

## Table of contents

Table of contents .....	1
About DMXzone Visual App Designer for Framework 7 .....	2
Features in Detail.....	3
The Basics: Setting Up Visual App Designer .....	41
Advanced: Manage Pages and Content of your App .....	45
Advanced: Using Navbars and Toolbar in Visual App Designer.....	53
Advanced: Insert List Blocks and List Items in Visual App Designer .....	64
Advanced: Insert Cards with Header, Footer and Content .....	74
Advanced: Using Media Lists in DMXzone Visual App Designer .....	84
Advanced: Building Database Driven Mobile Apps .....	95
Video: Setting Up Visual App Designer .....	121
Video: Manage Pages and Content of your App.....	121
Video: Using Navbars and Toolbar in Visual App Designer .....	121
Video: Insert List Blocks and List Items in Visual App Designer .....	121
Video: Insert Cards with Header, Footer and Content.....	121
Video: Using Media Lists in DMXzone Visual App Designer .....	121
Video: Building Database Driven Mobile Apps .....	122

## About DMXzone Visual App Designer for Framework 7

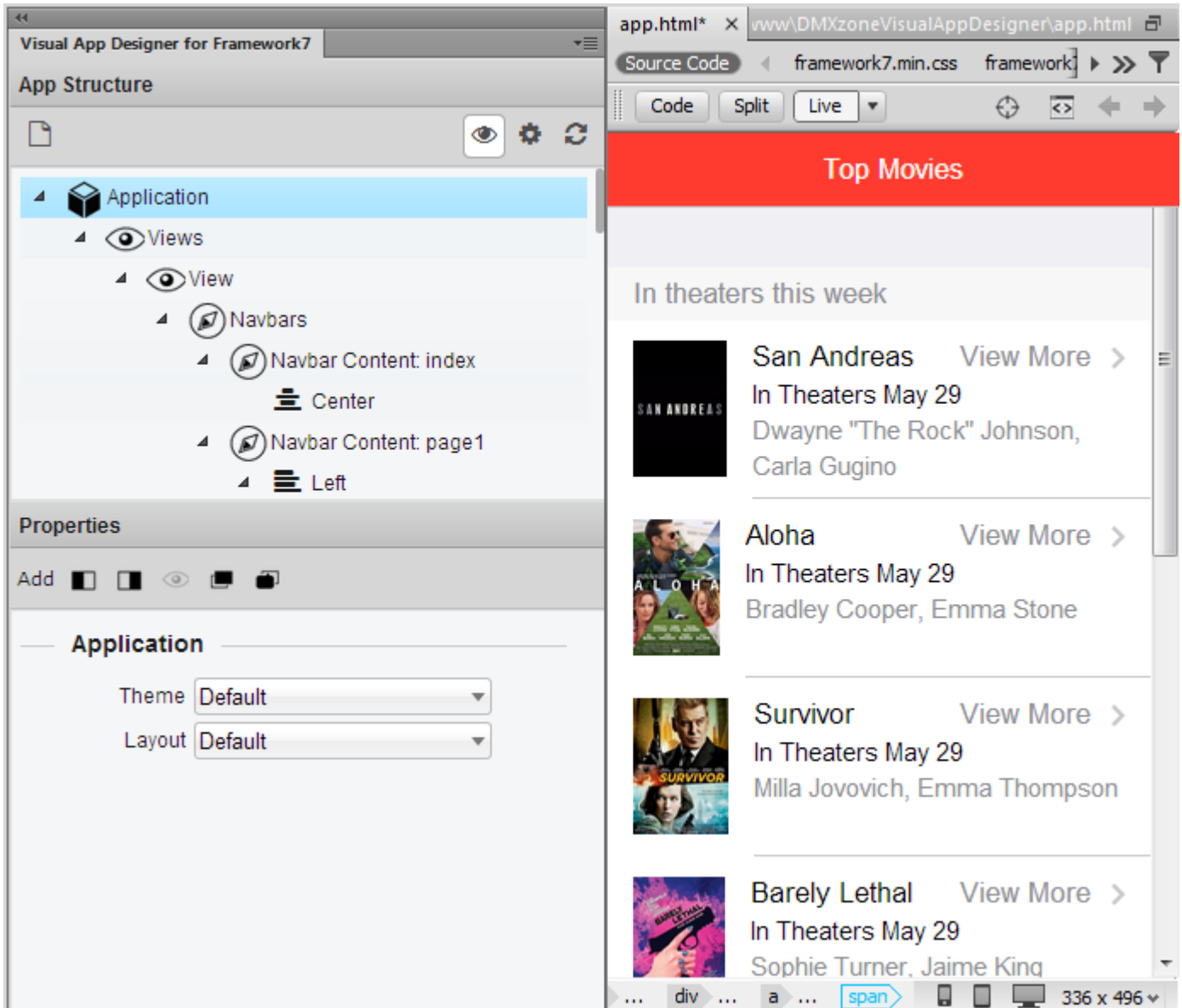
Create amazing HTML5 and CSS based mobile apps fully visually in Dreamweaver with [DMXzone Visual App Designer](#)! Use the ready to go Mobile UI Components and manage the app logical structure in the dedicated floating panel. Build your app with context aware drag & drop and enter the components properties in the smart property inspector. Empower the great looking Framework7 mobile framework for awesome native looking designs.

Make dynamic apps together with the [HTML5 Data Bindings](#) and connect them to databases with the [DMXzone Database Connector PHP/ASP](#) and [Updater PHP/ASP](#). When you are done with the app you can build, test and package it directly in [DMXzone Cordova Builder](#) for any platform.

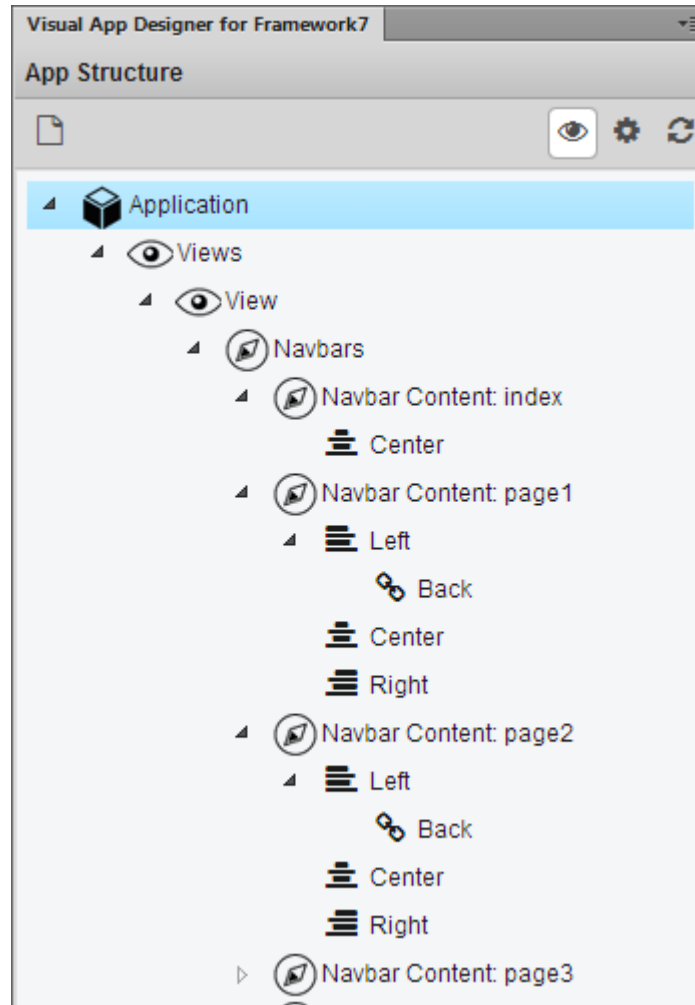
## Features in Detail

### Visual App Designer General Features

- Fully Visual App Design in Dreamweaver** - Get ready to design your apps fully visually in Dreamweaver with the DMXzone Visual App Designer for Framework7. Now you can directly see your logical app structure and its components in the dedicated floater and just add desired components and assign actions!

