Sequence diagram for the action "project switch"

If the user presses the button "Switch Project", this sequence of aktions will be occur.

Generics are used to avoid casting.

:ProjectSwitchController

view:V

model:M

static:FileHandler

final int decision = JOptionPane.showConfirmDialog(null, "Wollen Sie ihr Projekt vorher sichern?", "Projekt wechseln", JOptionPane.YES_NO_CANCEL_OPTION);

alt
[decision == 0]

User selected yes

Destroy the view which the controller knows.

1: view.dispose()

Generate for every project configuration file a project entry.

2: model.saveProjectModel()

2.1: writeProjectToFile(this)

Scans the directory where all project configuration files are stored.

3: scanProjectDirectory()

3.1: new LinkedList<ProjectEntry>()

projectList:LinkedList

projectList

4: new ProjectManagementModel(projectList)

projectManager

5: new JProjectManagement<ProjectManagementModel>(projectManagementModel)

[else]

alt
[decision == 1]

User selected no

6: model.destroy()

7: view.dispose()

Destroy the view and the model which the controller knows.

Scans the directory where all project configuration files are stored.

8: scanProjectDirectory()

9: new ProjectManagementMo

10:

Creates a new model which
manages all project entries.

Shows the project management.

ojectManagementModel:ProjectManagementModel ☰

new JProjectManagement<ProjectManagementModel>(projectManagementModel):JProjectManagement ☰

Generate for every proje
configuration file a proje

8.1: new LinkedList<ProjectEntry>()

projectList|0

projectList|0:LinkedList

agementModel(FileHandler.scanProjectDirectory())

projectManagementModel|0

10: new JProjectManagement<ProjectManagementModel>(projectManagementModel)

or every project
n file a project entry.

Creates a new model which
manages all project entries.

0:LinkedList ≡

Shows the project management.

projectManagementModel|0:ProjectManagementModel ≡

new JProjectManagement<ProjectManagementModel>(projectManagementModel)|0:JProjectManagement ≡