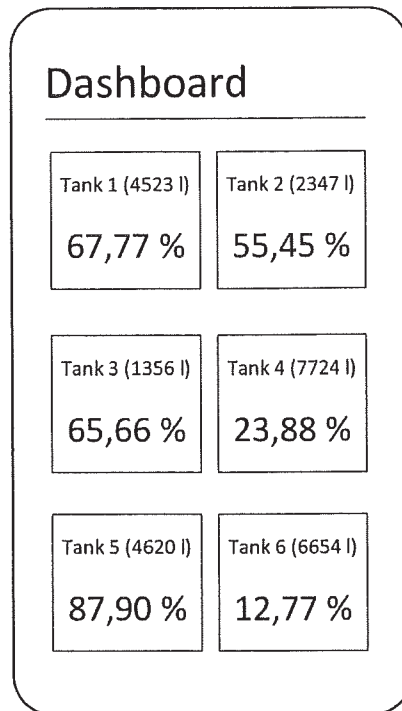


3. Handlungsschritt (25 Punkte)

Korrekturrand

Eine Smartphone-Anwendung mit unterschiedlichen Anzeigemöglichkeiten für Tankfüllstände soll entwickelt werden.



In einer ersten Implementierung soll eine „Dashboard“-Anzeige, welche die Füllstände aller Tanks anzeigt und später eine „History“-Anzeige, die den zeitlichen Verlauf des Füllstands eines Tanks darstellt, erstellt werden.

a) Zunächst soll eine Klasse *Tank* für Tankobjekte modelliert werden.

aa) Die Klasse *Tank* soll Folgendes beinhalten:

- Die nur klassenintern sichtbaren Instanzvariablen *bezeichner*, *fuellstand*, *fassungsvermoegen*
- Einen öffentlichen Konstruktor zur Initialisierung der Instanzvariablen.
- Beispielhaft für den *fuellstand* je eine öffentliche Set- und Get-Methode.

Hinweis: Geben Sie jeweils sinnvolle Datentypen an.

Erstellen Sie das UML-Klassendiagramm für die Klasse *Tank*.

7 Punkte

A large empty rectangular box intended for drawing the UML class diagram for the *Tank* class.

ab) Implementieren Sie in Pseudocode die Set- und Get-Methode für die Instanzvariable *fuellstand* der Klasse *Tank*. 4 Punkte

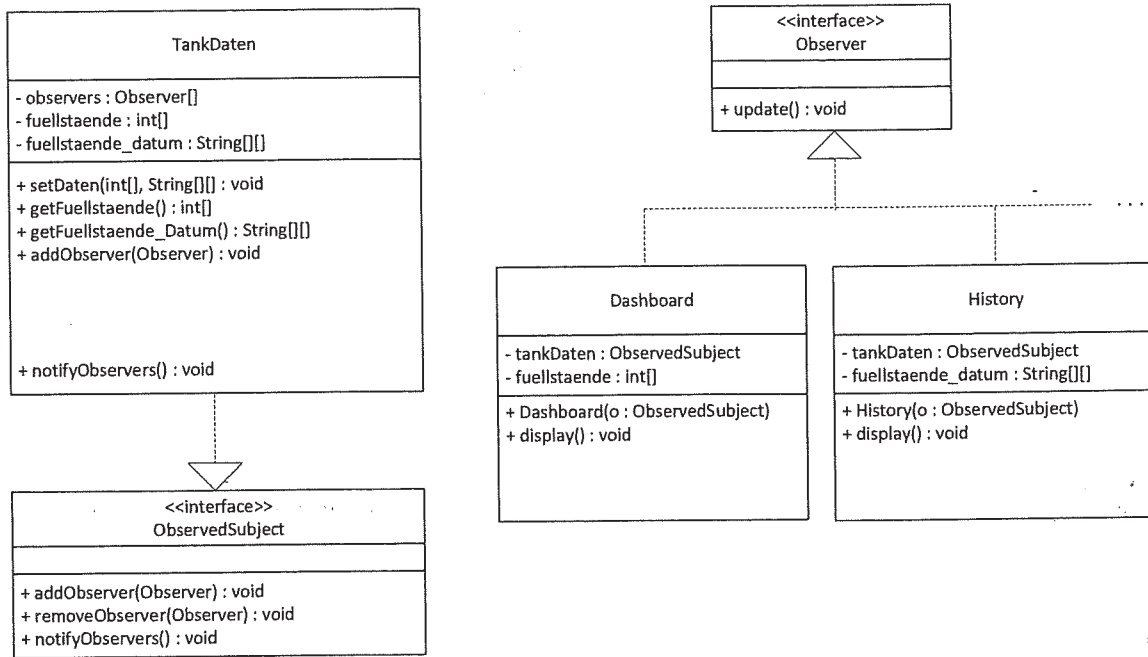
A large empty grid of 20 columns and 15 rows intended for writing the pseudocode for the Set and Get methods of the *fuellstand* variable.

Fortsetzung 3. Handlungsschritt →

Fortsetzung 3. Handlungsschritt

Korrekturrand

- b) Die Füllstände der Tanks werden stündlich aktualisiert. Alle Anzeigen sollen entsprechend angepasst werden. Ein noch unvollständiger Entwurf nach dem Observer-Muster zur Umsetzung dieser Anforderung liegt bereits vor.



- ba) Ergänzen Sie im obigen Klassendiagramm in den Klassen *TankDaten*, *Dashboard* und *History* die jeweils fehlende Implementierung und die Beziehung zwischen der Klasse *Tankdaten* und dem Interface *Observer*. 4 Punkte
- bb) Erläutern Sie die Art der Beziehung zwischen den Klassen *Tankdaten* und *ObservedSubject*. 2 Punkte