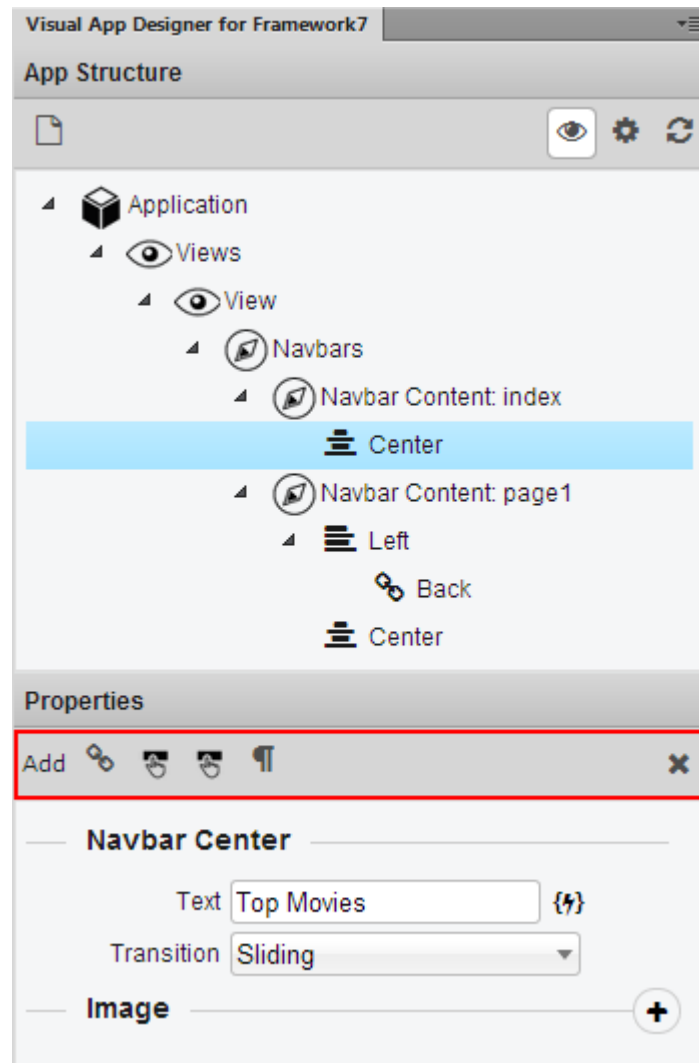


- **Visual App Structure at your disposal** - Work with a high level view of your app where the whole picture is clearly visible and you are not buried under nested structuring details.

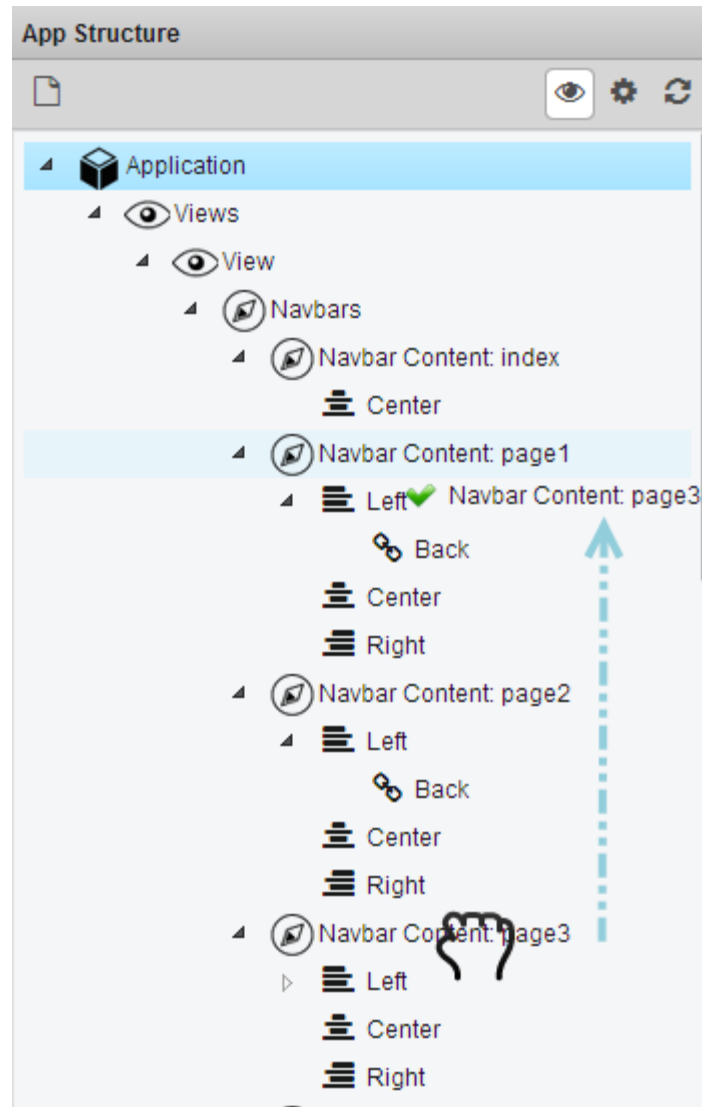




- **Context sensitive components editing** - You don't need to learn any complex app structures! The Visual App Designer does that for you – it will let you do add the UI components that are only allowed in that context! So the right structure is always guaranteed for you!



- **Visually reorder components with Drag & Drop** - So you want to change the order of your components and adjust the layout? It is all done by simple drag & drop actions! Again you can only drag components to places where they are allowed to be! The drag status is clearly indicated with icons.



- **Multi pages fast responsive apps** - Organize your app in clear pages that can be either inline in one page as a Single Page App and or be separated in child files that are loaded dynamically through Ajax for the best performance. In any way, the user will experience a fluid native app that does not have any loading delays but is mighty fast!

The image shows two views of a mobile application. The left view, titled "Top Movies", displays a list of movies currently in theaters. The right view, titled "Movie Info", shows the details for the selected movie, "San Andreas".

**Top Movies View:**

- San Andreas** View More  
In Theaters May 29  
Dwayne "The Rock" Johnson, Carla Gugino
- Aloha** View More >  
In Theaters May 29  
Bradley Cooper, Emma Stone
- Survivor** View More >  
In Theaters May 29  
Milla Jovovich, Emma Thompson
- Barely Lethal** View More >  
In Theaters May 29  
Sophie Turner, Jaime King
- Walking On** View More >

**Movie Info View (San Andreas):**

- San Andreas (2015)**
- Tickets & Showtimes**
- Journey 2: The Mysterious Island's Brad Peyton steps into disaster movie territory with this 3D film surrounding an earthquake that hits the West Coast. Carlton Cuse provides the script.

