Tragen Sie hierzu die Klassifikation zwischen den Klammern « ____ » ein. (5 Punkte)

Die Entwickler von *PhischIT* schicken das obige Anwendungsfalldiagramm an die Stakeholder. Daraufhin erhalten sie folgende Rückmeldung:

Im obigen Anwendungsfalldiagramm sind ein paar Anforderungen vergessen worden. Taucher sollten beim Anlegen einer Tauchaufzeichnung (*record dive*) die Option bekommen, Fauna und Flora zu dokumentieren (*add fauna record, add flora record*). Für eine Dokumentation von Fauna und Flora (*add fauna record, add flora record*) soll ein Foto der Spezies hochgeladen werden (*upload foto*). Zusätzlich werden die Daten des *SeaSafe*-Messinstruments benötigt, um die lokale Wasserqualität im Lebensraum zu erfassen. Die Daten sollen dementsprechend importiert und in die Aufzeichnung geladen werden (*import SeaSafe data*). Um zukünftigen Berichterstatlern die Arbeit zu erleichtern, müssen die fotografierten Spezies mithilfe einer automatischen Bilderkennung und den gegebenen Umgebungsdaten klassifiziert werden (*classify fauna, classify flora*). Insbesondere bei der Klassifikation von Fauna kann es häufig zu Fehlern kommen. In diesem Fall sollen Tauchern die besten Ergebnisse vorgeschlagen werden, sodass diese das zutreffende Ergebnis auswählen können (*select best classification results*). Abgesehen davon fehlt Biologen vor dem Schreiben eines Berichts (*write health report*) die Möglichkeit, Aufzeichnungen einfach nur ansehen zu können (*review records*).

- b) Vervollständigen Sie nun das Anwendungsfalldiagramm (use case diagram) auf der nächsten Seite genau so, dass es die Anforderungen im obigen Text beschreibt. Zeichnen Sie ausschließlich die Beziehungen von Aktoren (actors) zu Anwendungsfällen und zwischen Anwendungsfällen ein, die im Text beschrieben werden. Zeichnen Sie keine weiteren Anwendungsfälle oder Aktoren ein. (7 Punkte)



Name:

Matrikelnummer:

- c) Verwenden Sie das textuelle Anwendungsfallbeschreibungsschema *Fully-Dressed* zur vollständigen Spezifikation des Erfolgsfalls „Hinzufügen eines Tierfotos zur Aufzeichnung des Tauchgangs“ (*add fauna record*). Nutzen Sie hierfür die unten gegebene Schablone. Beschreiben Sie zusätzlich eine Erweiterung. (4 Punkte)

Name d. Anwendungsfalls

Use Case Name

Erstellen eines Tauchgangs

Zielebene

Goal Level

Nutzerziel (User Goal)

Primäraktor(en)

Primary Actors

Taucher

Stakeholders

Vorbedingungen

Preconditions

Taucher, Biologe

Nachbedingungen

Postconditions

Primäres Erfolgszenario

Main Success Scenario

Erweiterung (Weiterer Ablauf)

Extension; Additional Flow

Name:

