



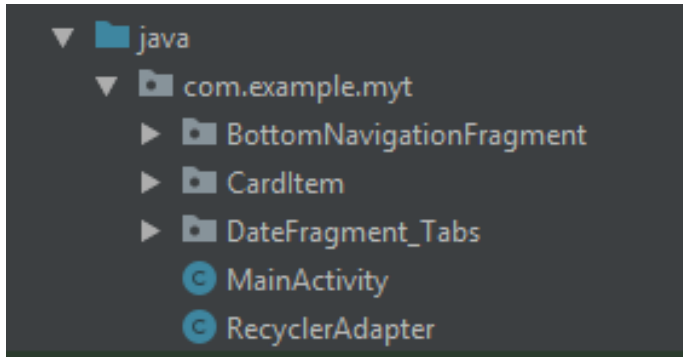
Nilufar
Pirastani
Mobile
Anwendungen

MYT

Manage your Team

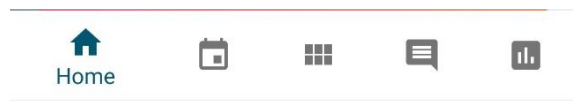
MYT Dokumentation

Zur besseren Übersicht des Programmcodes folgen hier ein paar kurze Erklärungen zur jeweiligen Klasse und wo diese zu finden sind und wie sie eingesetzt werden.



MainActivity

In dieser Klasse wird hauptsächlich die **Bottom Navigation Bar** angelegt, da dies der Hauptpunkt der ganzen App ist. Darüber navigiert man sich durch die diversen Fragments.

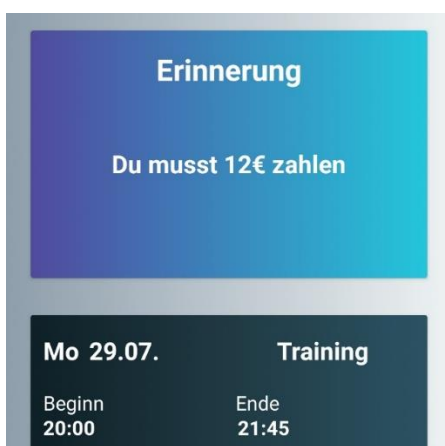


RecyclerViewAdapter

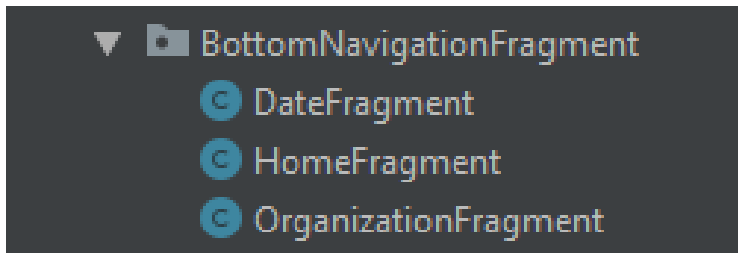
Diese Klasse wurde angelegt, da in einigen Layouts (fragment_date_list.xml, fragment_home.xml) die **recyclerView** eingefügt wurde. Um diesen zu nutzen braucht man diese Klasse. Dort wird es ermöglicht verschiedene Typen (Interface: ListItem) von **CardItems** (CardViews) anzulegen und diese dann als Liste darzustellen.

Zum Beispiel ermöglicht es diese Ansicht:

fragment_home.xml → HomeFragment.java

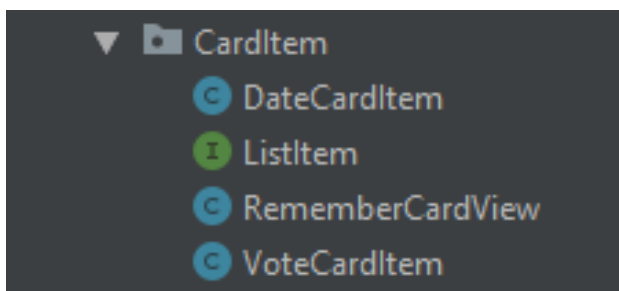


Bottom Navigation Fragments



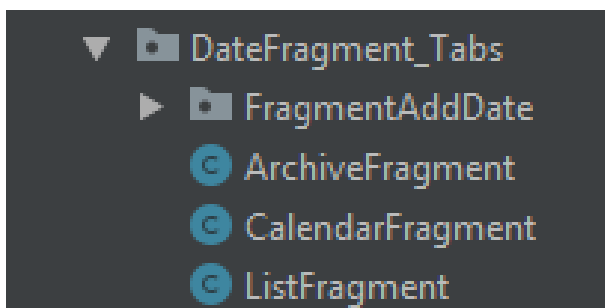
Im Ordner *BottomNavigationFragment* befinden sich die jeweiligen Fragments die entsprechend aufgerufen werden, wenn man über die *Bottom Navigation Bar* auf einer der Items klickt (über *MainActivity* werden diese aufgerufen mit ihrem entsprechendem Layout).