- **Smart visual components with dynamic properties** - Every UI component comes with its own dynamic properties. You can easily change them in the dedicated property inspector. The property inspector shows only the valid properties for the selected component in the current context. So you will see only the properties you need.

The screenshot shows a 'Properties' window for a 'Link' component. The window is divided into sections: Link, Action, Icon, and Badge. Each section contains various property fields and dropdown menus. The 'Link' section includes fields for ID, Link (with a '#' character and a dynamic property icon), Text (set to 'Back'), Text Color, Back Color, Animation, and Tooltip. There are also checkboxes for 'Icon Only' and 'External'. The 'Action' section has dropdowns for 'Action' (set to 'Back') and 'Page' (set to 'None'). The 'Icon' section has a dropdown for 'Icon' (set to '< Back') and a dropdown for 'Position' (set to 'Left'). The 'Badge' section is currently empty, indicated by a '+' icon.

**Properties**

**Link**

ID

Link #  {#} ↗

Text  {#}

Text Color  ▼

Back Color  ▼

Animation  ▼

Tooltip  {#}

Icon Only

External

**Action**

Action  ▼

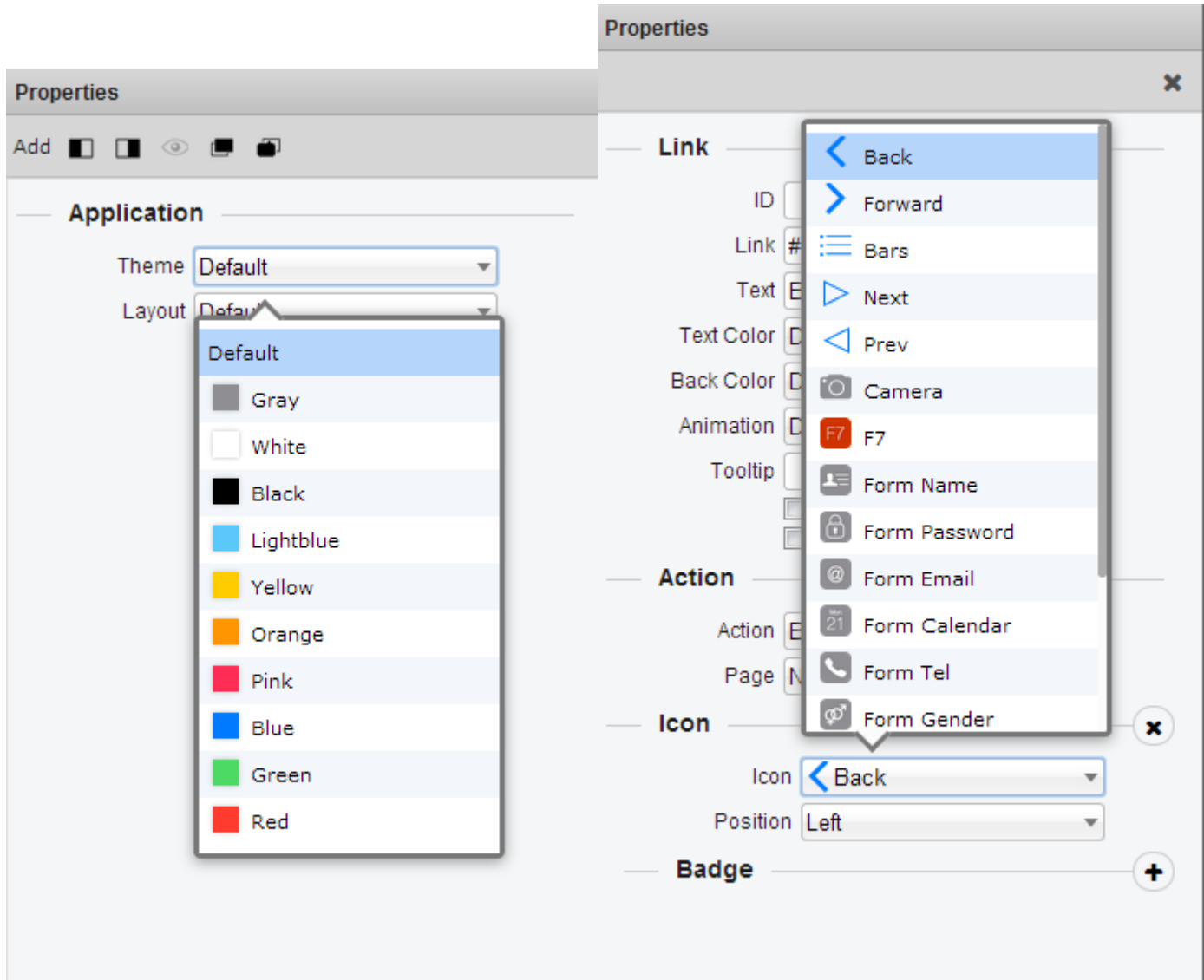
Page  ▼

**Icon**  ▼

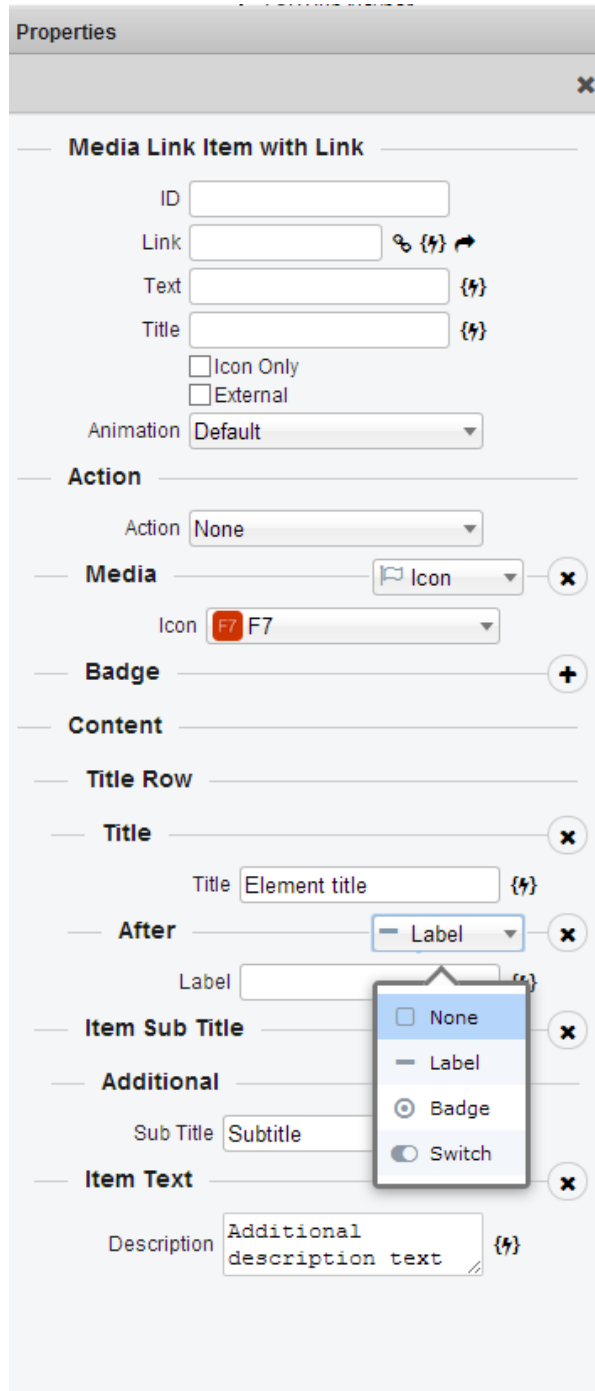
Position  ▼

**Badge**  +

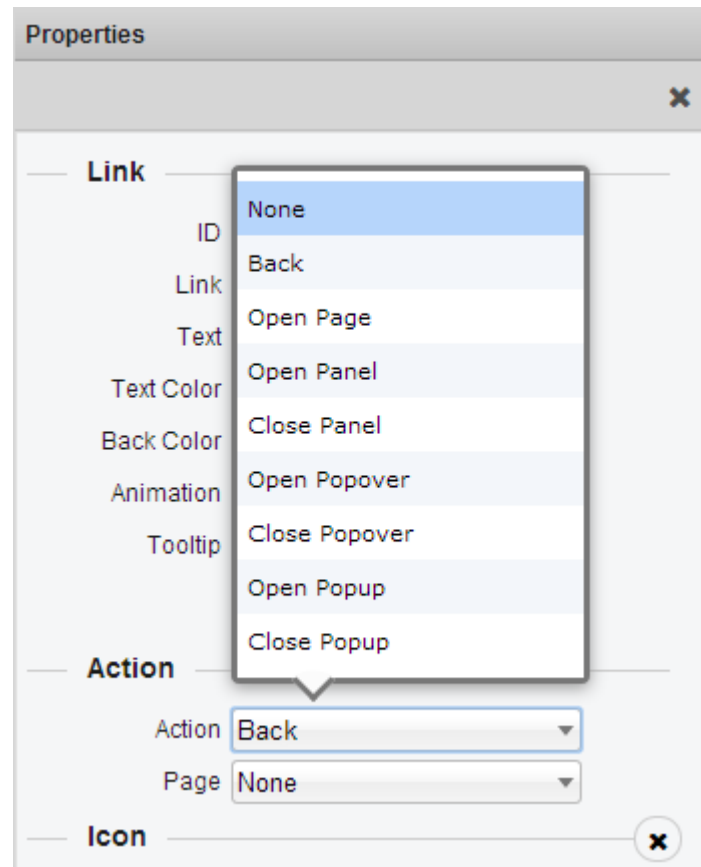
- **Rich UI property inspectors** - The property inspectors offer amazing controls and allow you to enter values as easy as possible, packing rich UI controls like amazing color, theme and icon pickers.



