

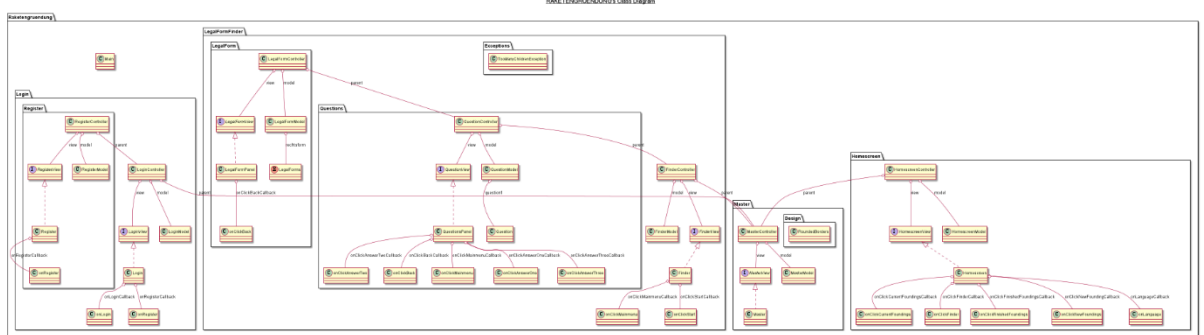
Technische Dokumentation

1. Programmaufbau

Das Programm ist modular aufgebaut. Aus diesem Grund sind alle Bestandteile eines Unterprogramms in einen separaten Ordner untergliedert. Der MASTER Ordner stellt dabei eine Ausnahme dar, da in diesem das Grundgerüst des Programms implementiert und verwaltet wird

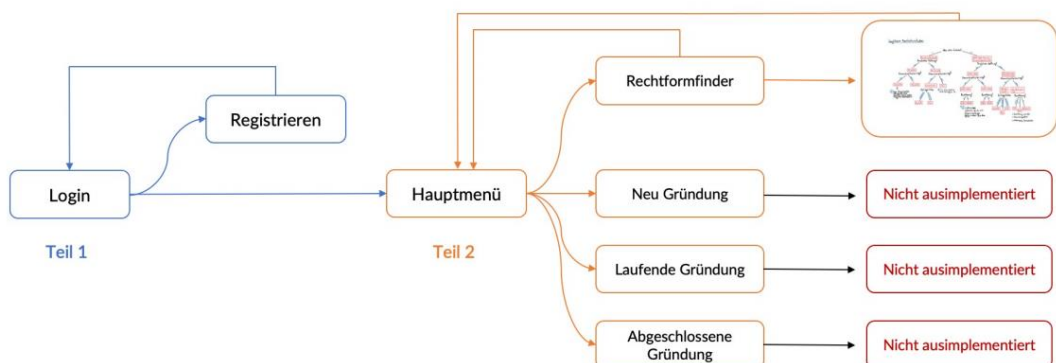
Um ein weiteres Unterprogramm zu starten werden 3 Objekte benötigt, die der Model-View-Controller Architektur folgen.

In diesem Klassendiagramm lässt sich der Aufbau erkennen:



2. Programmablaufplan

Programmablauf



3. Architektur

Wie bereits im Aufbau erwähnt verfolgt das Programm dem Model-View-Controller-Prinzip. Das heißt: Es gibt eine View die sich ausschließlich um das Design kümmert und noch keine Inhalte selbst implementiert, diese wird in unserem Fall um ein Interface erweitert. Das Model baut die Inhalte die angezeigt werden sollen. Als Beispiel ist hierbei das QuestionModel zu erwähnen, das den Entscheidungsbaum baut.

