

NativeScript “Forms” and Modal Dialogs

Jogesh K. Muppala



THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系



香港科技大學
THE HONG KONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY

NativeScript “Forms”

- Data entry through “form”-like elements are facilitated in NativeScript
- TextField component allows you to type text in your app
 - Attributes such as secure for handling password fields, and pipes for specifying the text format the control should use
 - Event bindings like textChange, tap, returnPress

Other NativeScript “Form” Widgets

- Switch
- Slider
- List Picker
- Date Picker
- Time Picker

NativeScript Forms Support

- `NativeScriptFormsModule`: supports one and two-way data binding between the “form” elements and code
- Can leverage Angular Template Forms and Reactive Forms support within the NativeScript application

NativeScript Modal Dialog

- Modals are displayed over the current page
- Modals are themselves just NativeScript components
 - Also must include in entryComponents property of NgModule
- Modals are opened and presented from another component
 - Inject ModalDialogService
 - Use the showModal() method of the service
 - Returns a promise
 - ModalDialogOptions enable passing in parameters:
 - Need to pass in ViewContainerRef of the current page
 - Use the context property to pass in parameters
 - Can specify if modal should be full screen

NativeScript Modal Dialog

- In the Modal component inject ModalDialogParams
 - Params gives access to context to retrieve the parameters that are passed into the modal
- Call Params.closeCallback() to close modal
 - Can send results back as parameters to closeCallback()
 - The promise resolve handler in the other component will receive the results