- Integrated inspectors for nested elements** - The dynamic property inspectors not only give you access to the base properties, some of them even offer direct access to nested components properties like icons and other media elements. So you have a great overview of the current component and its children.

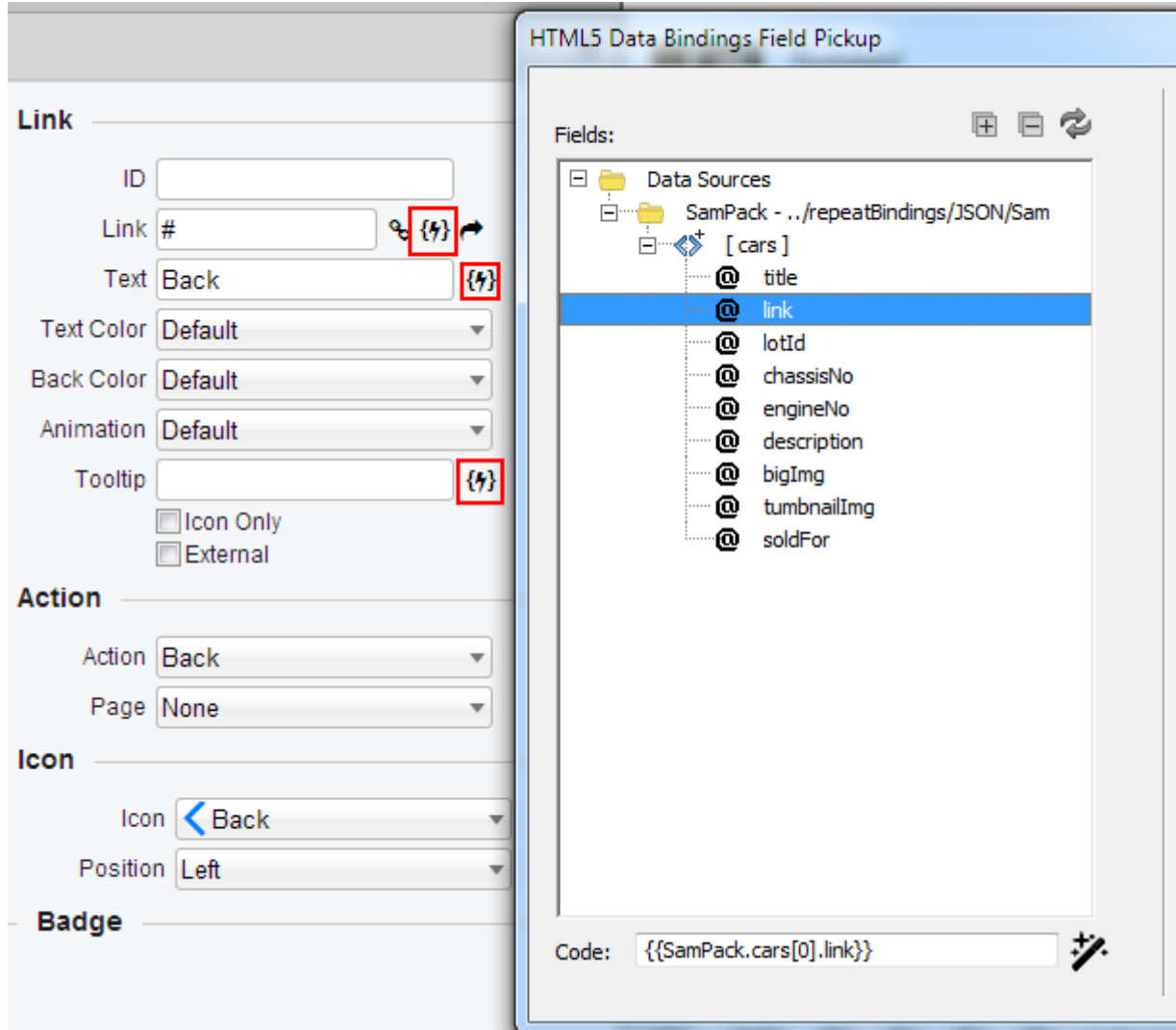


- **Visual action based programming design** - Every component has its own actions and those can be easily selected from the property inspector. Wherever you want a navigation action, open or close panels or popups it is all there a single click away – no coding whatsoever!



- **Behaviors integration for calling custom actions** - If the built-in actions are not enough – you can just assign any behavior action to any of the components!

- Integrated properties for HTML5 Data Bindings integration** - The UI components offer great integration with HTML5 Data Bindings – you can bind dynamic data or make repeating regions and elements directly from the dedicated property inspectors!





- **DMXzone Database Connector for dynamic data query and population** - Want to populate you app with data from your own remote database? It is all possible with the DMXzone Database Connector. Just define a remote connection to a dedicated data API site in Dreamweaver and you are ready to go!

The image shows the DMXzone Visual App Designer interface. On the left is the 'Link' configuration panel, and on the right is the 'HTML5 Data Bindings Field Pickup' dialog.