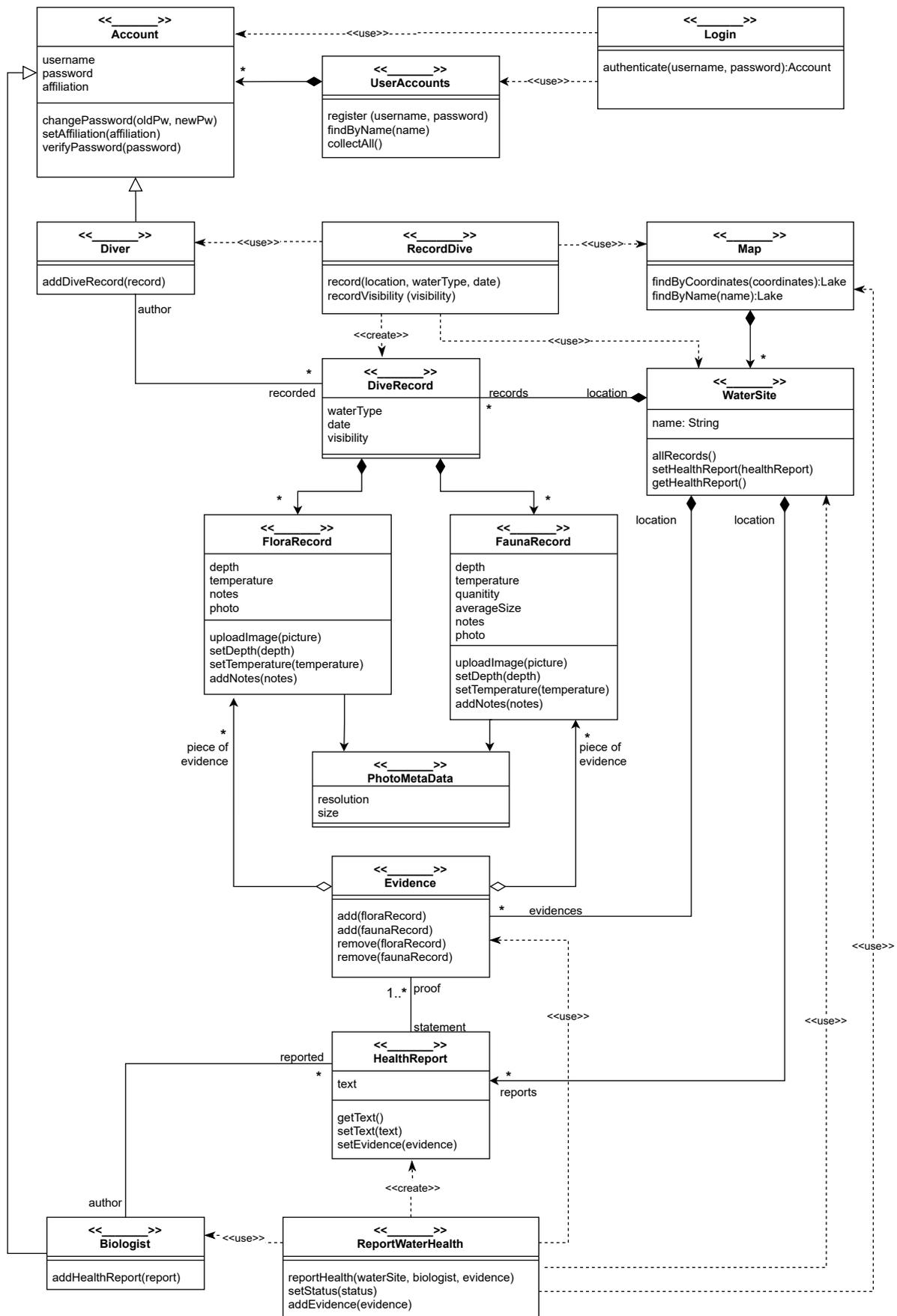
Matrikelnummer:

Aufgabe 4: Domänengetriebener Entwurf (9 Punkte)

Die folgende Aufgabe widmet sich der Anwendung des Domänen-getriebenen Entwurfs für das Wassergesundheits-Tracking-System (WTS). Dieses erlaubt Tauchern das Aufzeichnen von gesichteten Pflanzen und Tieren. Basierend auf diesen Aufzeichnungen können Biologen die Gesundheit der Seen beurteilen und Berichte erstellen. Um mit dem WTS zu interagieren, müssen sich beide Nutzergruppen zunächst anmelden.