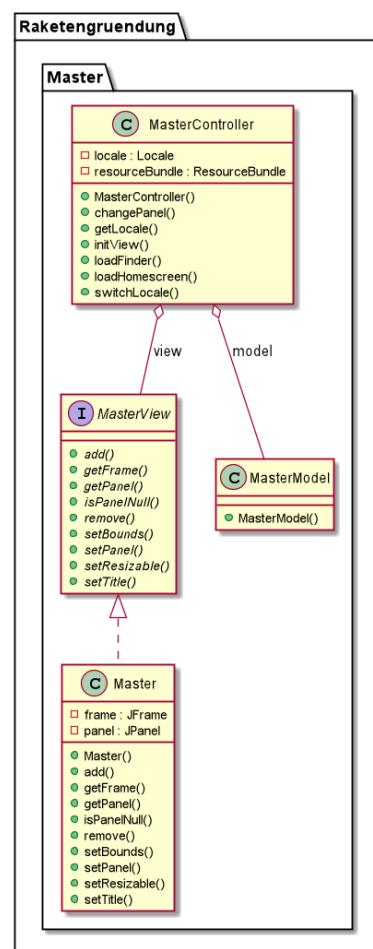
Der Controller ist schließlich dafür da um Inhalt und Design zu verknüpfen. In diesem werden die Texte festgelegt oder die Aktionen, die bei einem Knopfdruck ausgeführt werden sollen.

Damit ein Unterprogramm geladen wird, sind zwingend alle 3 Objekte zu erzeugen!

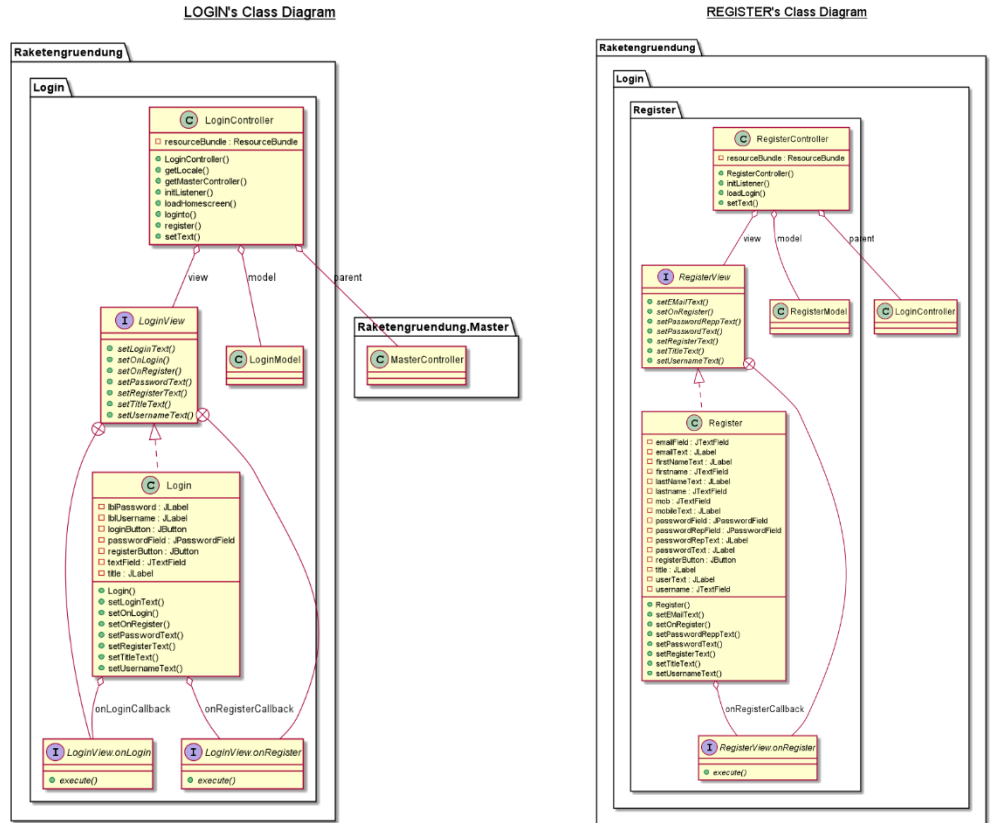
4. Klassendiagramme

a.