**Link Configuration Panel:**

- Link:** ID (text input), Link # (text input with a link icon), Text (text input: 'Back'), Text Color (dropdown: 'Default'), Back Color (dropdown: 'Default'), Animation (dropdown: 'Default'), Tooltip (text input: '{') with a red box around it, and checkboxes for 'Icon Only' and 'External'.
- Action:** Action (dropdown: 'Back'), Page (dropdown: 'None').
- Icon:** Icon (dropdown: '< Back'), Position (dropdown: 'Left').
- Badge:** (empty section).

**HTML5 Data Bindings Field Pickup Dialog:**

- Fields:** A tree view showing data sources. The selected path is: Data Sources > SamPack - ../repeatBindings/JSON/Sam > products - products > [ data ] > @ productDescription. The 'productDescription' field is highlighted in blue.
- Code:** The binding code is `{{products.data[0].productDescription}}`.

- **DMXzone Database Updater for performing actions as database updates** - If you want to perform any data manipulation on your remote database, it is all possible with the DMXzone Database Updater – actions are directly performed and triggered by the UI Components.



# Database Updater Action Executor

Connect Your Database Update Action

General

About

## Define Database Action Executor

### Available Data Bindings:

- \$FORM
  - @ customerName
  - @ contactLastName
  - @ contactFirstName
- Data Sources
  - Classicmodels - Classicmodels
- Repeat Regions
  - repeat1

Executor Name: insertcustomers

Database Action: insertCustomers

Site: repeatBindings

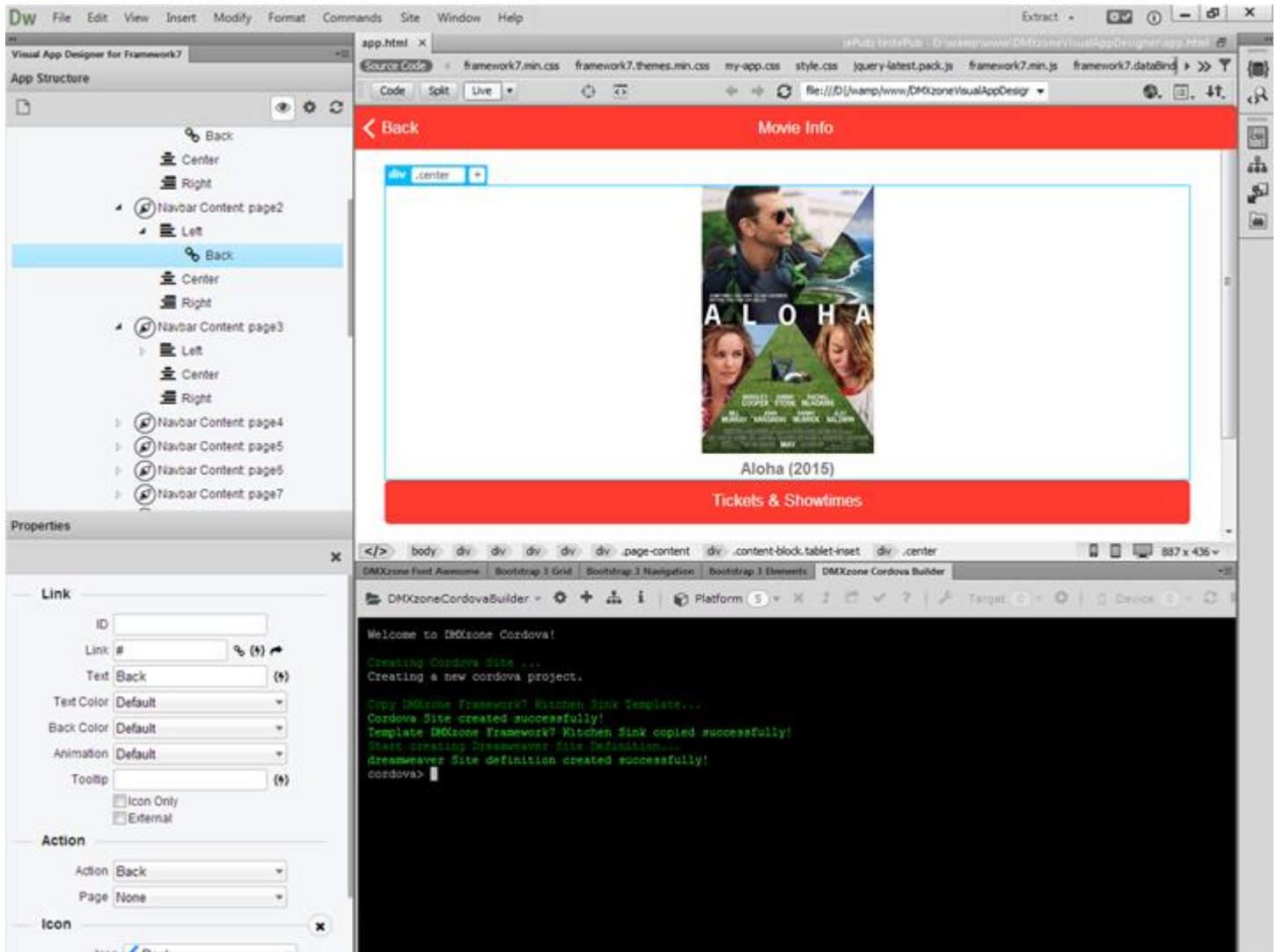
Columns Options

Name	Value	Req
customerName	{{FORM.customerName}}	
contactLastName	{{FORM.contactLastName}}	
contactFirstName	{{FORM.contactFirstName}}	

contactFirstName: {{FORM.contactFirstName}}

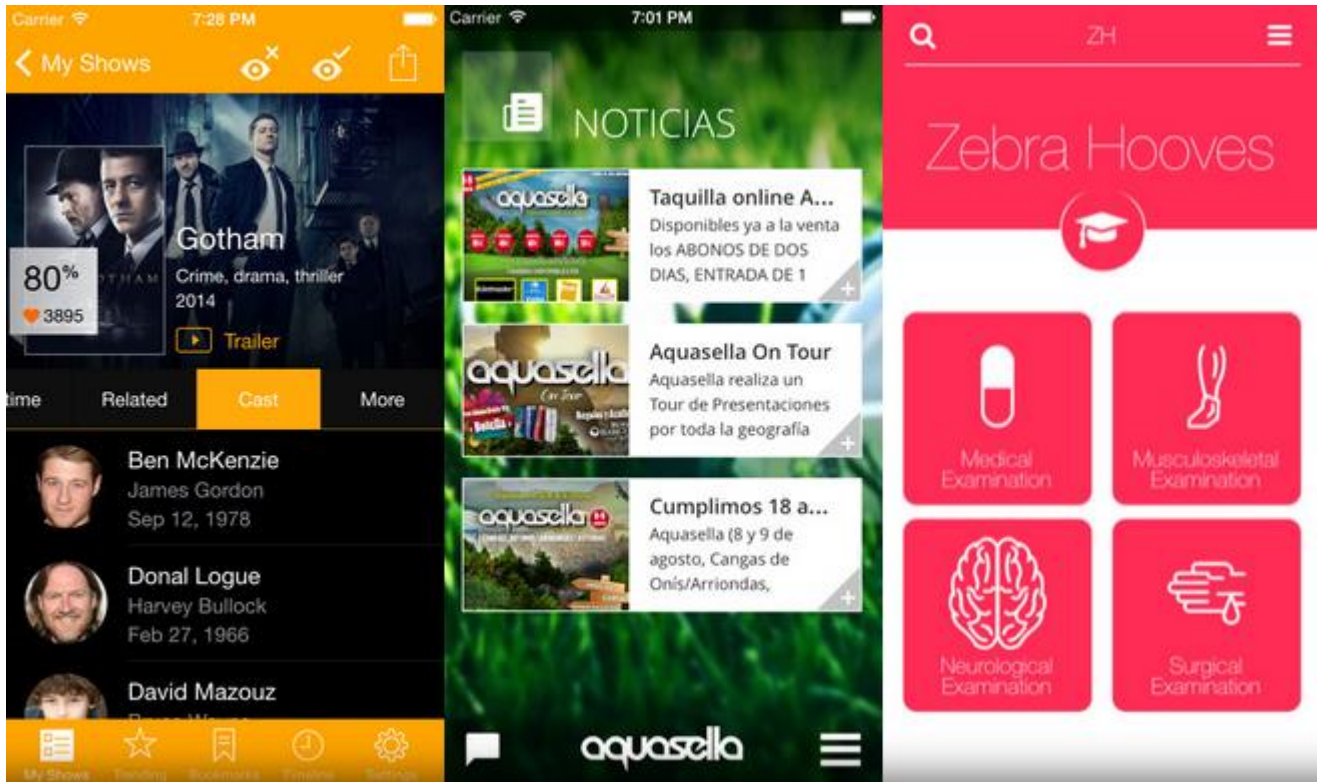
Enter the name of this Database Action Executor.

