

Alarm Clock

Jonathan Hyams
Pascal Schmalz

6. Juni 2017

Inhaltsverzeichnis

1	Zweck des Dokument	3
2	Architektur	3
3	Controller	3
4	Filtering	3
4.1	Higher Order Functions	3
4.2	echte Higher Order Functions in Java	3
4.3	Code Beispiel	3
5	Notification	4
6	Resources	4
7	Tests	4
8	Tools	4
9	Main	4
10	Versionskontrolle	5

1 Zweck des Dokument

2 Architektur

3 Controller

4 Filtering

4.1 Higher Order Functions

Die Marketingabteilung von Oracle behauptet gerne Java sei auch Funktional. Higer Order Functions werden aber nicht wirklich unterstützt. TODO wikipedia Higer Order function. Funktionen als Return Values oo

Java unterstützt leider keine Higher order functions. Man kann also keine Funktion als Inputparameter übergeben. Mittels Lambdas ist es lediglich möglich, die eine Funktion ausführen zu lassen und den Rückgabewert als Inputparameter weiter zu verwenden. Dies erlaubt eine kompaktere Notation. Dies reicht uns aber nicht, da wir den Remindern eine Funktion übergeben möchten, mit welcher jeder Reminder selber testet ob er eine Notification absenden soll.

4.2 echte Higher Order Functions in Java

Um dies zu erreichen haben wir eine Form von Higher Order Functions mit Objekten nachgebaut. Ein CriteriaTester ist ein Objekt, welches als Wrapper für eine Funtion dient. Anstelle dieser Funktion übergibt man nun diesen FunktionsWrapper als Inputparameter. Somit konnten wir Funktionen als Inputparameter mittels Objektorientierten prinzipien nachbauen. Man muss nun für jede Funktion ein Objekt erstellen, welches das Interface CriteriaTester implementiert. Und die Filterfunktion isTrue implementieren. Funktionen als Rückgabewert kann man so aber noch nicht wirklich nachbauen. Für uns war das aber nicht nötig.

Dank den oben erwähnten Lambdas kann man dies auch elegant on the Fly erledigen. Da es aber vorkommen kann dass man einen Filter mehrmals benutzt, habe wir uns entschieden die Filter jeweils als eigene Klassen zu implementieren.

4.3 Code Beispiel

```
Reminder r;  
Collection<CriteriaTester> criteria = new ArrayList<>();  
criteria.add(new IsPassed());  
//      example how CriteriaTester can be written on the fly  
//pus this to documentation  
criteria.add(  
    r -> (!r.getTags().contains("hidden"))  
);
```

```
//This lets the Reminder send a notification if the Reminder meets the  
//The first criteria it must pass it th IsPassed()  
//the second criteria is defined on the fly on line number TODO x. it
```

```
r.notifyIf(criteria);
```

5 Notification

6 Resources

7 Tests

8 Tools

9 Main

Dies ist die Klasse die die Main methode enthält. Sie erbt von der Klasse Application, da Sie das JavaFX-GUI launched.

```
Parent root = FXMLLoader.load(getClass().getResource(windowName));
```

windowName ist der Name der fxml Datei (mainWindow.fxml), in der das Aussehen und der Controller definiert wird. "mainWindow.fxml" befindet sich im package "resources".

```
public class Main extends Application {  
    private String windowName = "/mainWindow.fxml";  
    private String title = "Alarm Clock";  
  
    /**  
     * Main method.  
     *  
     * @param args  
     */  
    public static void main(String[] args) {  
        launch(args);  
    }  
  
    /**  
     * The Application Class requires the start method.  
     * It launches the GUI and the programs beneath it  
     *  
     * @param primaryStage  
     * @throws Exception  
     */  
}
```

```
@Override
public void start(Stage primaryStage) throws Exception {
    // If set to false, Notification-Popup still come even after the main
    Platform.setImplicitExit(false);
    primaryStage.setResizable(false);
    Parent root = FXMLLoader.load(getClass().getResource(windowName));
    primaryStage.setTitle(title);
    Scene scene = new Scene(root);
    if (new ConfigReader().isEnabledDarkMode())
        scene.getStylesheets().add("dark.css");
    if (new ConfigReader().isEnabledDarkMode()) {
        scene.getStylesheets().add("dark.css");
    } else {
        scene.getStylesheets().add("styles.css");
    }

    primaryStage.setScene(scene);
    primaryStage.show();

    try {
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}
```

10 Versionskontrolle

Manuelle Version: 1.0.0

Automatische Versionierung: