

Review-Protokoll

May 22, 2012

Certified Tester Praktikum 2

Jens Schaa, Lucas Jenss, Denis Fleischhauer

1 Review Protokoll

Im folgenden erstellen wir ein Review Protokoll, welches den CoffeeMaker auf Konformanz zu den Anforderungen überpüft, die Usability sicherstellen, sowie die allgemeine Zuverlässigkeit des Systems überprüfen soll.

2 Anforderungen (Requirements)

2.1 Waiting State

When the Coffee Maker is not in use it waits for user input. There are six different options of user input: 1) add recipe, 2) delete a recipe, 3) edit a recipe, 4) add inventory, 5) check inventory, and 6) purchase beverage.

2.1.1 Abweichungen

- Statt `Add recipe` ist der erste Menüpunkt `Add a recipe`.
- Statt `Purchase beverage` ist der sechste Menüpunkt `Make coffee`.
- Es gibt einen zusätzlichen Punkt `Exit`, welcher in den Anforderungen vorgesehen ist.

2.1.2 Usability

- Von der Benutzerführung ist es wenig erstrebenswert den CoffeeMaker abzuschalten.

2.1.3 Zuverlässigkeit

- Das Menü wird immer korrekt angezeigt.

2.2 Add a Recipe

Only three recipes may be added to the CoffeeMaker. A recipe consists of a name, price, units of coffee, units of milk, units of sugar, and units of chocolate. Each recipe name must be unique in the recipe list. Price must be handled as an integer. A status message is printed to specify if the recipe was successfully added or not. Upon completion, the CoffeeMaker is returned to the waiting state.

2.2.1 Abweichungen

- Es können mehr als 3 Rezepte hinzugefügt werden (4 Rezepte).

2.2.2 Usability

- Die Meldung, dass ein Rezept nicht hinzugefügt werden konnte taucht erst auf, wenn man schon alle Eingabe getätigt hat. Die Meldung sollte vorher erscheinen.
- Der Benutzer sollte über das korrekte Eingabeformat für die Rezept-Attribute informiert werden.

- Es sollte eine sofortige Prüfung der Eingaben vorgenommen werden, sodass der Benutzer gegebenenfalls nach einem richtigen Wert gefragt werden kann.

2.2.3 Zuverlässigkeit

- UTF-8 (Unicode) wird nicht unterstützt, in einem von 4 Testläufen stürzte das Programm bei der Eingabe von Umlauten ab.

2.3 Delete a recipe

A recipe may be deleted from the CoffeeMaker if it exists in the list of recipes in the CoffeeMaker. The recipes are listed by their name. Upon completion, a status message is printed and the Coffee Maker is returned to the waiting state.

2.3.1 Abweichungen

- Die Rezepte werden nicht vollständig gelöscht. Es wird lediglich der Name gelöscht, der Listen-Eintrag bleibt auch beim erneuten aufrufen von `Delete a recipe` vorhanden, und man kann auch weiterhin ein solches gelöscht Rezept bestellen.

2.3.2 Usability

- Wenn man ein Rezept löschen möchte, aber keines existiert, wird man trotzdem aufgefordert die Zahl des zu löschenden Rezepts einzugeben. Egal welche Eingabe man tätigt, wird danach immer eine Fehlermeldung ausgegeben.

2.3.3 Zuverlässigkeit

- Gibt man einen Getränke-Index ein, welcher am Ende ein Null hat, wird keine Fehlermeldung angezeigt, auch wenn dieser ungültig ist.

2.4 Edit a recipe

A recipe may be edited in the CoffeeMaker if it exists in the list of recipes in the CoffeeMaker. The recipes are listed by their name. After selecting a recipe to edit, the user will then enter the new recipe information. A recipe name may not be changed. Upon completion, a status message is printed and the Coffee Maker is returned to the waiting state.