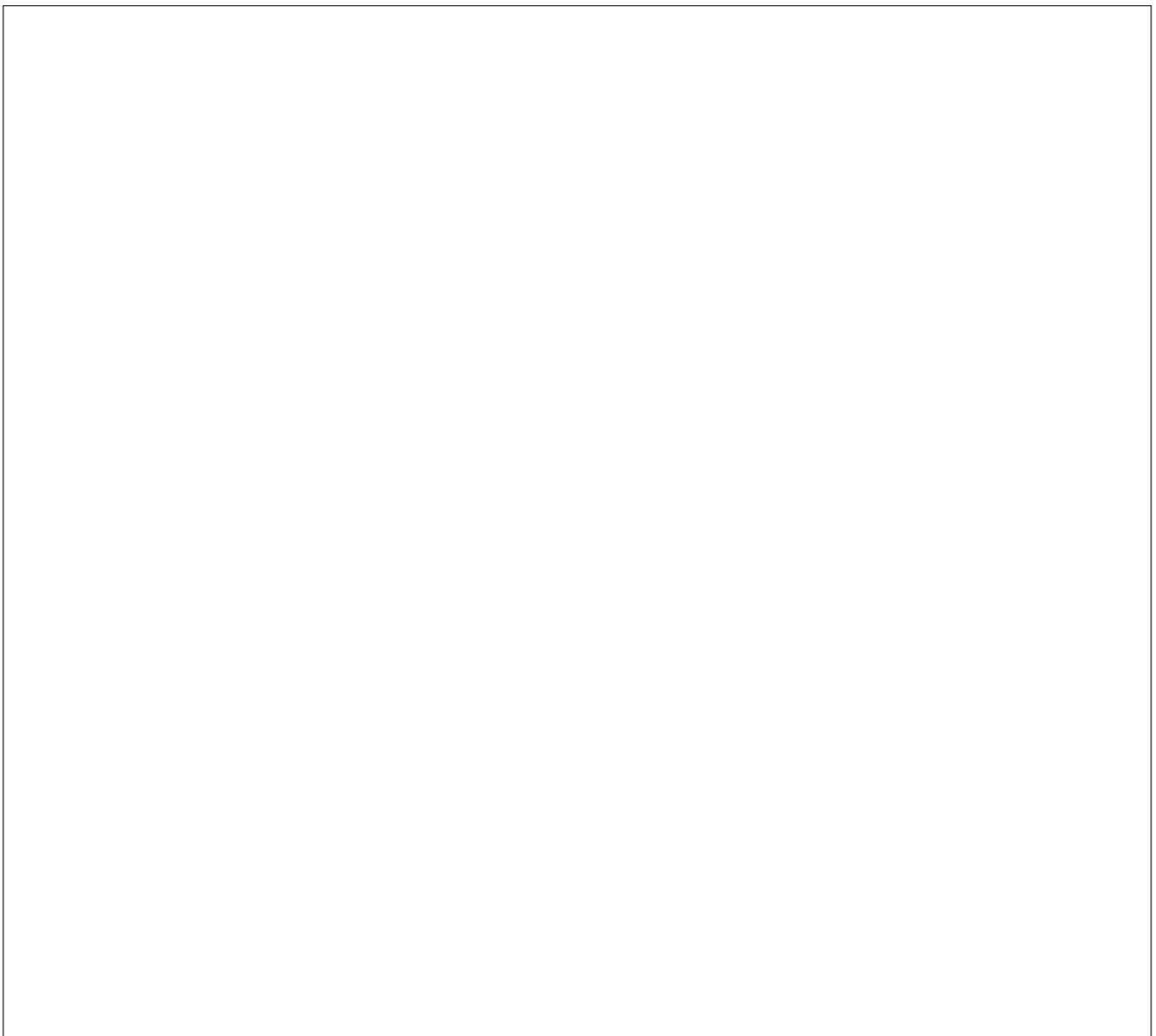
- a) Durch die Domänenanalyse des Trackingsystems ist das Klassendiagramm auf der nächsten Seite entstanden. Klassifizieren Sie jedes Domänenkonzept entsprechend der Bausteine (building blocks) des Domänen-getriebenen Entwurfs entweder als `ValueObject` mit V0, `Entity` mit E, `Service` mit S, `Factory` mit F, `Repository` mit R oder `Aggregate` mit A. Tragen Sie hierzu die Klassifikation zwischen den Klammern «____» ein. (7,5 Punkte)
- b) Nennen Sie in zwei Stichpunkten Gründe für Ihre Klassifikation der Domänenklasse `DiveRecord`. (1 Punkt)

