

Softwaretechnik

Aufgabenblatt 8

A1 Mobile Application

Szenario

Sie wollen objektorientiert ein hybrides Application Framework entwickeln, mit dem Sie einheitlich graphische Benutzerschnittstellen in Android und iOS steuern können. Sie benötigen die folgenden Klassen:¹

- Eine generalisierte Klasse **MobileElement** mit dem schreibgeschützten Attribut **title** vom Datentyp **String** sowie den rückgabellosen Operationen **render()** und **register(MobileForm form)**.
- Eine generalisierte Klasse **MobileForm** mit folgenden Klassenmitgliedern:
 - Liste der enthaltenen mobilen Elemente vom Typ **MobileElement**. Das Verhältnis ist eine Teile-Ganze-Beziehung.
 - Operation **addElement(string title)**: es wird ein neues mobiles Element mit dem angegebenen Titel erzeugt. Der Titel kann nur bei der Erzeugung festgelegt werden. Während der Erzeugung wird die zugehörige **MobileForm** registriert.
 - Operation **render()**, die die gleichnamige Operation in allem enthaltenen mobilen Elementen aufruft.
- Von **MobileElement** und **MobileForm** sollen keine Instanzen erzeugt werden können.
- Konkrete Elemente und konkrete Formen für Android und iOS, die erzeugt werden können. Die Betriebssysteme einer konkreten Form und den darin enthaltenen konkreten Elementen passen immer zusammen.
- Alle oben genannten Klassen sind im Paket **mobileapplication.ui** enthalten. Aus Sicherheitsgründen sollen neue konkrete Elemente von außerhalb des Pakets nur über die Operation **addElement** von **MobileForm** erzeugt werden können.
- Die Klasse **App** im Paket **mobileapplication** mit den folgenden Eigenschaften:
 - Attribut **form** vom Datentyp **MobileForm**.
 - Private Operation **getOS()**, die das aktuelle Betriebssystem als String zurückgibt. Mögliche Werte sind zurzeit »Android« und »iOS«.
 - Private Operation **initialize()**, die dem Attribut **form** eine neue Instanz passend zum Betriebssystem zuweist.
 - Operation **execute()**, die **initialize()** aufruft, drei neue Elemente der konkreten Form hinzufügt und anschließend alle Elemente rendert.

Aufgaben

- [a] Entwerfen Sie eine Lösung als Klassenverband. Nutzen Sie dafür das Entwurfsmuster Factory Method, wobei **MobileForm** die Rolle von »Creator« und **MobileElement** die Rolle von »Product« übernehmen soll. Modellieren Sie den Klassenverband als UML-Klassendiagramm.
- [b] Modellieren Sie die Abfolge der Aufrufe für Android als UML-Sequenzdiagramm. Beginnen Sie mit der Operation **execute()** der Klasse **App** als gefundene Nachricht. Nutzen Sie dabei die konkreten und nicht die abstrakten Klassen.
- [c] Implementieren Sie den Rumpf der Lösung in Java.

Hinweis: Pakete und die darin enthaltenen Klassen werden als UML-Paketdiagramm modelliert, so wie es in Abbildung 1 dargestellt ist.

¹ Wenn nicht anders angegeben, dann sind die genannten Klassenmitglieder öffentlich.

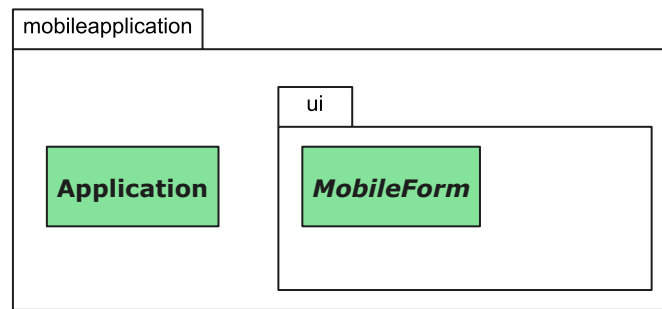


Abbildung 1: UML-Paketdiagramm (Package Diagram)