2.4.1 Abweichungen

- Die Eingaben werden nicht auf korrektes Format überprüft (siehe Add a recipe (Unterabschnitt 2.2))

2.4.2 Usability

- Der User sollte über anzugebende Formate informiert werden (siehe Add a recipe (Unterabschnitt 2.2))
- Die Überprüfung auf eine falsche Eingabe sollte sofort geschehen (siehe Add a recipe (Unterabschnitt 2.2))

2.4.3 Zuverlässigkeit

- Nachdem man ein Rezept editiert hat, wird der Name nicht mehr in der Liste angezeigt (das Rezept ist aber noch vorhanden).
- Gibt man einen Getränke-Index ein, welcher am Ende ein Null hat, wird keine Fehlermeldung angezeigt, auch wenn dieser ungültig ist.

2.5 Add Inventory

Inventory may be added to the machine at any time from the main menu, and is added to the current inventory in the CoffeeMaker. The types of inventory in the CoffeeMaker are coffee, milk, sugar, and chocolate. The inventory is measured in integer units. Inventory may only be removed from the CoffeeMaker by purchasing a beverage. Upon completion, a status message is printed and the CoffeeMaker is returned to the waiting state.

2.5.1 Abweichungen

- Es ist nicht möglich Zucker eine positive Mengen Zucker hinzuzufügen.
- Es ist möglich eine negative Menge Zucker hinzuzufügen, wodurch dann Zucker aus dem Inventar entfernt wird.

2.5.2 Usability

- In der Fehlermeldung wird nicht näher erklärt, warum das Hinzufügen fehlschlug.
- Es findet keine sofortige Typ-Prüfung statt, wenn ein falsches Format verwendet wird (siehe Add a recipe (Unterabschnitt 2.2)).

2.6 Check inventory

Inventory may be checked at any time from the main menu. The units of each item in the inventory are displayed. Upon completion, the Coffee Maker is returned to the waiting state.

Keine Probleme gefunden

2.7 Purchase Beverage

The user selects a beverage and inserts an amount of money. The money must be an integer. If the beverage is in the RecipeBook and the user paid enough money the beverage will be dispensed and any change will be returned. The user will not be able to purchase a beverage if they do not deposit enough money into the CoffeeMaker. A user's money will be returned if there is not enough inventory to make the beverage. Upon completion, the Coffee Maker displays a message about the purchase status and is returned to the main menu.

2.8 Abweichungen

- Beim Kauf eines Getränkes nimmt die Menge an Kaffee zu statt ab.
- Gibt man bei der Auswahl des Getränkes eine ungültige Zahl ein (ein Index der nicht existiert) stürzt das Programm ab. Da man vorher schon dazu aufgefordert wurde, Geld einzuwerfen (bevor die Korrektheit der Eingabe überprüft wurde), wird kein Rückgeld gegeben.
- Zwar ist es nicht möglich ein Getränk zu bestellen, wenn nicht mehr genug Zutaten im Inventar vorhanden sind, allerdings lautet dann die Fehlermeldung `Insufficient funds`, unabhängig davon ob man genug Geld eingeworfen hat oder nicht.

2.9 Usability

- Dem Benutzer wird kein Preis für das Getränk angezeigt
- Den Benutzer sollte aufgefordert werden eine Zahl einzugeben, da er möglicherweise nichts mit dem Begriff `positive integer` anfangen kann.