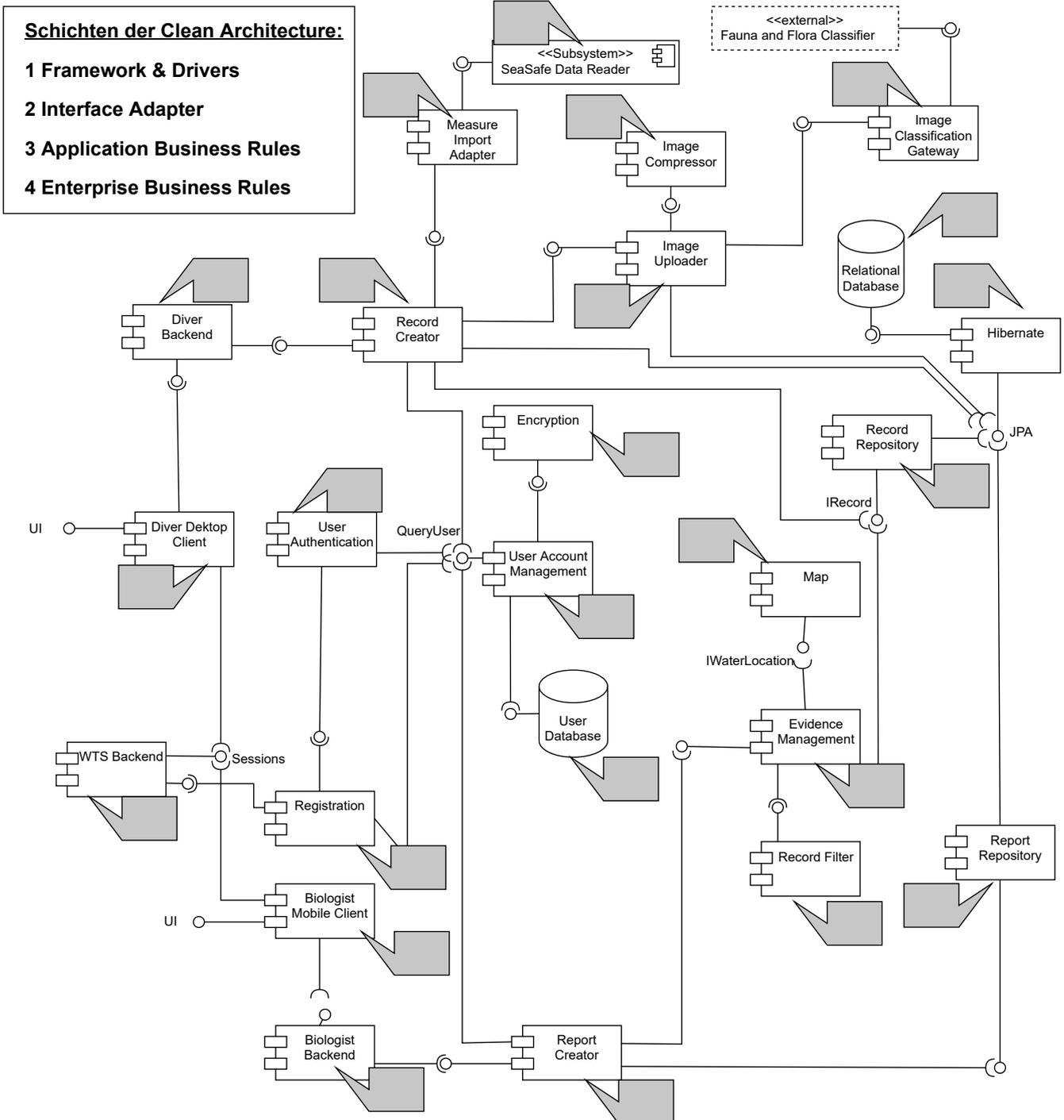
- c) Welche Invariante muss in `Evidence` sichergestellt werden? (0,5 Punkte)