- **Great integration with DMXzone Cordova Builder**- Once you're ready with your app design in DMXzone Visual App Builder, define, generate and build your mobile app with a few clicks in DMXzone Cordova Builder for any popular platform.

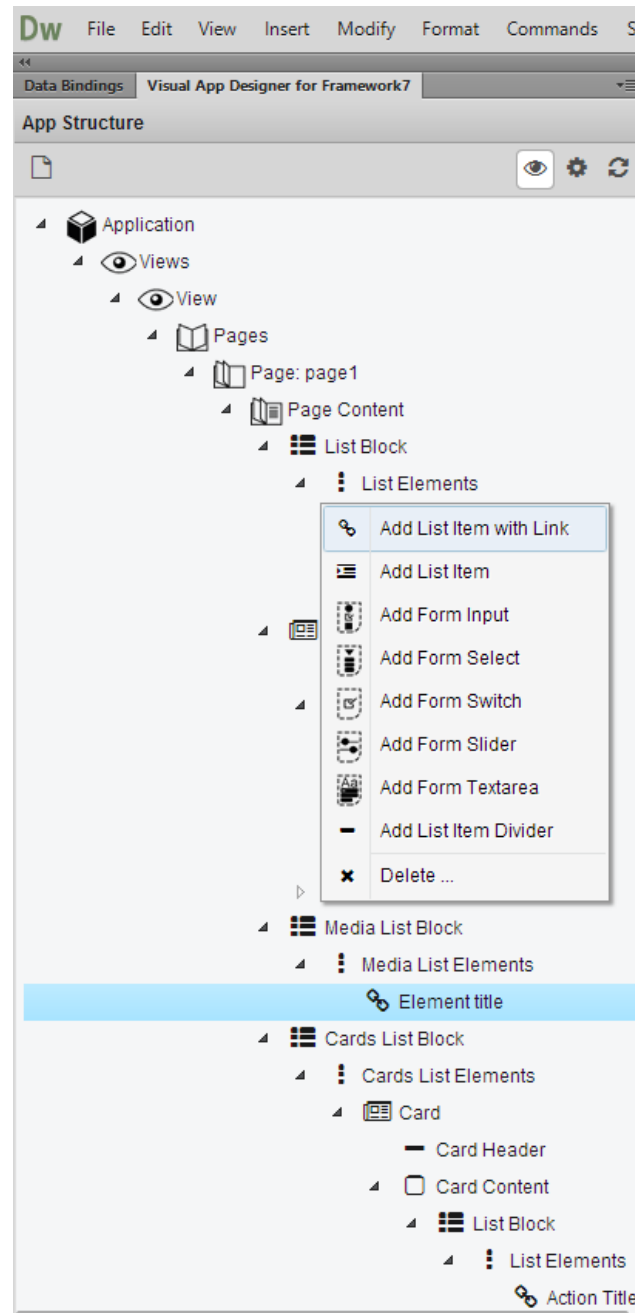


## Framework 7 Features in Visual App Designer

- **Ultra easy to use. Not harder than website** - To create apps using Framework7 is as easy as website creation. Try to start and you will be surprised how easy it is.



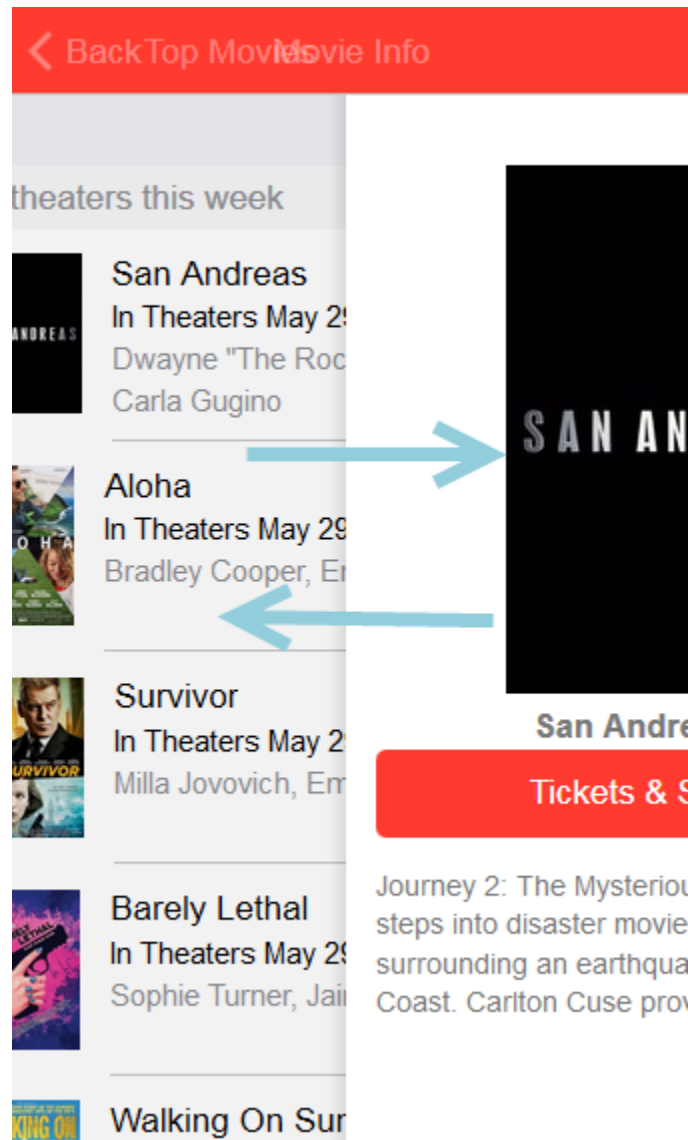
- **Awesome UI components to choose from** - Framework7 comes with a bunch of ready to use UI elements like buttons, popup, popover, list views, media lists, cards, preloader (activity indicator), form elements, etc.
  - In Dreamweaver



