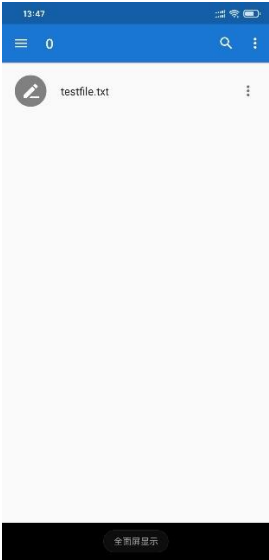
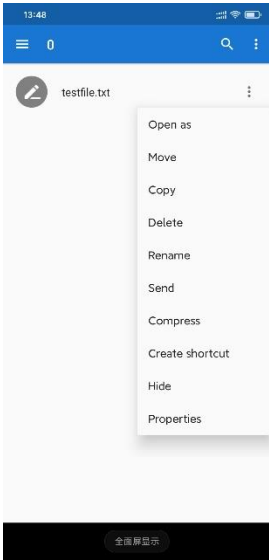


示例场景 文件重命名



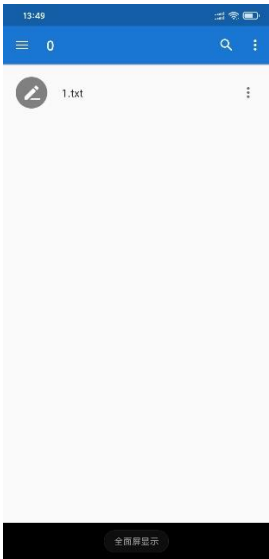
主界面



文件功能菜单



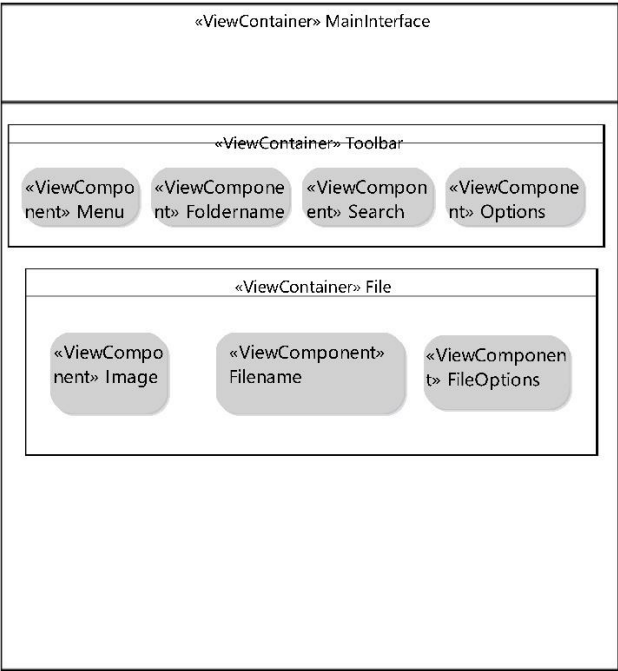
重命名窗口



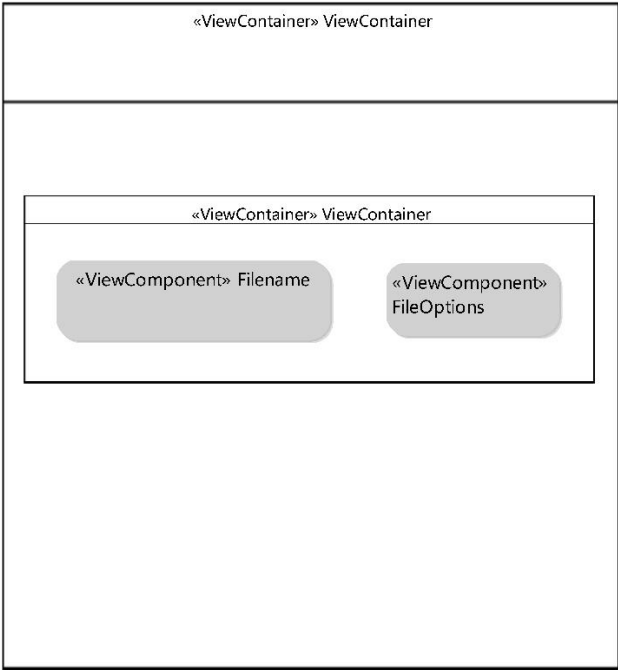
主界面示例，下图将主界面的所有控件加入 ifml 模型中。

在我们的作业中，有以下推荐的要求

- 1. 基本只使用 ViewContainer 和 ViewComponent 两种 ifml 控件，而不使用如 Button, Image, Text 这些更加细化的控件。其中 ViewContainer 为容器，对应于界面，工具栏等，ViewComponent 为控件，对应于按钮，文本框等。
- 2. 控件的位置与 app 中的位置保持相对正确的关系
- 3. 尽量让所有同类控件的名称互不相同，同时又能表现出一定的控件功能特征



由于可以省略无用的控件，事实上可以简略为下图。只保留能表现出重命名作用的文件名控件，以及通向重命名功能的文件选项控件。



