

Neue Funktionen in Version 1.2 des Programms Einsatzplaner

Stand: Freitag, 28. Juli 2017

1. Einstellen der Mindeststunden

Funktionen

Im Programm soll es die Möglichkeit geben, die Anzahl der Mindeststunden einzutragen. Diese Anzahl gibt an, wie viele Stunden abgeleistet werden müssen, bis die Mindestanzahl erreicht wurde und die Person in den Listen entsprechend markiert wird. Dieses Fenster wird durch den Menü-Eintrag „Eigenschaften“ im Menü „Datei“ aufgerufen. Diese Stundenanzahl wird für jede einzelne Kategorie angegeben. Somit kann für Lokführer, Zugführer, Zugbegleiter, Werkstatt, ... eine individuelle Stundenzahl angegeben werden.

Ebenso gibt es eine weitere Zahl, die beschreibt, wie viele Stunden insgesamt geleistet werden müssen.

Diese Zahlen gelten jeweils für die Mindeststunden in einem Jahr.

Konzeption

Es wird ein Eintrag „Eigenschaften“ im Menü Datei hinzugefügt. (Position: Nach „Personaldaten sichern unter...“ und vor „Schließen“ getrennt durch einen Trenner) Damit wird ein Dialog geöffnet, in dem für jede Kategorie ein Wert für die Stunden als ganze Zahl eingetragen werden kann. Dieser Dialog blockiert die gesamte Anwendung. Ebenso soll es ein weiteres Feld geben, in dem eine Gesamtmindeststundenzahl eingegeben werden kann.

Die Daten werden dann in der .ako Datei gespeichert und sind somit datei- und nicht programmabhängig. Siehe json.txt für den genauen Speicherort der Daten.

Diese Daten werden ebenso übernommen, wenn das Personal exportiert werden soll, ebenso alle Daten aus FILE:/personal/.

Gliederung:

- Benutzeroberfläche
- Verbinden mit dem Rest des Programms

Für die Klasse „ManagerPersonal“ werden folgende Attribute und Methoden hinzugefügt:

Im Fenster „Personalmanagement“ wird im Menü „Personalmanagement“ nach „Aktualisieren“ und vor „Schließen“ der Eintrag „Mindeststunden“ eingefügt.

Das „triggern“ dieses Eintrags wird an den Slot „editMinimumHours()“ in der Klasse „PersonalWindow“ gebunden.

In der Methode „editMinimumHours()“ wird ein neuer Dialog der Klasse „MinimumHoursEditorDialog“ eingefügt.

Dieser Dialog enthält im Formularlayout angeordnete Felder. Für jeden Wert von AActivity::Category wird ein Feld für einen Integer-Wert zur Verfügung gestellt. Der Wert liegt im Intervall [0,8760].

Unten gibt es ein weiteres Feld für die Gesamtmindeststundenzahl. Ebenfalls mit einem Feld für einen Integerwert.

Jedes Feld wird auf der rechten Seite mit einem Label versehen, dass angibt, um welchen Wert es sich handelt.

Im unteren linken Eck befindet sich ein Button mit dem Text „Standard wiederherstellen“. Durch seine Betätigung werden alle Werte auf die Standardwerte zurückgesetzt (über die Methode „setDefault()“ in der Klasse zum Dialog). Der Dialog wird dabei nicht geschlossen.

In der rechten unteren Ecke befinden sich zwei weitere Buttons nebeneinander: Rechts: „Ok“. Schließt den Dialog und übernimmt bei Betätigung die Daten aus den Feldern und speichert sie in dem Manager für die Reservierungen. Wird von der Methode „actionOK()“ durchgeführt.

Links: „Abbrechen“. Schließt den Dialog und speichert die Werte allerdings nicht. (Über die Methode actionCancel()“ in der Klasse des Dialogs)

Die Klasse „ManagerPersonal“ sorgt dafür, dass die Daten wie folgt in der .ako-Datei gespeichert werden:

```
personal/ (object)
    minimumHours (object)
        key (AActivity::Category)
        value (int)
```

Die Daten werden dann in der .ako Datei gespeichert und sind somit datei- und nicht programmabhängig.

Diese Daten werden ebenso übernommen, wenn das Personal exportiert werden soll.

2. Summe der Einsatzstunden in der Exportübersicht

Funktionen

In der Anzeige der Stundenübersicht des personal, soll es die Möglichkeit geben, dass eine Summe der jeweiligen Kategorien angezeigt werden.

Diese Anzeige soll sich immer am unteren Ende der Tabelle befinden.

Diese Anzeige soll auch beim Export der Tabellendaten ausgegeben werden.

Konzeption

Folgende Methoden müssen direkt angepasst werden:

```
void PersonalWindow::refreshGesamt()
```

```
void PersonalWindow::print(QPrinter *p)
```

Und jeweils die davon aufgerufenen Methoden, unter Umständen auch in anderen Klassen

3. Kennzeichnung von Personen auf Triebfahrzeugen

Funktionen

- Internes Personal, das eine Ausbildung macht, wird mit „Azubi“ gekennzeichnet
Diese Funktion gilt auch für die Kategorie Zugführer

- Externe Personen, die einen Schnupperkurs machen, sollen mit „ELF“ gekennzeichnet werden, oder alternativ auch mit „Ehrenlokführer“

- Externe Personen, die eine Führerstandsmitfahrt gebucht haben, sollen mit „FS“ oder „Führerstandsmitfahrt“ gekennzeichnet werden

- Sonstige Externe Personen, die auf der Lok mitfahren, sollen mit „Extern“ gekennzeichnet werden.

Diese Kennzeichnung muss jeweils durch einen Strichpunkt vom Namen der Person getrennt werden.

Die Annotationen sollen dabei vor der Berechtigung für die entsprechende Kategorie überprüft werden.

Konzeption

Umgestalten der gesamten Implementierung der eingebundenen Methoden, sodass besser zwischen Model und View/Controller unterschieden wird.

Danach in `AActivity::addPerson(...)` die neuen Funktionen einfügen.