

# Mobile Native Pop-ups

---

[Short Overview](#)

[API References](#)

[Message pop-up](#)

[Dialog pop-up](#)

[Rate pop-up](#)

[Native Dialog](#)

[PlayMaker Actions](#)

[Support](#)

## *Shot Overview*

---

Plugin will allow you to add native platform popups and preloaders in your project. API is cross platform, so you need to implement it only once, and it will work on all supported mobile platforms. You also can use Playmaker actions as alternative to coding.

### **Supported platforms:**

- IOS
- WP8
- Android

### **Pop-ups types:**

- Rate App
- Dialog
- Message
- Preloader

### **Features:**

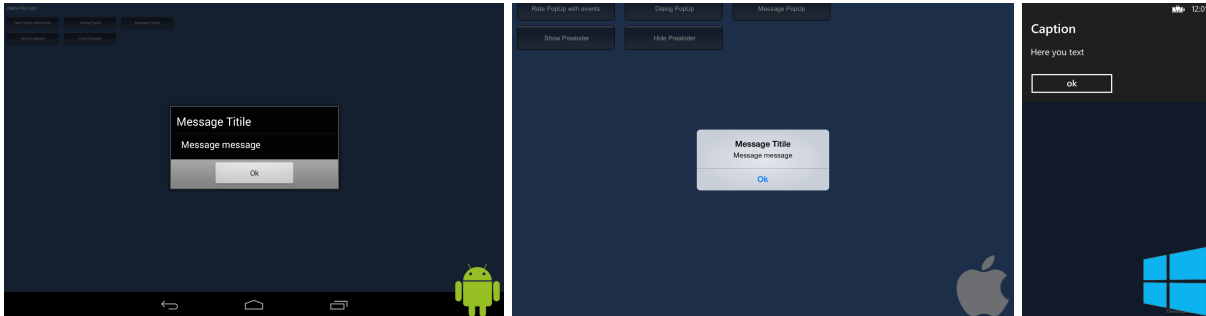
- Cross Platform API
- PlayMaker Actions included
- Fully Documented
- Works out of the box
- Example scene included

## API References

### Showing message pop-up

```
MobileNativeMessage msg = new MobileNativeMessage("Message Titile", "Message message");
```

Result for this API call is showed below:



If you need to find out when message is closed you should add listener:

```
msg.addListener(BaseEvent.COMPLETE, OnMessageClose);
```

**OnMessageClose** function will be called as soon as pop-up is closed.

**OnMessageClose** function example:

```
private void OnMessageClose(CEvent e) {  
    //removing listner  
    e.dispatcher.removeListener(BaseEvent.COMPLETE, OnMessageClose);  
    new MobileNativeMessage("Result", "Message Closed");  
}
```

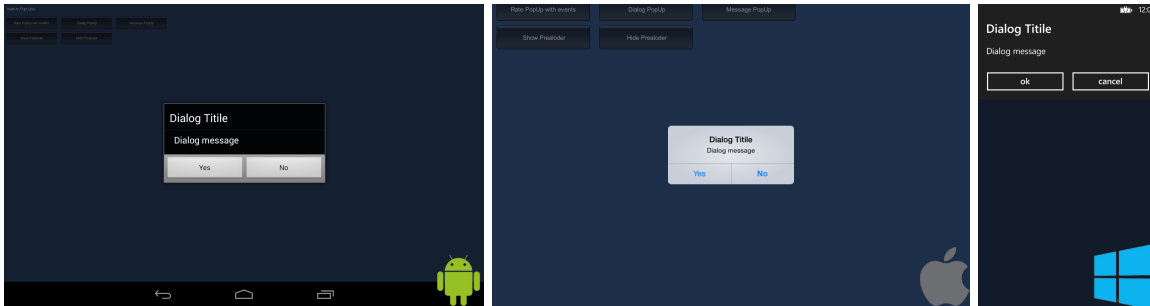
**MobileNativeMessage** contains 2 constructors, in case you need to set text for “Ok” button too

```
public MobileNativeMessage(string title, string message)  
public MobileNativeMessage(string title, string message, string ok)
```

## Showing dialog pop-up

```
MobileNativeDialog dialog = new MobileNativeDialog("Dialog Title", "Dialog message");
```

Result for this API call is showed below:



If you need to find out dialog result you should add listener:

```
dialog.addEventListener(BaseEvent.COMPLETE, OnDialogClose);
```

**OnDialogClose** function will be called as soon as pop up is closed.

**OnDialogClose** function example:

```
private void OnDialogClose(CEvent e) {  
    //removing listner  
    e.dispatcher.removeEventListener(BaseEvent.COMPLETE, OnDialogClose);  
    //parsing result  
    switch((MNDialogResult)e.data) {  
        case MNDialogResult.YES:  
            Debug.Log ("Yes button pressed");  
            break;  
        case MNDialogResult.NO:  
            Debug.Log ("No button pressed");  
            break;  
    }  
}
```

**MobileNativeDialog** contains 2 constructors, in case you need to set text for “Yes” and “No” buttons.

```
public MobileNativeDialog(string title, string message)  
public MobileNativeDialog(string title, string message, string yes, string no)
```

## Showing rate pop-up

```
MobileNativeRateUs ratePopUp = new MobileNativeRateUs("Like this game?", "Please rate to support future updates!");
```

if user will select "Rate" option, he will be redirected to the app rating page. So we need to specify your app data before starting pop up.

