**Functions for server files and directories**

**Classes: UtilFiles, UtilFilesData, UtilUrl**

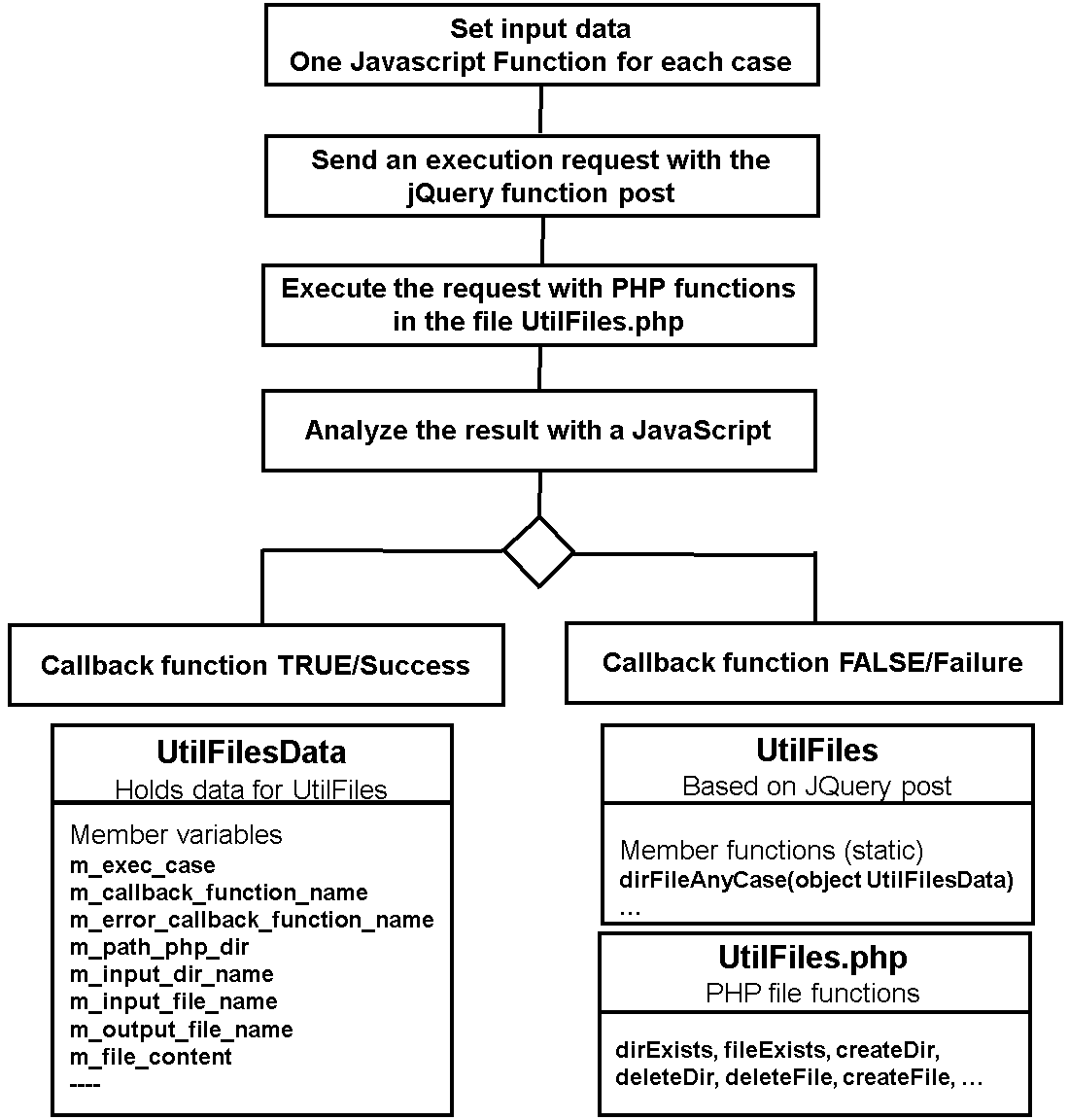
**Bibliotheksfunktionen**

Die Bibliothek hat Funktionen für das Kreieren von einer Datei auf dem Server, für das Löschen einer Datei, eine Datei kopieren, etc. Für jede Bibliothek-Funktion gibt es eine Funktion für die Definition der Eingabe-Daten.