o In browser

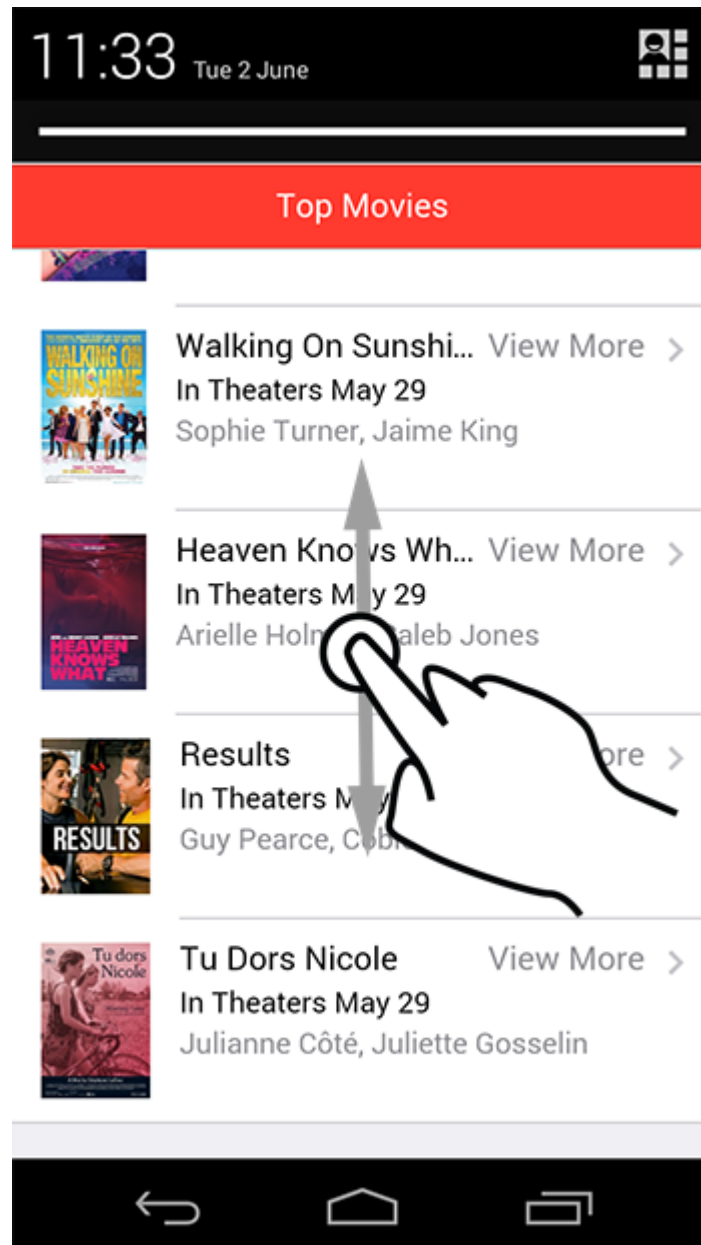
<p> Gender</p> <p> Birth date</p> <p> Date time</p> <p> Switch</p> <p> Slider</p> <p> Textarea</p>	<p>Button   Button</p> <p>Button   Button</p> <p><b>BIG BUTTONS</b></p> <p>Active</p> <p><b>THEMED FILL BUTTONS</b></p> <p>Submit</p> <p><b>LIST-BLOCK BUTTONS</b></p> <p>List Bu</p> <p>List Bu</p>	<p><b>MIXED AND NESTED</b></p> <p> Ivan Petrov</p> <p>  Two icons here</p> <p>No icons here</p> <p> Ivan Petrov</p> <p>  Two icons</p> <p>No icons here</p> <p> Ultra long text</p> <p> With switch</p> <p> Ultra long text goes</p> <p> With switch</p>	<p><input checked="" type="checkbox"/> <b>BOOKS</b></p> <p><input type="checkbox"/> Movies</p> <p><input checked="" type="checkbox"/> Food</p> <p><input checked="" type="checkbox"/> Drinks</p> <p><b>RADIO BUTTONS GROUP</b></p> <p>Books</p> <p>Movies</p> <p>Food</p> <p>Drinks</p> <p><b>WITH MEDIA LISTS</b></p>
--	--	--	--

- **Full support for swipe back** - One of Framework7' killing feature is supporting of iOS well known swipe back gesture from left border of screen when you want to get to the previous page. It simply works, and works perfectly as you expect it to do. Just swipe from left (or drag with mouse) area of the page to see smooth transition to the previous page. It is not just A-B transition. It really follows your finger with parallax animation while touch!





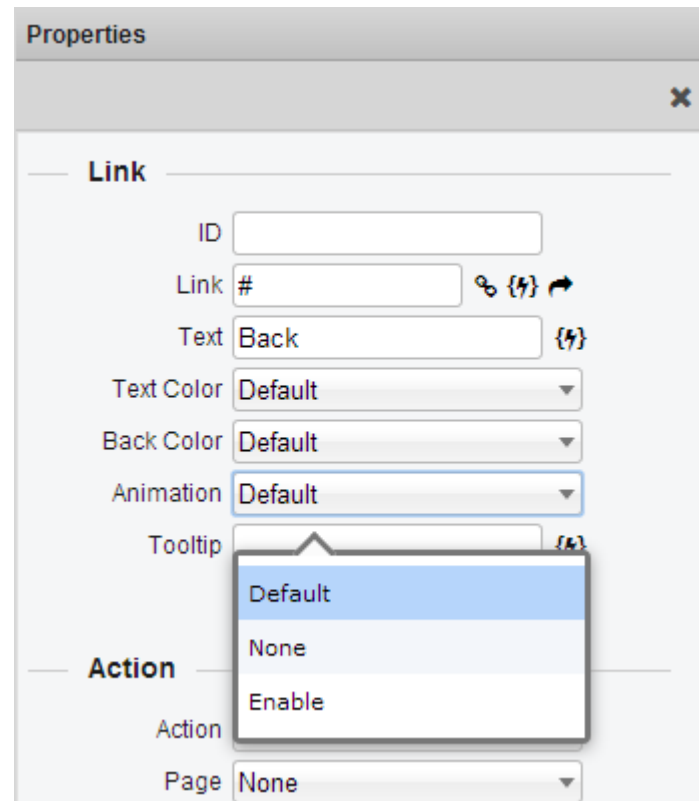
- **Native scrolling** - It can sound even curious but one of the most important Framework7 feature is that it uses only native scrolling. Moreover - it multiplies its advantages! So you can still have this awesome scrolling with momentum and resistance bounce without any scrolling issues!



- **High-performance animation** - It is all about performance and Framework7 uses only high performance hardware accelerated css animations and transitions to achieve the best result.



- **Awesome pages animation** - One of the main targets of Framework7 is to have native look and feeling of iOS application. And Framework7 is the only framework that solves it and offers 1:1 page animation with smooth parallax effect, changing opacity and shadow when loading new page.



- **XHR + Caching + History + Preloading** - This is an unbeatable combination to make your app routing as perfect as possible. Framework7 is ready to route your app pages using XHR (optional) with its own internal methods and internal HTML configurable caching that allow loading pages faster and saving a lot of time and traffic for your users! It loads page via Ajax only ones per specified caching period (10 minutes by default), other times it just gets it from memory. It allows making further requests immediately!

