

Datenbankanwendung zu seltenen Haustierkrankheiten

Bachelorarbeit

Übersicht des Vortrags

- Anforderungen & Motivation
- Wahl der Werkzeuge
- Agile Entwicklung
- Entwicklungsverlauf
- Wichtige Designentscheidungen
- Probleme und Zeitabschätzungen

- Verbesserungen und Fortführung

Motivation

- Es gibt kaum vergleichbare Anwendungen
- Häufig nicht auswertbare Daten
- EVSSAR erhält häufig Fälle zu seltenen Krankheiten

Anforderungen

- Datenbankanwendung die auf verschiedenen Geräten funktioniert
- Individuelle Formulare für jede Krankheit
- Möglichst niedrige Kosten
- CRUD
- Wünschenswert wäre eine Möglichkeit **neue Formulare zu erstellen**
- Administrationssystem

Wahl der Werkzeuge

Erfahrung des „Entwicklerteams“:

- Relationale Datenbanken.
- Wenig PHP, Python
- Kein Ruby, Perl

MariaDB

- Erfahrung aus Studium (Mysql)
- Lizenz

PHP als Sprache:

- Geringer Aufwand Ersatz zu finden
- Vielfältige Hilfe und Erfahrung

Framework Codeigniter:

- Einfacher Einstieg
- Gute Dokumentation
- Selbsttest erfolgreich

Agiler Prozess

- Regelmäßige treffen
- Präsentation des aktuellen Standes
- Planung der nächsten Funktionen mit Priorisierung
- Zusätzlich Definition von Füllstoff.
- Um auf Verbesserungsvorschläge reagieren zu können.
- Echte Zeitabschätzung erschien sinnlos.
- Falls die Priorisierungsliste zu schnell abgearbeitet werden könnte.

Entwicklungsverlauf

Erstellung der Basisfunktionen

- Login
- Erlernen des Frameworks anhand der Implementation der ersten Funktionen.

Zweite Iteration

- Umsetzung EAV
- Administrationslogik

Erste Iteration

- Lösungsfindung für das Problem der Unabhängigkeit der Anwendung.
- „Database Forge“
- EAV

Dritte und vierte Iteration

- Implementation fehlender Funktionen.
- Vierte als Fehlerverbesserung
Umgestaltung des Freigabeprozesses

Report Vs. Form

Trennung der Controller nach Funktion und benötigten Zugriffsrechten

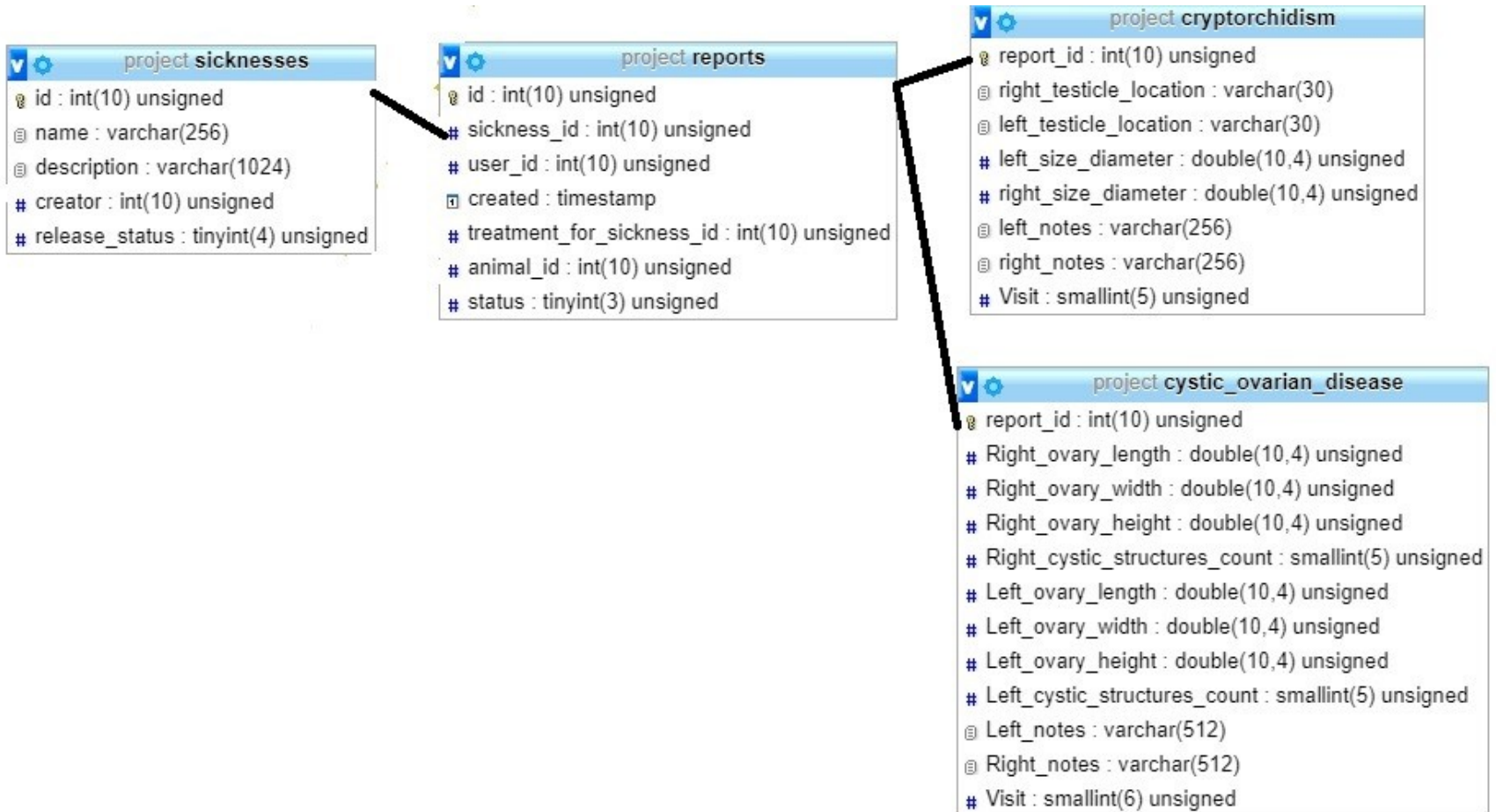
Form Controller um neue Formulare zu erstellen und einige andere Administrationsfunktionen

