

Webshop user module - Frontend:

Current System

Framework: Angular

Framework Version: 1.6.1

Technologies Involved: AngularJS, HTML5, CSS3, JQUERY, JSON, AJAX, Java Script

Disadvantages:

Lagging UI, Name Clashes, Overloading DOM, Loads more JS and CSS files

Proposed User Module - Frontend:

Framework: Angular

Framework Version: 4.4.6

Technologies Involved: AngularJS, HTML5, CSS3, JSON, Type Script

Advantages:

UI Loads much faster, Lightweight

Angular4 is Component Based development framework. The logics will be written in TypeScript and a map file will be generated automatically from ".ts" file. This helps generating the "polyfill" and "vendor" files for production.

The Angular4 Linknet will be module based. We will be creating a module for "user" and "admin" (in future). Modules will be importing its own components.

The routing will be much efficient to handle in Angular4. The routing path can be made visible or hidden in the address bar based on the requirement.

We are going to follow the Angular.org proposed folder structure in Linknet project

The difference between the current folder structure(Angular 1.X) and the proposed folder structure(Angular 4) will be massive and this will improve performance.

For example, in "Category Mngement" widget, the following is the Current and Proposed folder structure.

Category Management:

Current Folder Structure

```
|
|-index.html
|
|-widgets
|
|----category
|
|-----template.html
|
|-----subCategoryTemplate.html
|
|-----dialogueTemplate.html
|
|-----category.js
|
|-----draganddrop.js
```

Proposed Folder Structure (Only for development. The same will not be reflected in production build)

```
|
|-index.html
|
|-src
|
|----app
|
|-----app.component.html
|
|-----app.component.ts
|
|-----categories
|
|-----app.categories.component.html
|
|-----app.categories.component.ts
```

|
|-----app.categories.component.css

Post Development:

|
|-index.html
|-polyfills.js
|-vendor.js

Usage of External Libraries

There are many supporting external libraries that can be hooked to the application hassle free. All we need to do is just download the plugin from the source URL using npm and import it in Ngmodules and whatever components we need the helper file.

The source file of the library will also be packed to the “vendors”, This will reduce the loading of multiple JS files in the browser.

Following are the libraries that we are using in Webshop,

Angular 1.x	Angular 4.x	Required Yes/No
angular-carousel	angular4-carousel	Yes
angular-moment	--	NO
angular-sanitize	--	NO
angular-ui-router	--	NO
bootstrap-carousel	--	NO
JQuery	--	NO
md-data-table	--	NO
ng-flow	--	NO

Support for Material Design:

As our current system (Angular1.X) is using Material Design User Interface and User Experience pattern, we are not going to compromise in Material Design template pattern.

\

We will be using “Angular Material 2” ([material2](#)) - The Material Design version for Angular 4

Framework. For responsive UX, we will be using [flex-layout](#)

Mobile First Architecture:

The new proposed design will be implemented and architected in the following way,



Responsive Web Design

Current UI/UX Architecture

Mobile First Web Design

Proposed UI/UX Architecture ANGULAR - 4.x