Aufgabe 5: Saubere Unternehmensarchitekturen (20 Punkte)

- a) Das Komponentendiagramm auf der nächsten Seite stellt die Softwarearchitektur des WaterHealth-Tracking-Systems (WTS) dar. Ordnen Sie nun jede Komponente einer Schicht der sauberen Architektur zu, indem Sie die Nummer der Schicht in die grauen Sprechblasen eintragen. (12 Punkte)
- b) Erfüllt das gegebene Komponentendiagramm den Stil einer Clean Architecture? Begründen Sie kurz ihre Antwort anhand des Diagramms. (2 Punkte)





Name:

Matrikelnummer:

- c) Identifizieren Sie sechs unterschiedliche Arten von Verstößen gegen die Praktiken des sauberen Programmierens (Clean-Code-Prinzipien und Coding-Conventions aus der Vorlesung) und zugehörige Zeilen im Quelltextbeispiel (nächste Seite), Formattierung ausgenommen. Benennen Sie jeweils die Art des Verstoßes mit der zugehörigen Zeilennummer. (6 Punkte)

```

1 package model;
2 public abstract class Record {
3     private final String waterSiteName;
4     // location should be in coordinates
5     private String location;
6     private boolean verified = false;
7     private RecordManager manager;
8
9     public Record( String waterSiteName, String location, RecordManager manager ){
10         this.waterSiteName = waterSiteName; this.location = location; this.manager = manager;
11         if ( manager != null ) manager.store(this);
12     }
13
14     public String getWaterSite() { return waterSiteName; }
15
16     public String getLocation() { return location;}
17
18     public boolean isFaunaRecord() { return false; }
19
20     public boolean isVerified() {
21         verified = true;
22         return verified;
23     }
24
25     public void setVerified(boolean verified) { this.verified = verified; }
26 }

```

```

27 package model;
28 public class RegularRecord extends Record {
29     public RegularRecord(String waterSiteName, String location, RecordManager manager) {
30         super(waterSiteName, location, manager);
31     }
32 }

```

```

33 package model;
34 public class FaunaRecord extends Record {
35     public FaunaRecord(String waterSiteName, String location, RecordManager manager) {
36         super(waterSiteName, location, manager);
37     }
38
39     public void setVerified(boolean verified) { verified=true; /* aber ist es ok? */ }
40
41     public boolean isFaunaRecord() { return true; }
42 }

