Internetserver-programmierung

Projektbericht im Rahmen der Portfolio-Prüfung SoSe 2020

**2. Zeitraum: 06.07.2020 – 02.08.2020**

**Gruppe 16**

**Gruppenmitglied: Felicitas Yeboah**

**Gruppenmitglied: Christian Till**

**SEITE FÜR FREITEXT - MUSS VOR ABGABE GELÖSCHT WERDEN!**

**Issues/Fragen: -> Meinungen und Hinweise dazu gerne kundtun! 😉**

* **Soll ein Termin mehrere Kategorien haben können?**

**wenn ja -> zusätzliche Datenbanktabelle für m:n-Beziehung nötig**

* **Sind eigene Klassen zur Kommunikation mit der Datenbank sinnvoll?**

**Dann könnte man das Entwurfsmuster „Fassade“ verwenden. Dies entspricht im Grunde einer Klasse „Datenbank“, welche dann abhängig von der Art der Datenbank agieren kann … allerdings haben wir ja nur eine Datenbank und dann ist das wohl zu viel. Trotzdem sollte es eine Klasse für ein Datenbank-Objekt geben, welche dann entsprechend die DB-Zugriffe übernimmt.**

* **Nutzen von PDO oder MySQLi?**

**Im Skript nur MySQLi, aber im Kurs-Forum hat sie auch auf PDO verlinkt. Daher scheint beides in Ordnung zu sein.**

[**https://websitebeaver.com/php-pdo-vs-mysqli**](https://websitebeaver.com/php-pdo-vs-mysqli)**:**

**Aufgrund des dort genannten Vorteils für PDO:**

**„**Allow to pass variables and values directly into execute“

**würde ich mich dafür entscheiden.**

**Ich stelle mir den Ablauf nämlich folgendermaßen vor.**

**Es gibt eine Klasse für das Objekt „Termin“ und eine Klasse für die Datenbankanbindung.**

**Die Datenbank-Klasse implementiert so eine Art REST-Server und beinhaltet alle Funktionen, welche für die Interaktion mit der Datenbank benötigt wird, z.B.**

* + **Lese Termin für Monat**
  + **Lese Termine für Kategorie xy**
  + **Schreibe Termin**

**Die Klasse Termin hält wiederum in den Membern die Daten des Termins (bis diese an die Datenbank übergeben werden). Und bietet unter anderem eine Funktion, welche den Termin, aufbereitet in HTML-Code, ausgibt.**

1. **Spezifikation**
   1. **Benutzergruppen**
   2. **Funktionale und nicht-funktionale Anforderungen**
2. **Entwurf**
   1. **Klassendiagramm**
   2. **Entwurfsmustereinsatz**

**Observer**

Bei Änderung eines Termins wird die Anzeige des Kalenders aktualisiert.

**Fabrik**

Implementiert einen Termin, der sich in der Kategorie unterscheidet oder Erzeugung eines Objekts für die Datenbank.

1. **Implementierung**
   1. **Screenshots**
   2. **Genutzte Frameworks**
2. **Zusammenfassung**
   1. **Aufgabenteilung**
   2. **Ausblick**