Report Controller um Fälle zu erstellen und zu verwalten

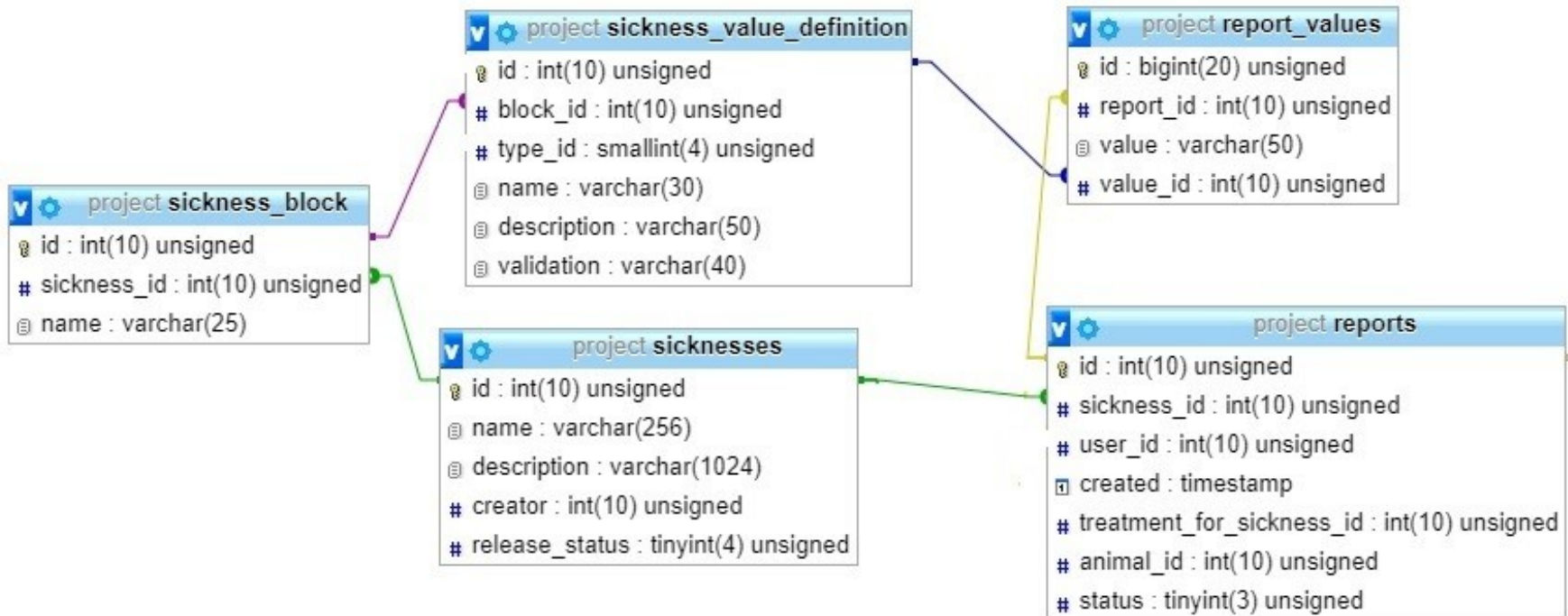
Entity - Attribute - Value

- Umstrittenes Design
- Lösung geeignet für kleine Datensätze
- Vergleichsweise komplexe Datenbankabfragen
- Verbreitet in Datenbanken der Medizin

Relational



EAV Beispiel



Resultat

- Erfüllt die definierten Anforderungen
- EAV als Lösung
- Aufgrund der funktionsorientierten Entwicklung noch viele Optimierungsmöglichkeiten

Einige Beispiele:

1. HTTPS
2. Vorschaufunktion unnötig?
3. Visuelle Verbesserungen
4. Inkonsistenz zwischen den Bezeichnungen DB, Code und Anzeige
5. Besseres Nutzermanagement (PW Reset ,...)
6. Automatisierte Tests

Schwierigkeiten

- Möglichst autonome Anwendung
- Htaccess
- Serverprobleme
- Dropdown

Fortführung

- Formularerstellung
- Nutzertests

Vielen Dank