```

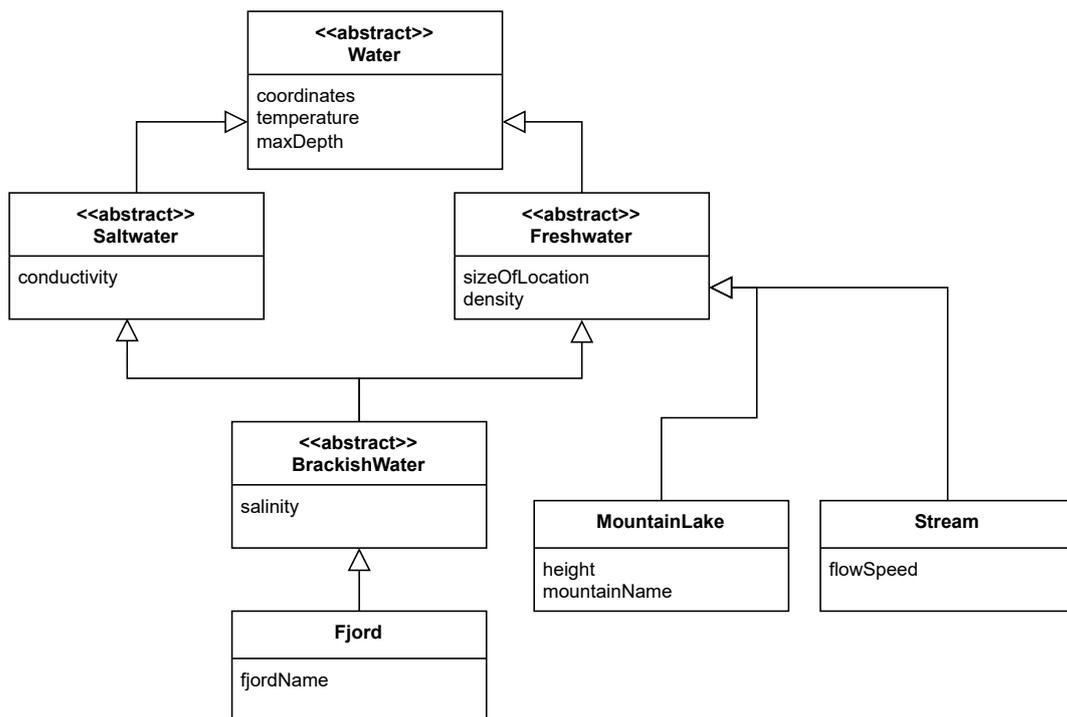
```

43 package controller;
44
45 public class RecordManager {
46     private List<RegularRecord> records = new ArrayList<>();
47     private List<FaunaRecord> faunaRecords = new ArrayList<>();
48
49     public void storeRecord(Record record) {
50         if (record != null) return;
51         if (record instanceof FaunaRecord) faunaRecords.add(record);
52         if (record instanceof RegularRecord) records.add(record);
53     }
54
55     public boolean selectRecords(String waterSiteName, String location) {
56         //TODO: [SC] will write this: Find records by waterSiteName and check location */
57     }
58
59     public void createReport(Record record, String text) {
60         Report report = new Report();
61         report.waterSite.setName(record.getWaterSite());
62         report.addDescription(text);
63     }
64 }

```

Aufgabe 6: Entwurfsmuster für Unternehmenssoftware (14 Punkte)

