

Using Scannr URL scheme

You can use a Scannr URL scheme to launch Scannr and perform ID scanning to obtain data from drivers licence. URL scheme that allows you to launch Scannr app is:

```
scannr:///callbackScheme=<my_url_scheme>*
```

For example, you can launch Scanner by following code:

```
if ([[UIApplication sharedApplication] canOpenURL:[NSURL URLWithString:@"scannr:///callbackScheme=foo"]]) {  
    [[UIApplication sharedApplication] openURL:[NSURL URLWithString:@"scannr:///callbackScheme=foo"]];  
} else {  
    NSLog(@"Can't use Scannr");  
}
```

Define **foo** URL scheme in your application project settings. The result is displayed on image:

| General | Capabilities | Info | Build Settings | Build Phases | Build Rules |
|---|--------------|---------------------------------|----------------|--------------|-------------|
| ▼ Custom iOS Target Properties | | | | | |
| Key | Type | Value | | | |
| Bundle versions string, short | String | 1.0 | | | |
| Bundle identifier | String | com.infinum.\${PRODUCT_NAME:rfc | | | |
| InfoDictionary version | String | 6.0 | | | |
| Main storyboard file base name | String | Main | | | |
| Bundle version | String | 1.0 | | | |
| Bundle name | String | \${PRODUCT_NAME} | | | |
| Executable file | String | \${EXECUTABLE_NAME} | | | |
| Application requires iPhone environment | Boolean | YES | | | |
| ▶ Supported interface orientations | Array | (3 items) | | | |
| Bundle display name | String | \${PRODUCT_NAME} | | | |
| Bundle OS Type code | String | APPL | | | |
| Bundle creator OS Type code | String | ???? | | | |
| Localization native development region | String | en | | | |
| ▼ URL types | Array | (1 item) | | | |
| ▼ Item 0 | Dictionary | (2 items) | | | |
| URL identifier | String | | | | |
| ▼ URL Schemes | Array | (1 item) | | | |
| Item 0 | String | foo | | | |
| ▶ Required device capabilities | Array | (1 item) | | | |

After that implement method `-(BOOL)application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation` to obtain scanning result. An example of implementation:

```

- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString
*)sourceApplication annotation:(id)annotation
{
    if ([sourceApplication isEqualToString:@"hr.infinum.photopay.scannr"]) {
        NSString *urlString = [url.absoluteString
stringByReplacingPercentEscapesUsingEncoding:NSUTF8StringEncoding];
        NSString *resultString = [urlString stringByReplacingOccurrencesOfString:@"foo://"
withString:@""];

        NSError *error = nil;
        NSDictionary *resultDict = [NSJSONSerialization JSONObjectWithData:[resultString
dataUsingEncoding:NSUTF8StringEncoding] options:kNilOptions error:&error];
        if (!error) {
            NSLog(@"RESULT OF SCANNING: %@", resultDict);
        } else {
            NSLog(@"ERROR: %@", error);
        }

        return YES;
    }

    return NO;
}

```

Variable `resultDict` will contain result of scanning. [Here](#) you can find more about keys and values contained in `resultDict`.