|  |  |
| --- | --- |
| **Eingabe-Funktion** | **Beschreibung der Bibliothek-Funktion** |
| **setDataExecCaseDirExists** | Herausfindet ob einen Ordner existiert |
| **setDataExecCaseFileExists** | Herausfindet ob eine Datei existiert |
| **setDataExecCaseCreateDir** | Kreiert einen Ordner |
| **setDataExecCaseDeleteDir** | Löscht einen Ordner |
| **setDataExecCaseCreateFile** | Kreiert eine Datei |
| **setDataExecCaseDeleteFile** | Löscht eine Datei |
| **setDataExecCaseCopyFile** | Kopiert eine Datei |
| **setDataExecCaseMoveFile** | Bewegt eine Datei zum einen anderen Ordner |
| **setDataExecCaseScanDir** | Macht eine Ordner-Liste mit Dateien und Unter-Ordner |

**So funktioniert die Bibliothek-Funktionen**

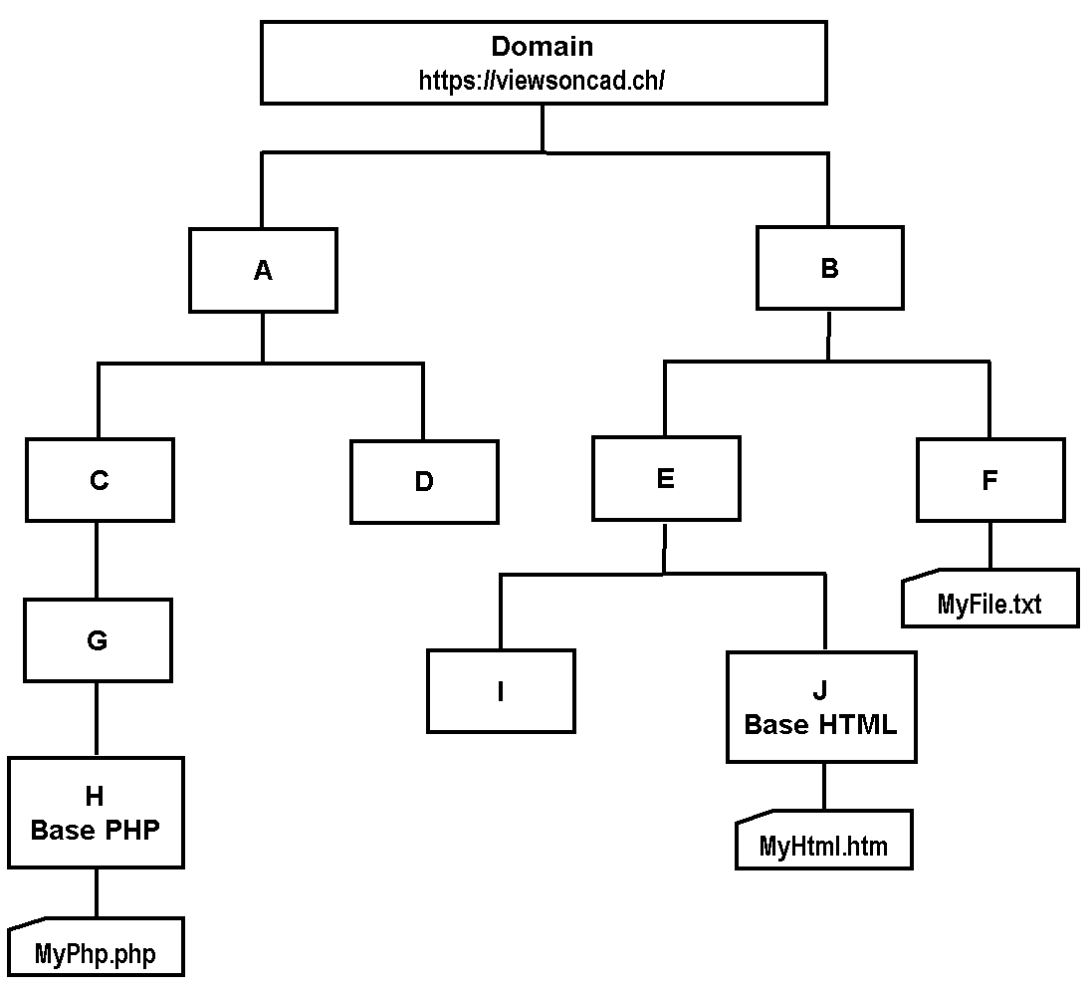
|  |  |
| --- | --- |
| 1 | Die gewählte Eingabe-Funktion definiert die Operation (zum Beispiel Kopieren) in einem Objekt der Klasse **UtilFilesData**. Auch die notwendige Daten für die Operation werden in diesem Objekt definiert |
| 2 | Das Objekt der Klasse UtilFilesData ist die Eingabedaten zur Funktion **DirFileAnyCase** in der Klasse **UtilFiles**. Diese Funktion macht eine jQuery Request für die Operation mit einer PHP Funktion in der Datei UtilFiles.php |
| 3 | Die PHP Funktion in der Datei **UtilFiles.php** macht die Operation. Das Resultat kommt zurück in Form von zwei Parametern |
| 4 | Das Resultat wird analysiert und die ‘richtige’ Callback-Funktion wird angerufen: TRUE/Success oder FALSE/Failed |