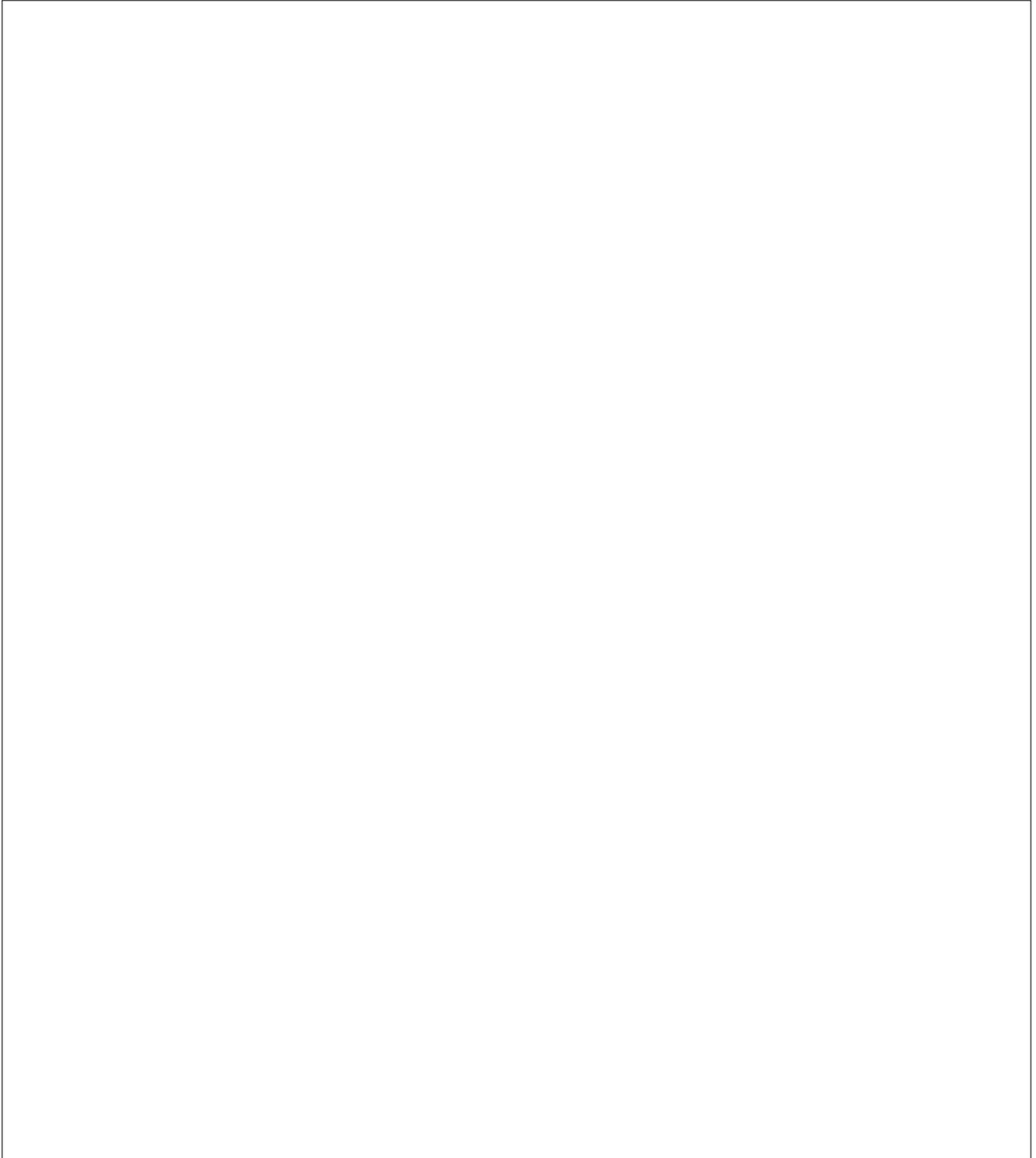
In der Geschäftslogik des WaterHealth-Tracking-Systems gibt es unter anderem die folgende Klassenhierarchie für mögliche Wassertypen, die von den SeaSafe-Geräten erkannt werden. Um diese Klassenhierarchie in einer relationalen Datenbank abzubilden, sollen Sie in dieser Aufgabe die unterschiedlichen Objekt-Relationalen Strukturmuster (object-relational structural patterns) anwenden, welche in der Vorlesung behandelt wurden.



Name:

Matrikelnummer:

