

NMAS app switch

20180107

(updated: 20180420 by Mathias Vielwerth)

General Description

This note provides code examples for using app switch towards the NemID code app.

iOS – Objective-C

```
/// Helper class to switch to the NemId code app
/// Note: This App-switch functionality will only work, if adding "nemid-codeapp" to the plist file using key: LSAplicationQueriesSchemes
///
/// Snippet: from plist file:
/// <key>LSApplicationQueriesSchemes</key>
/// <array>
/// <string>nemid-codeapp</string>
/// </array>
///
/// example of use: [NemIDAppSwitcher doAppSwitchWithReturnUrl:@"myownapp://nemidfinished"]
@interface NemIDAppSwitcher : NSObject

+ (BOOL)codeappAvailable;
+ (void)doAppSwitchWithReturnUrl:(NSString*)returnUrl;

@end

@implementation NemIDAppSwitcher

/// Checks the scheme for the code app on the device
/// - Note: Do not use the "nemid-codeapp://" scheme to open the code app
/// - Returns: true or false whether the codeapp is installed on the device
+ (BOOL)codeappAvailable {
    UIApplication *application = [UIApplication sharedApplication];
    NSURL *URL = [NSURL URLWithString:@"nemid-codeapp://codeapp.e-nettet.dk"];
    return [application canOpenURL:URL];
}

/// Opens the universal link for the codeapp, if it's available
/// - Note: Remember to dismiss notification if using the app switch
/// - Parameter returnUrl: The client app scheme/universal Link, where the codeapp will return to
+ (void)doAppSwitchWithReturnUrl:(NSString*)returnUrl {
    if ([NemIDAppSwitcher codeappAvailable]) {
        UIApplication *application = [UIApplication sharedApplication];

        NSURL *url = [NSURL URLWithString:[NSString stringWithFormat:@"https://codeapp.e-nettet.dk?return=%@", returnUrl]];
        if (url != nil) {
            [application openURL:url];
        }
    }
}

@end
```



iOS - Swift

```
/// Helper class to switch to the NemId code app
/// Note: This App-switch functionality will only work, if adding "nemid-codeapp" to the plist file using key: LSApplicationQueriesSchemes
///
/// Snippet: from plist file:
/// <key>LSApplicationQueriesSchemes</key>
/// <array>
/// <string>nemid-codeapp</string>
/// <string>nemid-codeapp-business</string> // Only for Business to Bank solutions
/// </array>
///
/// example of use: NemIdAppSwitcher.doAppSwitch(returnUrl: "myownapp://nemidfinished")
class NemIdAppSwitcher {

    /// Checks the scheme for the code app on the device
    /// - Note: Do not use the "nemid-codeapp://" scheme to open the code app
    /// - Note: For Business for Bank solutions use "nemid-codeapp-business://codeapp.e-nettet.dk"
    /// - Returns: true or false whether the codeapp is installed on the device
    static func codeappAvailable() -> Bool {
        guard let url = URL(string: "nemid-codeapp://codeapp.e-nettet.dk") else {
            return false
        }
        return UIApplication.shared.canOpenURL(url)
    }

    /// Opens the universal link for the codeapp, if it's available
    /// - Note: Remember to dismiss notification if using the app switch
    /// - Parameter returnUrl: The client app scheme, where the codeapp will return to
    func doAppSwitch(returnUrl : String) {
        if NemIdAppSwitcher.codeappAvailable() {
            UIApplication.shared.openURL(URL(string: "https://codeapp.e-nettet.dk?return=\(returnUrl)"))
        }
    }
}
```

iOS – Configurations for environments and Business for Bank

There are 4 different URL that can be passed to openURL(). Two for KOPI and two for production.

KOPI:

- <https://codeapp.e-nettet.dk/kopi>
- <https://codeapp.e-nettet.dk/business/kopi>

PROD:

- <https://codeapp.e-nettet.dk>
- <https://codeapp.e-nettet.dk/business>

These will determine what app is opened by the app-switch.

When using the apps on KOPI, append “-kopi” to the schemes, in order to distinguish from PROD, eg. “nemid-codeapp-kopi”. Remember to add it both in the plist and the code.



Android

```
/**
 * Checks for the Common App by looking for information on the package name the Common App uses.
 * This name is unique for the Common App.
 * @return If the device has the Common App installed
 */
public boolean deviceHasSecondFactor() {
    try {
        getPackageManager().getPackageInfo(packageName, 0);
        return true;
    } catch (PackageManager.NameNotFoundException e) {
        return false;
    }
}

/**
 * Create and launch and intent for the Common App main activity. This will perform the app-switch.
 * If the user uses the back button, they will return to the calling app.
 * If the users completes the flow, the Common App will return to this app, as it was opened
 * through the startActivityForResult call.
 */
public void startSecondFactor() {
    Intent secondFactorIntent = getPackageManager().getLaunchIntentForPackage(packageName);
    secondFactorIntent.setFlags(0);
    startActivityForResult(secondFactorIntent, 0);
}
```

Configurations:

Use different package names in order to distinguish the two apps from each other.

Package names

private : dk.e_nettet.mobilekey.everyone
business for banking : dk.e_nettet.mobilekey.employee

In order to switch to the code app on KOPI, append .kopi to the package names