Profiles Engine API

Note: :This API is not part of the public SDK. It can be found in the SDK API Plug-in.

ProfileEngine API's are mainly used to get the list of available profiles on the mobile, get the information about the active profile, to get information of the profile change, set a profile and related information about it.

Header files

```
#include <mprofileengine.h>
#include <mprofilesnamesarray.h>
#include <mprofile.h>
#include <mprofilename.h>
#include <mprofiletones.h>
```

Link against

```
profileeng.lib
```

Capabilities required for activating new profile: WriteDeviceData

Use cases

These API's are used to show the Profiles. Also used to retrieve information of the currently Active Profile,its Id, Ringtone, MessageAlertTone. They can also be used to set a particular Profile active.

Example code

Before using the API's first we create an instance of MProfileEngine using "CreateProfileEngineL()" API

```
MProfileEngine* lProfileEngine = CreateProfileEngineL();
```

Free the resources using "Release()" API

```
lProfileEngine->Release();
```

or alternatively first Push it onto CleanupStack and then PopAndDestroy()

```
CleanupReleasePushL( *lProfileEngine );
....
CleanupStack::PopAndDestroy();
```

The following code snippet is used to get the available profiles on the phone.

(C) Copyright Nokia 2014. All rights reserved.

```
MProfilesNamesArray* profilesNamesArray =
                        lProfileEngine->ProfilesNamesArrayLC();
TInt lCount = profilesNamesArray->MdcaCount();

for(TInt i=0;i<lCount;i++)
    {
    TBuf<20> lName;
    lName= profilesNamesArray->ProfileName(i)->Name();
    CEikonEnv::Static()->AlertWin(_L("Available Profiles:"),lName);
    lName.Zero();
    }

CleanupStack::PopAndDestroy(profilesNamesArray);
```

The following code snippet is used to list the details of the Active profile and its properties like Profile Name, Id, etc.,

```
MProfile* lProfile =  lProfileEngine->ActiveProfileL();
CleanupReleasePushL( *lProfile );

const MProfileName& lMProfileName = lProfile->ProfileName();
TBuf<10> lProfileName = lMProfileName.Name();
CEikonEnv::Static()->AlertWin(_L("Profile Name:"),lProfileName);

TInt lId = lMProfileName.Id();
TBuf<10> lProfileId;
lProfileId.AppendNum(lId);
CEikonEnv::Static()->AlertWin(_L("its Profile Id:"),lProfileId);

const MProfileTones& lMProfileTones = lProfile->ProfileTones();
TBuf<255> lProfileTone1 = lMProfileTones.RingingTone1();
CEikonEnv::Static()->AlertWin(_L("Profile Ringtone1:"), lProfileTone1);

TBuf<255> lProfileMsgalrt = lMProfileTones.MessageAlertTone();
CEikonEnv::Static()->AlertWin(_L("ProfileMsgAlrtToneName:"),lProfileMsgalrt);

CleanupStack::PopAndDestroy();
```

The following code snippet is used to set the Profile:

We pass the Id of the Profile Name to set the Profile active.

```
lProfileEngine->SetActiveProfileL(0);
```

The following link is useful in getting notifications of profile change:Profile Change

Example project

File:ProfileEngine SampleCode.zip

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.