NativeScript Navigation

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
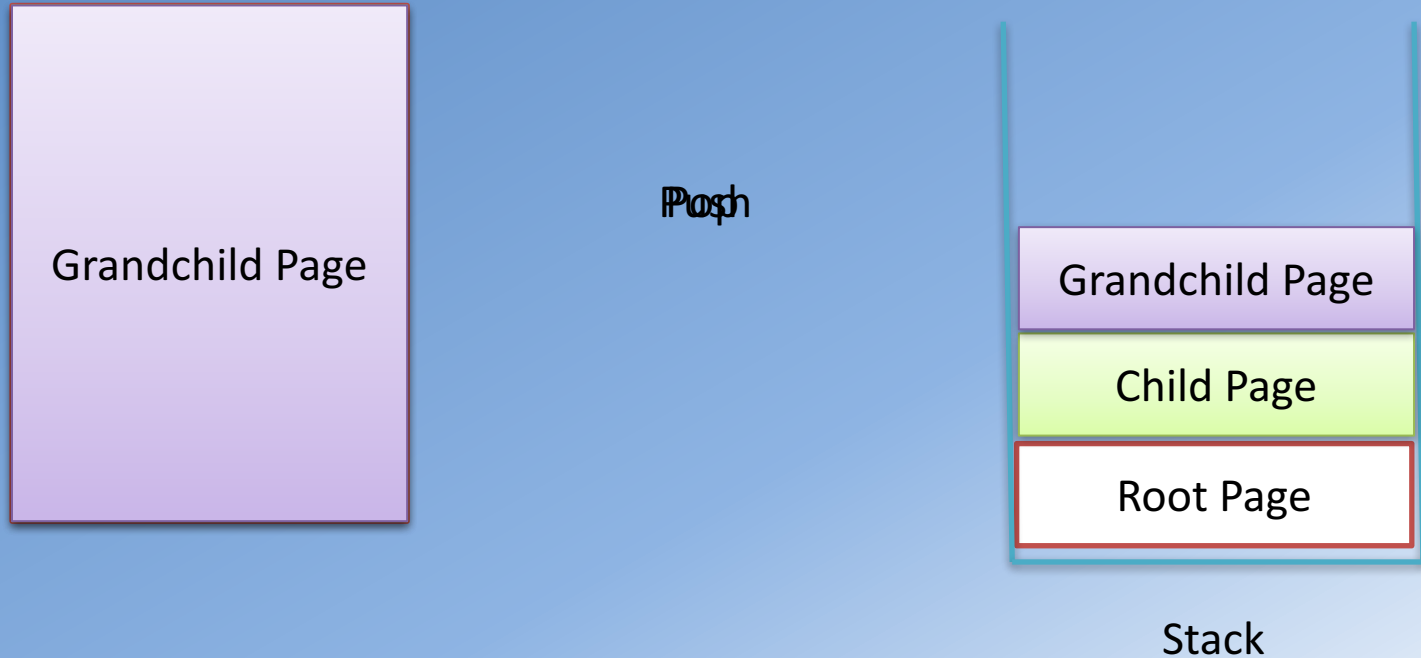SCIENCE AND TECHNOLOGY

NativeScript Navigation

- NativeScript leverages Angular router for implementing navigation

Angular	NativeScript
RouterModule	NativeScriptRouterModule
<router-outlet>	<page-router-outlet>
routerLink	nsRouterLink

NativeScript Navigation

- NativeScript apps built around pages
  - Separate application screens
  - Instances of Page class
- Page navigation integrated with native navigation elements on the mobile platform
  - Back button in Android
  - NavigationBar in iOS

Typical Mobile Navigation

Grandchild Page

Push
Pop

Grandchild Page

Child Page

Root Page

Stack

Page Router Outlet

- pageRouterOutlet: uses pages to navigate
  - `<page-router-outlet></page-router-outlet>`
  - New component will be loaded as a root view in a new page
  - Content outside the `<page-router-outlet>` will not be included in the new page
  - Use as single root element in the application component
  - When navigating forward current page and views are saved in native navigation stack. Component not destroyed

NsRouterLink and RouterExtensions

- nsRouterLink the NativeScript counterpart for Angular's routerLink

  `<GridLayout class="sidedrawer-list-item" rows="*" columns="*"`

      `orientation="horizontal" [nsRouterLink]="['/menu']" clearHistory="true">`

- RouterExtensions

  - Enables navigation through code
  - Several methods available: back(), backToPreviousPage(), navigate(), navigatebyUrl() etc.
  - back() - goes back to the previous router location even if the navigation occurred inside the child router outlet on the current page
  - backToPreviousPage() - goes back to the previous page. The method skips all child router-outlet navigations inside the current page and goes directly to the previous one.

Route Parameters

- Route parameters can be specified using a link parameter array while specifiyng the link
  - e.g., <StackLayout [ngRouterLink]="['/dishdetail', dish.id]">
- Can also be used within a method
  - e.g., this.router.navigate(['/dishdetail', dish.id]);

ActivatedRoute Service

- Router service that provides useful information about the routes, including:
  - url: An Observable of the route path(s), represented as an array of strings for each part of the route path.
  - params: An Observable that contains the required and optional parameters specific to the route
  - queryParams: An Observable that contains the query parameters available to all routes.