3 Anomalien

3.1 Datenfluss-Analyse

```
public synchronized int makeCoffee(int recipeToPurchase, int amtPaid) { //d(change);
int change = 0;

    // r(recipeToPurchase);
    if (getRecipes()[recipeToPurchase] == null) {

        // d(change); r(amtPaid);
        change = amtPaid;

    // r(recipeToPurchase); r(amtPaid);
    } else if (getRecipes()[recipeToPurchase].getPrice() <= amtPaid) {

        // r(recipeToPurchase);
        if (inventory.useIngredients(getRecipes()[recipeToPurchase])) {

            // d(change); r(amtPaid); r(recipeToPurchase);
            change = amtPaid - getRecipes()[recipeToPurchase].getPrice();
        } else {

            // d(change); r(amtPaid);
            change = amtPaid;
        }
    } else {

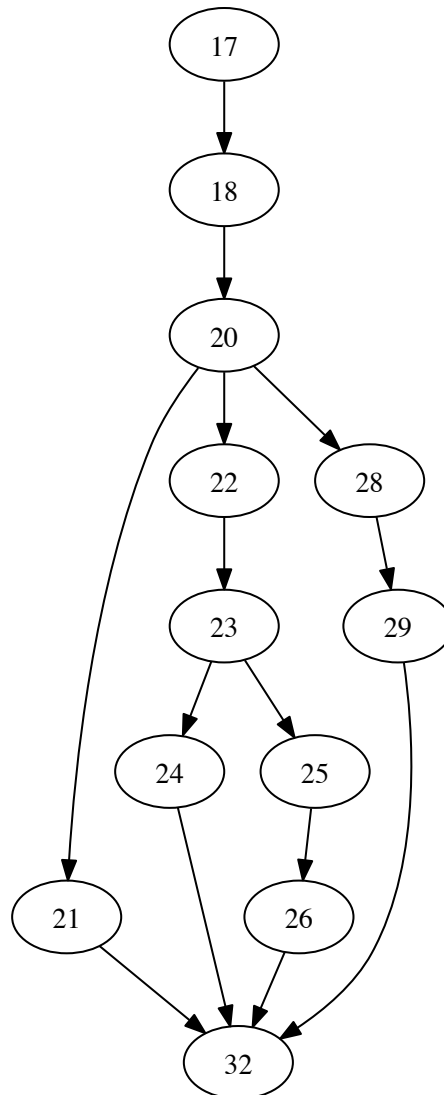
        // d(change); r(amtPaid);
        change = amtPaid;
    }

    // r(change);
    return change;
}
```

Die Variable change weist sowohl eine du als auch eine dd-Anomalie auf.

3.2 Kontrollfluss-Graph

Es folgt der Kontrollfluss-Graph der in der Datenfluss-Analyse analysierten Methode `makeCoffee`.



4 Teststufen

4.1 Komponententest

Die Komponententests wurden in der Praktikumsaufgabe 1 durchgeführt.

4.2 Integrationstest

Im Integrationstest testen wir die einzelnen Software-Komponenten des `CoffeeMakers`. Diese werden sukzessive integriert, bis die Interaktion aller Komponenten erfolgreich getestet wurde. Für diesen Test ist folgende Struktur angedacht:

1. Zuerst wird die Interaktion des `RecipeBook` mit dem `Recipe` getestet.
2. Anschließend wird die `CoffeeMaker`-Komponente mit dem `RecipeBook` integriert.
3. Schlussendlich wird das `Inventory` mit dem `CoffeeMaker` integriert.