CardItems



Im Ordner *CardItem* befinden sich die jeweiligen **CardViews** bzw. dessen Attribute, die man „bearbeiten“ kann. Dabei befindet sich noch ein **Interface ListItem**, welches dafür zuständig ist, dass man in der Klasse *RecyclerViewAdapter* mit den unterschiedlichen Layouts der CardItems arbeiten kann, ohne mehrere *RecyclerViewAdapter* für die jeweilige cardView zu erstellen.

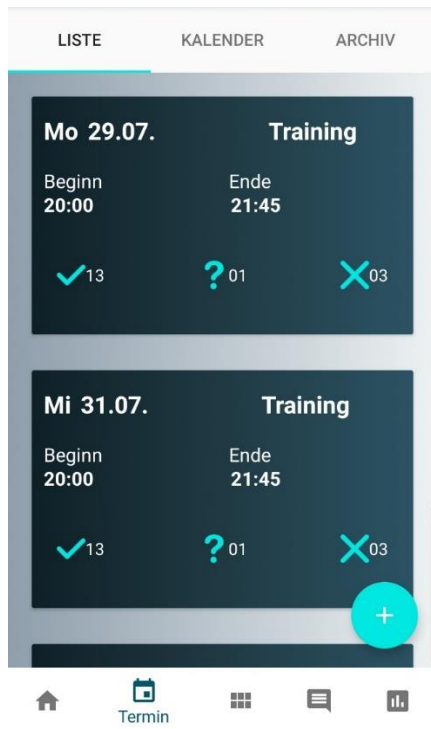
Tabs vom DateFragment



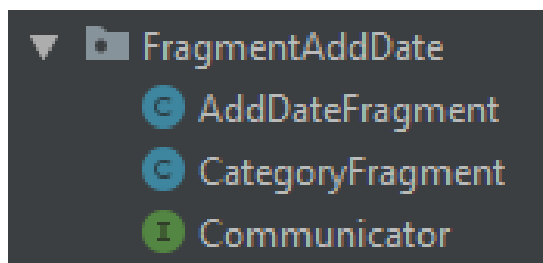
Im *DateFragment* wurde ein **Tabhost** mit drei unterschiedlichen Tabs (Liste, Kalender, Archiv) hinzugefügt. Innerhalb dieser Tabs befinden sich wieder jeweils ein Fragment, welches durch die Layouts *fragment_date_list*, *fragment_date_calendar* und *fragment_date_archiv* dargestellt werden.

Beispiel zu eines dieser Fragments:

fragment_date_list.xml → ListFragment.java



Hinzufügen eines Termins



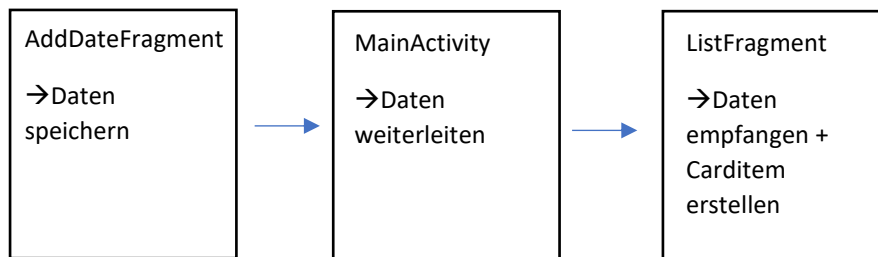
fragment_adddate.xml → AddDateFragment.java



fragment_category.xml → CategoryFragment.java

Über dem FAB-Button im ListFragment gelangt man nachdem Aussuchen der Kategorie (CategoryFragment → Training) zum AddDateFragment. Im AddDateFragment werden die Daten vom User eingetragen, um ein Termin erstellen zu können. Diese Daten werden gespeichert und über das **Interface Communicator**

(über die Methode: `respond(...)`) wird mit der MainActivity kommuniziert, um dann mit dem ListFragment zu kommunizieren:



Die Kommunikation über die beiden Fragments funktioniert.

Hinweis zu der Kommunikation:

Das Erstellen des Carditems funktioniert theoretisch, jedoch optisch nicht. Das heißt, dass das **recyclerView** sich nicht aktualisiert bzw. sobald die Aktion **adapter.notifyDataSetChanged()** in ListFragment in der Methode *createCardItem(...)* ausgeführt wird, gibt es eine Fehlermeldung (deshalb auskommentiert).

Wahrscheinlich liegt das Problem schon in der MainActivity in der Methode *respond(...)*, wo die gewünschte Id des ListFragments nicht gefunden wird. Dies ist also eine provisorische Lösung, um zu vermeiden, dass das Programm abstürzt. Somit kann man nur zur Zeit statisch Carditems anlegen.