**Eingabe-Daten: Absolute oder Relative URLs**

Die Funktionen in der Klasse UtilFiles nehmen Ordner und Datei URLs als Eingabe-Daten. Sie können absolute oder relative URLs sein. Für den PHP Ordner soll der URL relative zum HTML Ordner (mit der HTML Datei) sein. Für alle anderen Ordner und Dateien relative den PHP Ordner (mit der PHP Datei)

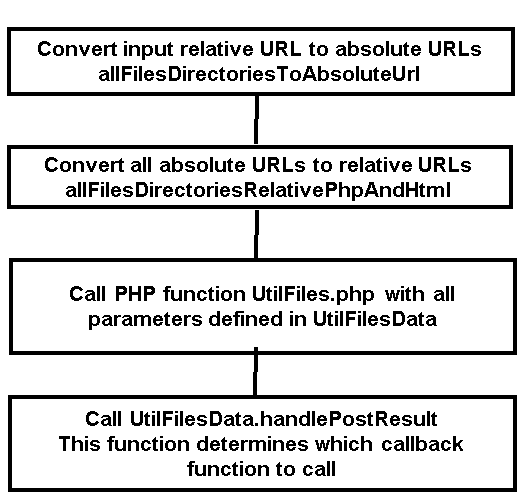






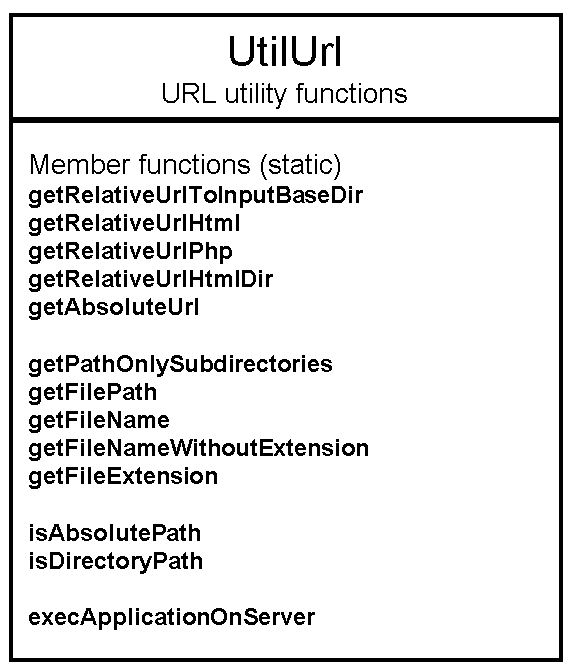
**Funktion UtilFiles.DirFileAnyCase**

Alle relative URLs werden zuerst in der Funktion allFilesDirectoriesToAbsoluteUrl konvertiert zu absolute URLs. Danach werden alle absolute URLs mi der Funktion allFilesDirectoriesRelativePhpAndHtml relative zum Ordner PHP und Ordner HTML konvertiert. Die Funktionen in UtilFiles.php können nur relative IRLs hantieren.



**Klasse UtilUrl für Absolute und Relative URLs**

Das konvertieren zwischen absolute und relative URLs werden mit Funktionen in der Klasse **UtilUrl** gemacht.



**Ordner-Liste mit Dateien und Unter-Ordner**

(**setDataExecCaseScanDir)**

Die Implementation für diese Funktion ist anders als die anderen. Eine temporäre XML Datei ist in einem Unter-Ordner des PHP Ordners kreiert mit der UtilFiles.php Funktion **dirListXml**.

Die XML Datei enthält die Liste mit Namen von Dateien und Ordner. Die Funktion **UtilFile.loadOneXmlFile** wird danach das XML Objekt erzeugen und letztlich wird die gewünschte Array kreiert von der Funktion **UtilFiles.getDirScanArray**.

Diese Kette von Funktionen ist im Objekt UtilFilesData gespeichert und das Objekt wird auch als Eingabe-Daten für die Ketten-Funktionen verwendet.