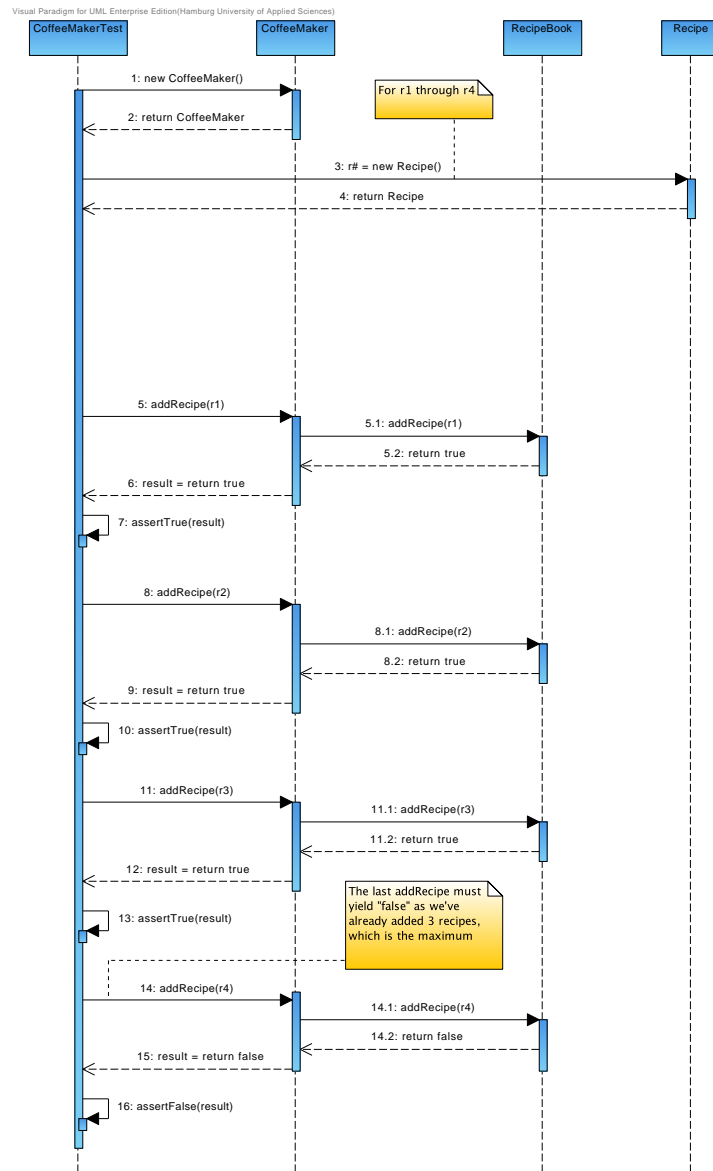
4.3 Systemtest

Der `CoffeeMaker` wird nun an die Hardware angebunden, welche in der Produktion vom Benutzer verwendet wird, um die Kaffee-Maschine zu steuern, sowie den Kaffee auszugeben. Hier müsste man dann entsprechende Schnittstellen schreiben, welche Zugriff auf die einzelnen Funktionen über die Hardware ermöglichen, sowie eine Schnittstelle, welche die Ausgaben des Java-Codes auf dem Display der Kaffee-Maschine anzeigen kann. Das Gesamt-System, bestehend aus Hardware und Software, kann nun ausgiebig getestet und auf korrekte Funktionalität nach Spezifikation geprüft werden.

4.4 Teststrategien

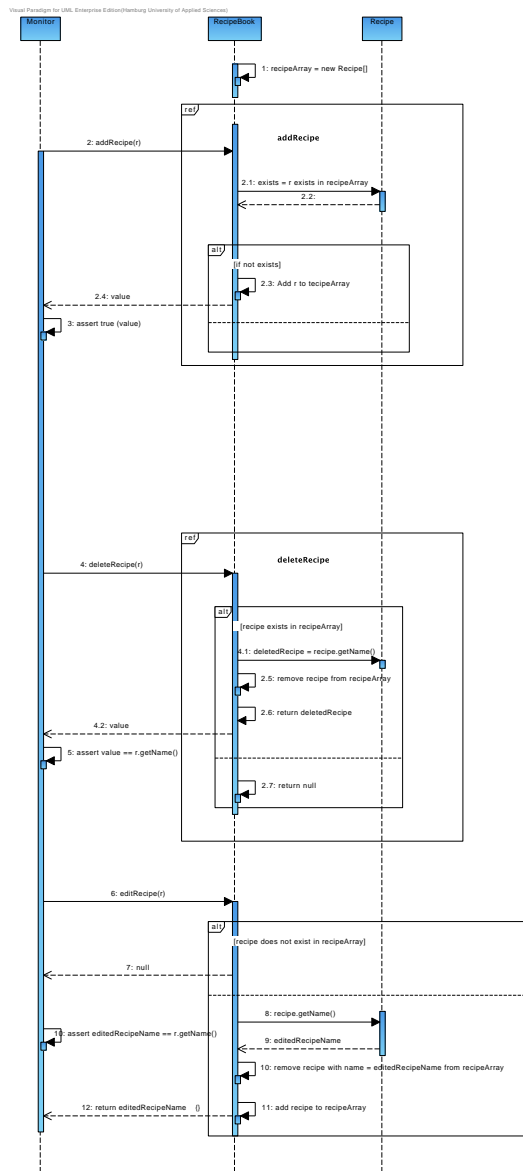
Für Integrations- sowie Systemtest bietet sich das Anwendungsfall-basiertes Testen an, da die Anwendungsfälle bereits spezifiziert sind.

4.4.1 Sequenz-Diagramm - Komponenten-Test



4.4.2 Sequenz-Diagramm - Integrations-Test

Das folgende Diagramm beschreibt den Integrations-Test zwischen RecipeBook und der Recipe-Klasse.



4.4.3 Sequenz-Diagramm - System-Test

In diesem Test wird zusätzlich zu der Funktionalität, welche im Integrationstest geprüft wurde, auch noch das korrekte erzeugen des Kaffees geprüft.

