Berufsprüfung ICT-Applikationsentwicklung

Informationen zur Prüfung

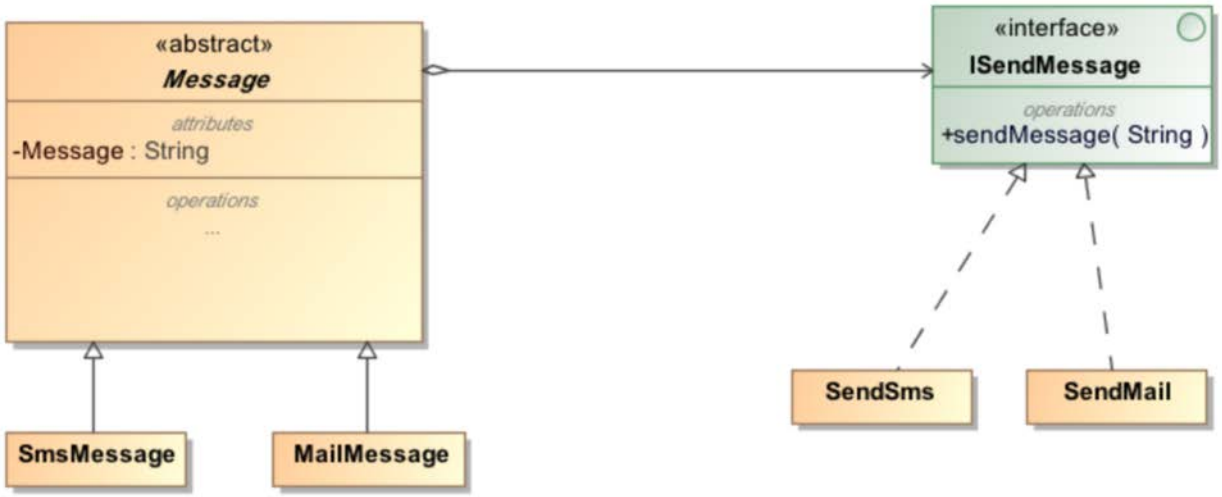
- Bei jeder Aufgabe sind die Punktzahlen angegeben, die Sie für die korrekte Beantwortung der jeweiligen Aufgabe maximal erhalten. Die Punktzahl entspricht gleichzeitig auch der Richtzeit für die Bearbeitung (Minutenpunkte).
- Schreiben Sie in gut lesbarer Schrift. Unleserliches wird nicht korrigiert und nicht bewertet.
- Korrekturen in Ihrer Lösung müssen eindeutig als solche erkennbar sein (z.B. mittels Durchstreichen). Mehrdeutige Lösungen werden nicht korrigiert und nicht bewertet.
- Für Ihre Lösung steht Ihnen bei jeder Aufgabe ein entsprechendes Feld zur Verfügung. Texte oder Skizzen ausserhalb der Feldbegrenzungen werden nicht korrigiert und nicht bewertet. Für umfangreiche Korrekturen stehen Ihnen am Ende des Prüfungshefts Korrekturblätter zur Verfügung. Verweisen Sie bei der Verwendung von Korrekturblättern im Lösungsfeld einer Aufgabe klar auf das Korrekturblatt.
- Formale Vorgaben und Begrenzungen an die Lösung einer Aufgabe (z.B. "in 3 Sätzen" oder "mit max. 5 Stichworten") sind verbindlich. Abweichende Lösungsformen oder überzählige Antworten werden nicht korrigiert und nicht bewertet, wobei bei der Korrektur mit den erstgenannten Elementen begonnen wird.
- Ihre Antworten müssen einen konkreten Bezug zum Fallbeispiel haben. Wenn beispielsweise nach Massnahmen oder nächsten Schritten gefragt wird, genügen allgemeingültige Globalantworten wie "Planen" oder "Kommunizieren" nicht.

Implementieren eines Design-Pattern

Ausgangslage

Sie arbeiten in der Softwareentwicklung einer kleinen Firma, welche Kommunikationslösungen verschiedener Art implementiert. Für das zentrale Kommunikationsmodul haben Sie einen Ausschnitt eines Klassendiagramms erhalten, welches die Klassen für das Versenden einer Textmeldung darstellt.

Das dargestellte, unvollständige Klassendiagramm für das Versenden einer Textmeldung basiert auf dem XY-Pattern.



Aufgabe 1	**16 Punkte**

Identifizieren Sie basierend auf dem Klassendiagramm in der Ausgangslage die (Prototypen der) Methoden in der Klasse ‚Message'.

Antwortstruktur
Spezifizieren Sie in der vorgegebenen Tabelle im Antwortfeld für jede identifizierte Methode den Methodennamen, den Typ des Rückgebewerts, den oder die Parameter und die Sichtbarkeit der Methode.

Antwortfeld

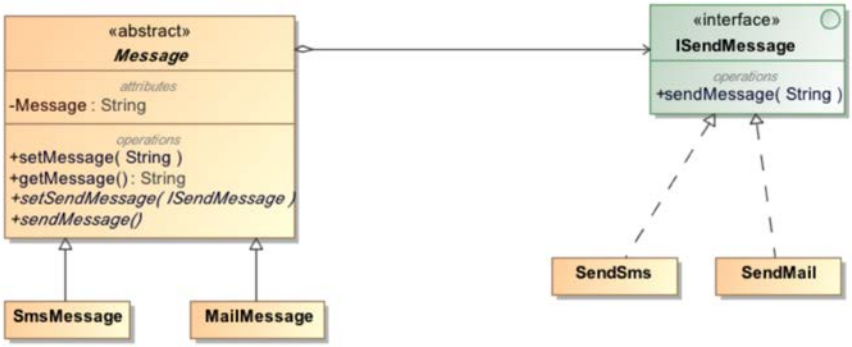
Methodenname	Typ des Rückgabewert (return type)	Parameter	Sichtbarkeit (Modifier)

Korrektur (leer lassen)

Bemerkungen zur Korrektur (leer lassen)

Aufgabe 2	**15 Punkte**
Programmieren Sie die Klasse "SmsMessage" und deren Methoden aus dem Klassendiagramm aus. Notieren Sie dazu korrekte Syntax in Java oder C#. Auf die Deklaration des Imports von Packages oder Klassen und den Konstruktoren kann dabei verzichtet werden.	
Antwortstruktur Korrekt programmierte Klasse, mit Variablen, und Methoden.	
Antwortfeld	*Korrektur (leer lassen)*
Bemerkungen zur Korrektur (leer lassen)	

Lösung Aufgabe 1 — 16 Punkte



[→ Kopfzeile wie in Aufgabenstellung anpassen]

Methodenname	Typ des Rückgabewert (return type)	Parameter	Sichtbarkeit (Modifier)
setMessage	void	String	public
getMessage	String		public
setIsendMessage	void	ISendMessage	public abstract
sendMessage()	void		public abstract

Bewertungsvorgaben / Punkteschüssel

Für jede Metdhode 4 max.

- Korrekte Methodennamen 1
- Type des Rückgabewertes 1
- Parameter 1
- Modifier 1

Lösung Aufgabe 2	**15 Punkte**

In Java:

```
public class SmsMessage extends Message{

  private ISendMessage sendMessage = new SendSms();


  public void setSendMessage(ISendMessage sendMessage) {
    this.sendMessage = sendMessage;
  }

  public void sendMessage() {
    sendMessage.sendMessage(getMessage());
  }

}
```

In C#:

```
 public class SmsMessage : Message {

    private ISendMessage sendMessage = new SendSms();

    public void setSendMessage(ISendMessage sendMessage) {
      this.sendMessage = sendMessage;
    }

    public override void sendMessage() {
      sendMessage.sendMessage(getMessage());
    }

  }
```

Bewertungsvorgaben / Punkteschüssel

- Richtige Java oder C# Notation	5
- Klassendeklaration richtig:	
o Klassenname richtig	1
o Vererbung richtig	2
- Variable ISendMessage vorhanden	1
o Default Initialisierung vorhanden	1
- Methode setSendMessage vorhanden	1
o Variable richtig gesetzt	1
- Methode sendMessage vorhanden	1
o sendMessage.sendMessage(getMessage() richtig	2