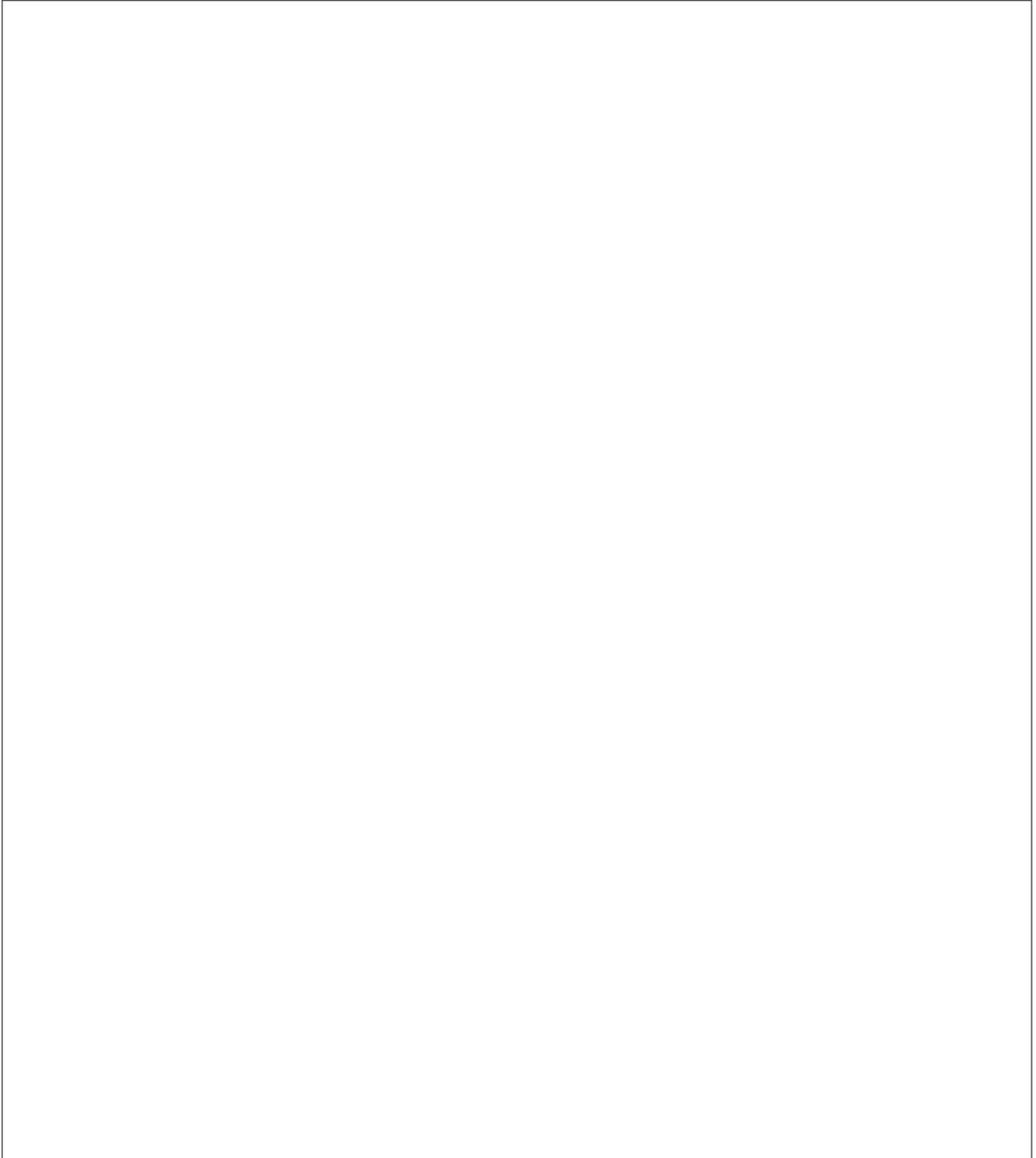
- a) Erstellen Sie ein Datenbankschema nach dem Strukturmuster Single Table Inheritance. Erstellen Sie hierzu ein UML-Klassendiagramm der notwendigen Tabellen (dargestellt als UML-Klasse mit dem «table» Stereotyp) und ihrer Attribute. Führen Sie, falls nötig, neue Attribute ein, um Primärschlüssel und Vererbung darzustellen. (4 Punkte)



Name:

Matrikelnummer:

- b) Erstellen Sie ein Datenbankschema nach dem Strukturmuster Class Table Inheritance. Erstellen Sie hierzu ein UML-Klassendiagramm der notwendigen Tabellen (dargestellt als UML-Klasse mit dem «table» Stereotyp) und ihrer Attribute. Führen Sie, falls nötig, neue Attribute ein, um die Primärschlüssel und Vererbung darzustellen. (7 Punkte)



Name:

Matrikelnummer:

- c) Stellen Sie stichpunktartig Single Table Inheritance, Concrete Table Inheritance und Class Table Inheritance gegenüber. Nennen Sie zu jeder Technik einen Vor- und Nachteil. (3 Punkte)