

Übungsblatt 4 – Multimediaprogrammierung

Achtung: Zur Lösung dieser Übungsaufgabe dürfen ausschließlich die Klassen verwendet werden, die mit der Standardinstallation von JavaFX und Java kommen. Abgaben, die zusätzliche Klassen enthalten, erhalten automatisch null Punkte.

Aufgabe 1: Erweiterung des Songviewers (10 Punkte) (für Hauptfächler)

1. Suchen Sie zunächst nach frei verfügbaren Liedern (z.B. auf www.freeload.de, www.beemp3.com, www.jamendo.com) und entsprechenden Songcovern im Internet. Diese URLs benötigen Sie für Ihren Songplayer. (Also die Dateien nicht herunterladen, sondern in Ihrem Code auf die URLs im Web verweisen)
2. Erweitern Sie nun den einfachen Songviewer aus ÜB3. Nehmen Sie dazu Ihre eigene Abgabe oder die Vorlage von der Homepage (Lösung zu ÜB3).
 - a. Fügen Sie einen Stylesheet ein, in dem Sie das Aussehen der Überschrift und der restlichen Textfelder festlegen.
 - b. CustomNodes erstellen:
 - i. Der erste Knoten zeigt das derzeitige Lied an. Er enthält ein Rectangle, das den gesamten Knoten in einer Farbe einfärbt, das SongCover und ein Label mit Artist und Songname.
 - ii. Der zweite Knoten zeigt eine Vorschau auf den vorherigen/nächsten Song an. Er enthält ein Rectangle mit einer Hintergrundfarbe und das Cover des Songs.
 - c. Fügen Sie die GUI zusammen. Benutzen Sie dazu Layouts:
 - i. Fügen Sie eine Überschrift ein.
 - ii. Fügen Sie darunter die CustomNodes ein: Zuerst die Ansicht auf den vorherigen Song, dann das aktuelle Lied und schließlich die Ansicht auf den nächsten Song.
 - iii. Fügen Sie darunter Buttons zur Steuerung ein: „PreviousSong“, „Play“, „NextSong“
 - d. Fügen Sie einen MediaPlayer in Ihre Anwendung ein, der die Songs entsprechend abspielt und beim Songwechsel pausiert.
 - e. Fügen Sie die Interaktion mit der Anwendung ein:
 - i. Implementieren Sie die Funktionalität der Buttons: Backward, Play, Forward
 - ii. Klickt man auf das Songcover des aktuellen Songs, wird der Track abgespielt.
 - iii. Klickt man auf das Bild des vorherigen/nächsten Songs, wird in der Playliste nach hinten/vorne gesprungen.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe1“, der die Lösung zu dieser Aufgabe enthält.



Figure 1: Mögliches UI des Songplayers

Aufgabe 2: Szenegraph (4 Punkte) (für Nebenfächler)

In der Übung haben Sie gehört, dass JavaFX zur Darstellung graphischer Objekte das Prinzip des Szenengraphen verwendet. Recherchieren Sie diesen Begriff nun genauer. Was versteht man unter Wurzel, Knoten und Blättern eines Szenengraphen? Schreiben Sie eine kurze Erklärung (max. 5 Sätze).

Skizzieren Sie einen Szenengraph (Baumdarstellung) der graphischen Elemente vom Screenshot von Aufgabe 1. Beachten Sie, dass CustomNodes und Layouts (z.B. HBox, VBox) verwendet werden und nehmen Sie diese in Ihren Szenengraphen auf. Geben Sie den Szenengraph mit Ihrer Lösung ab. Mögliche Formate sind PNG, JPEG, GIF und PDF.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe2“, der die Lösung zu dieser Aufgabe enthält.

Aufgabe 3: Programmierskills sind nützlich (6 Punkte) (für Nebenfächler)

In dieser Aufgabe sollen Sie sich darüber klar werden, wofür Programmierkenntnisse, insbesondere im Multimedia-Bereich, nützlich sind.

Lesen Sie dazu die folgenden Artikel (insbesondere die Einleitung und Motivation):

Peter Bøgh Andersen, Jens Bennedsen, Steffen Brandorff, Michael E. Caspersen, Jesper Mosegaard: **Teaching programming to liberal arts students: a narrative media approach** (Proceedings of the 8th annual conference on Innovation and technology in computer science education, 2003)

Mark Guzdial, Elliot Soloway: **Teaching the Nintendo generation to program** (Communications of the ACM, Volume 45, Issue 4, 2002)

Zugriff über DocWeb mit Cip-Kennung:

<https://docweb.lrz-muenchen.de/cgi-bin/doc/nph-webdoc.cgi/000110A/http/portal.acm.org/>

Fassen Sie den Artikel zunächst in ca. 5 Sätzen zusammen.

Gehen Sie anschließend auf die genannten Probleme beim Erlernen vom Programmieren ein.

Welche Gründe werden genannt, warum Programmieren, insbesondere Multimedia-Programmierung, auch für Nicht-Informatiker, z.B. Künstler, wichtig ist.

Nennen Sie weiterhin Gründe, wofür das Programmieren in Ihrem Fach sinnvoll sein kann.

Insgesamt sollten sie ca. 0.5 - 1 Seite zu dem Thema schreiben.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe3“, der die Lösung zu dieser Aufgabe enthält.

Abgabe

„Für Hauptfächler“ bedeutet, dass diese Aufgaben nur von Hauptfach-Studierenden (Diplom Informatik/Medieninformatik, Bachelor Informatik/Medieninformatik) abgegeben werden müssen. „Für Nebenfächler“ bedeutet, dass diese Aufgaben nur von Nebenfach-Studierenden (z.B. Kunst und Multimedia) abgegeben werden müssen. Diese können allerdings auch gerne die normalen Aufgaben (für Hauptfächler) bearbeiten und müssen dann Aufgabe 2 + 3 nicht bearbeiten.

Bitte geben Sie Ihre Lösung als ZIP-Datei bis zum 31.05.2010 10:00 Uhr im UniWorx Portal (<http://www.pst.ifi.lmu.de/uniworx>) ab. Bitte fügen Sie Ihrer Abgabe eine Datei studiengang.txt hinzu, welche Ihren Studiengang enthält. Dies dient der einfacheren Korrektur.