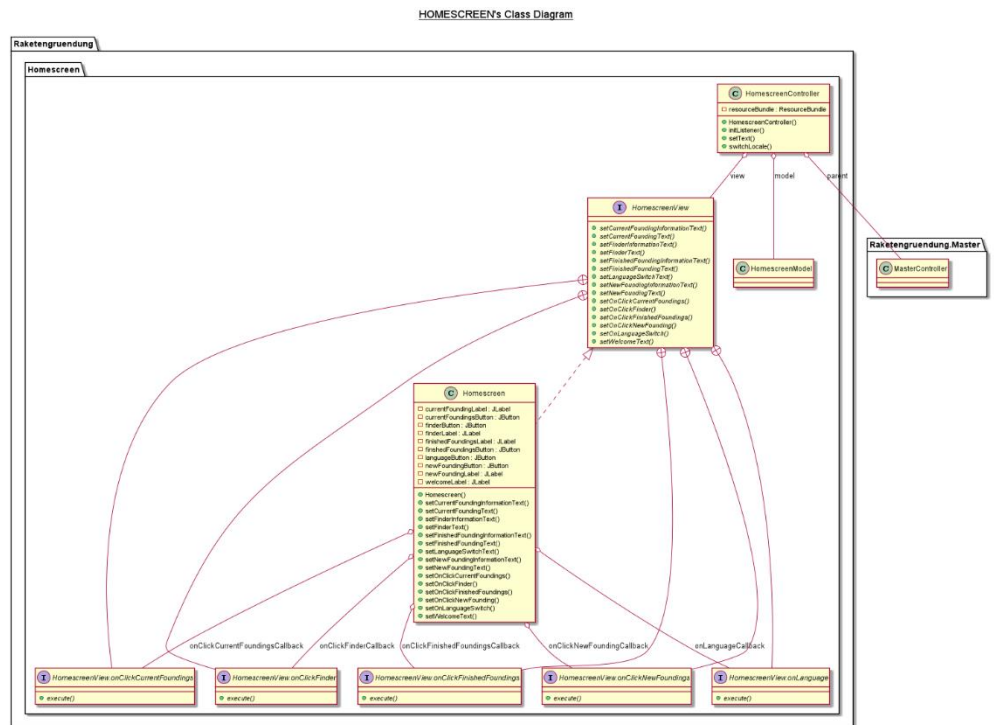
MASTER's Class Diagram



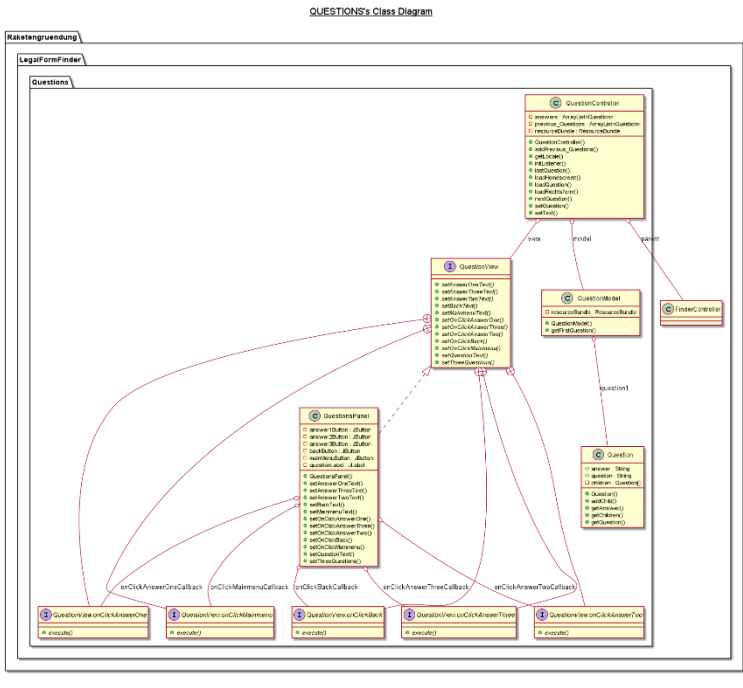
b.



C.



d.



PlantUML diagram generated by SketchUML (<http://www.philippsketchuml.com>)
For more information about this tool, please contact philipp.schneiders@gmail.com

LEGALFORMFINDER's Class Diagram

LEGALFORM's Class Diagram