总体示例

在完成每个界面的建模的基础上，新增 event 和 navigation flow 两项，用以表示界面之间的跳转和交互。

在我们的作业中，有以下推荐的要求

1. 所有的 event 分为 4 种

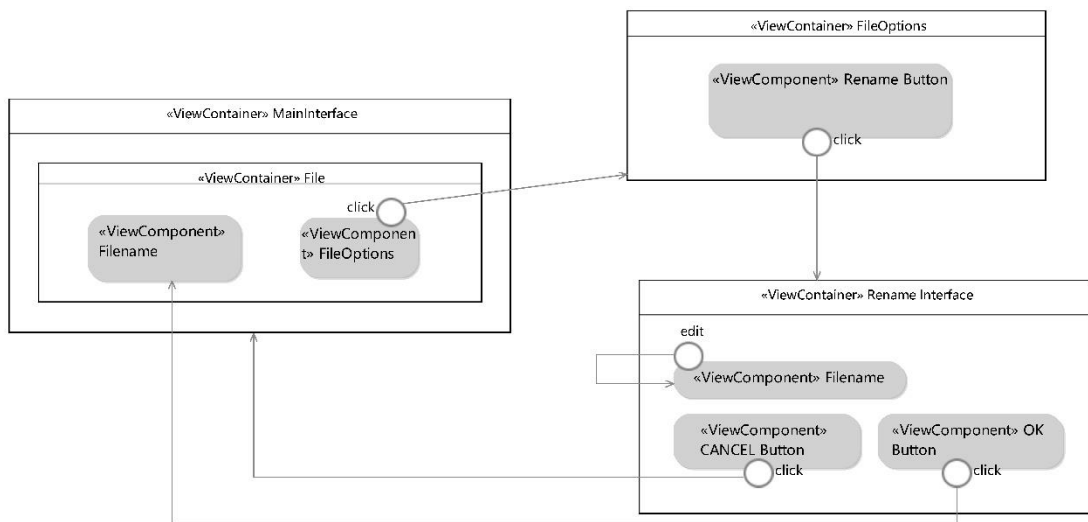
- a) click 点击
- b) longclick 长按
- c) edit 编辑
- d) back 返回

如果有其他 event，建议在实验报告中注释说明

2. 事件附着的对象指代该动作进行的对象。如 FileOptions 上的 click 表示点击 FileOptions 按钮，Filename 上的 edit 表示编辑 Filename 文本框，ViewContainer 上的 back 表示在该界面进行系统返回。

3. 箭头指向的对象表示该事件将会跳转到什么界面。此时又有两种指向方法

- a) 指向某 ViewContainer，表示跳转到该界面。
- b) 指向某 ViewComponent，表示该事件使得此控件的状态改变，同时跳转到该 ViewComponent 所属 ViewContainer 的界面。



可选的加上 back 事件后的界面。