Set apple Id for IOS app.

```
ratePopUp.SetAppleId(appleId);
```

Set Play Market Url for Android app

```
ratePopUp.SetAndroidAppUrl(androidAppUrl);
```

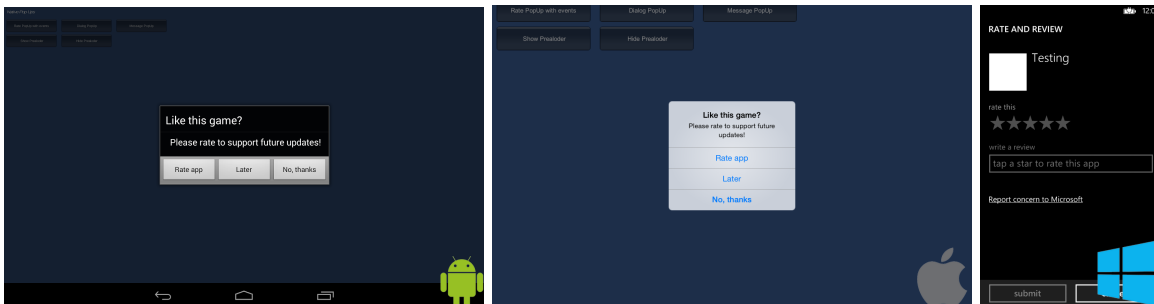
Do nothing for WP8.

And then we can start rate pop up by calling:

```
ratePopUp.Start();
```

Result for this API call is showed below:

Note: For WP8 will be created dialog pop up, but Yes option will redirect to the rating page as on screenshot below



If you need to find out pop-up result you should add listener:

```
ratePopUp.addEventListener(BaseEvent.COMPLETE, OnRatePopUpClose);
```

**OnRatePopUpClose** function will be called as soon as pop up is closed.

**OnRatePopUpClose** function example:

```
private void OnRatePopUpClose(CEvent e) {  
    //removing listener  
    e.dispatcher.removeEventListener(BaseEvent.COMPLETE, OnRatePopUpClose);  
    //parsing result  
    switch((MNDialogResult)e.data) {  
        case MNDialogResult.RATED:  
            Debug.Log ("Rate Option pickied");  
            break;  
        case MNDialogResult.REMIND:  
            Debug.Log ("Remind Option pickied");  
    }  
}
```

```

        break;
    case MNDialogResult.DECLINED:
        Debug.Log ("Declined Option pickied");
        break;
    }
}

```

**MobileNativeRateUs** contains 2 constructors, in case you need to set text for “Yes”, “Later” and “No” buttons.

```

public MobileNativeRateUs(string title, string message)
public MobileNativeRateUs(string title, string message, string yes, string later,
string no)

```

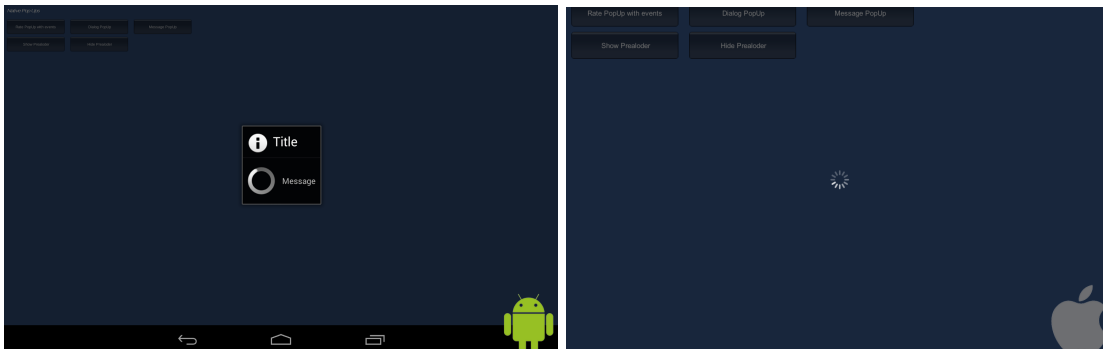
## Native Preloader

**Warning:** Not yet supported for WP8

If you want to show platform native preloader and disable all user interaction use:

```
MNP.ShowPreloader("Title", "Message");
```

Result for this API call is showed below:



When you need to disable preloader all allow user interaction, simply call:

```
MNP.HidePreloader();
```

## *PlayMaker Actions*

As alternative to the coding you can use Playmaker actions.

Actions can be found in zip archive under:

**Assets/Extensions/MobileNativePopUps/Addons/PlayMakerActions.zip**

After extracting action files, you can use native pop-ups actions. In action browser actions can be found under the **Mobile Native Pop-ups** tab

Here is list of actions currently available with the plugin:

- MNP\_MessagePopup
- MNP\_DialogPopup
- MNP\_RatePopup
- MNP\_ShowPreloader
- MNP\_HidePreloader