- **10 stunning iOS color themes to style your apps** - The extension features 10 color themes to choose from for your apps. The applied color theme affects the interactive elements such as links, buttons, form elements, icons.

< Back
Movie Info




**Aloha (2015)**

Tickets & Showtimes

Bradley Cooper stars as a defense worker who teams up with a pilot (Emma Stone) to stop a satellite launch in this romantic comedy from Cameron Crowe. Rachel McAdams co-stars.

< Back
Movie Info



**San Andreas (2015)**

Tickets & Showtimes

Journey 2: The Mysterious Island's Brad Peyton steps into disaster movie territory with this 3D film surrounding an earthquake that hits the West Coast. Carlton Cuse provides the script.

< Back Movie Info

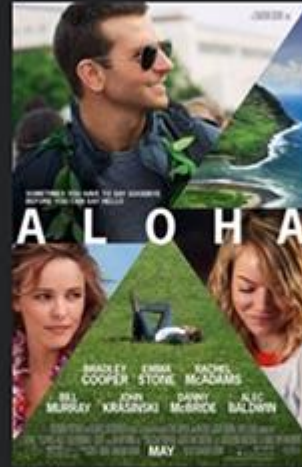


San Andreas (2015)

Tickets & Showtimes

Journey 2: The Mysterious Island's Brad Peyton steps into disaster movie territory with this 3D film surrounding an earthquake that hits the West Coast. Carlton Cuse provides the script.

< Back Movie Info











Aloha (2015)

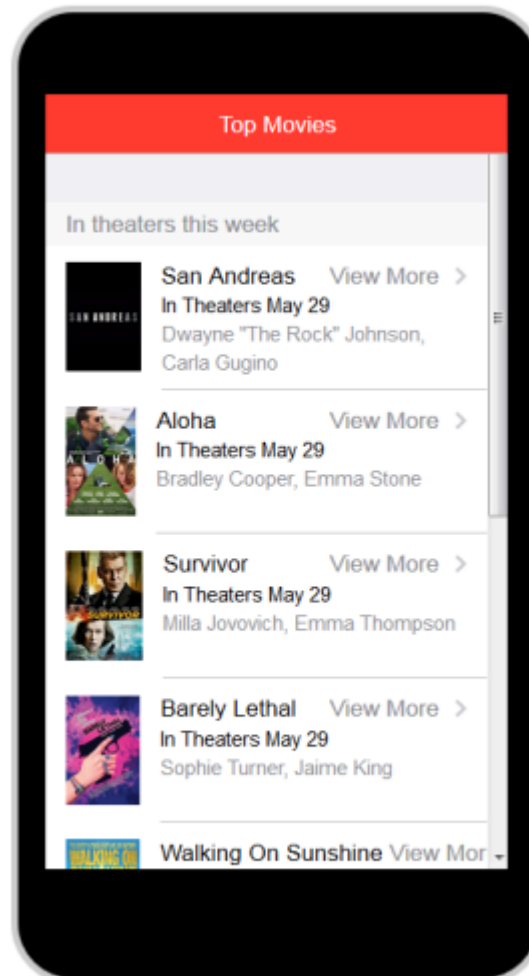
Tickets & Showtimes

Bradley Cooper stars as a defense worker who teams up with a pilot (Emma Stone) to stop a satellite launch in this romantic comedy from Cameron Crowe. Rachel McAdams co-stars.

- **2 beautiful layout themes** - There are two available layout themes to choose from - white and dark.

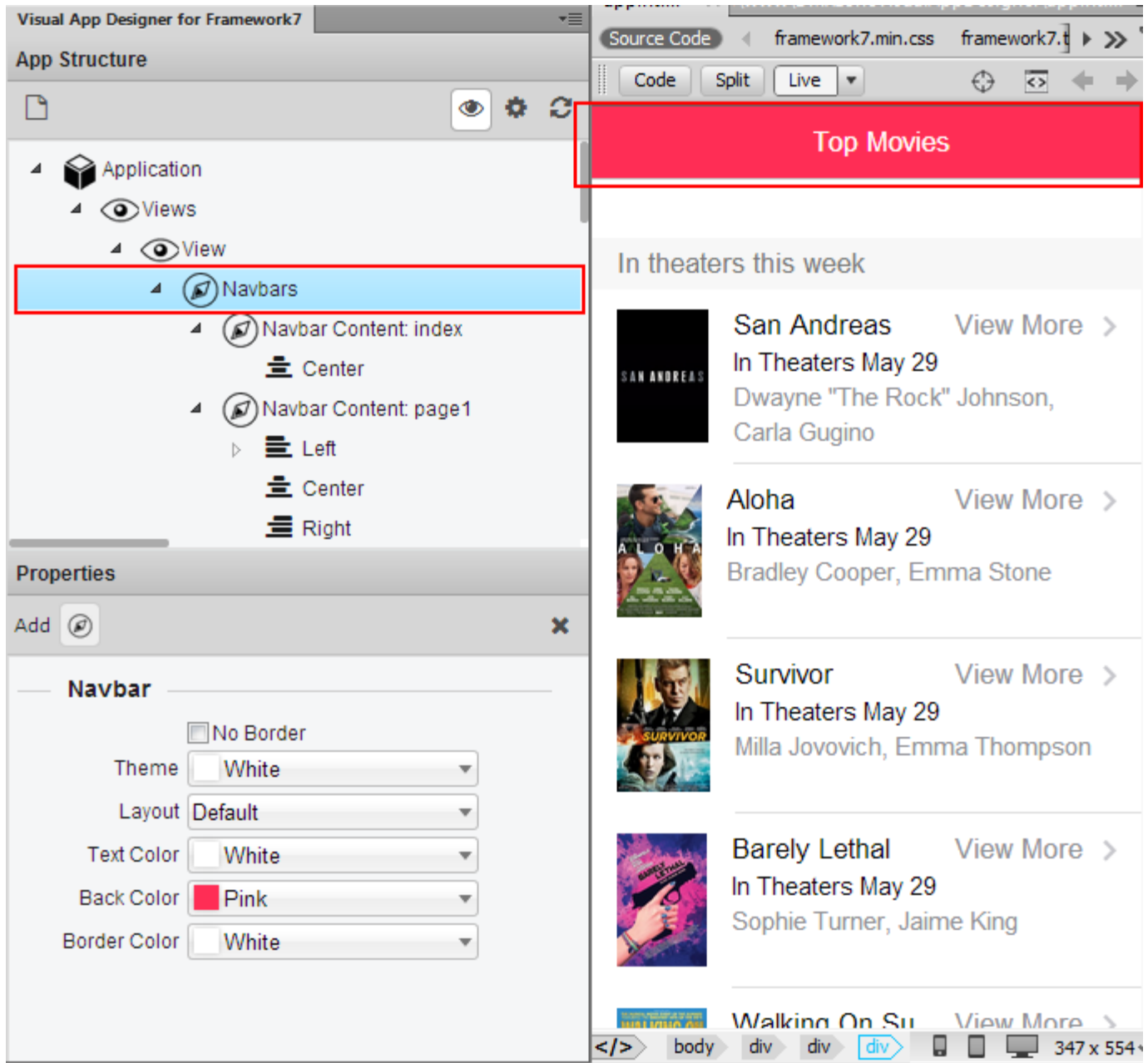
Top Movies	Top Movies
<p>In theaters this week</p> <div style="border-bottom: 1px solid #ccc; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>San Andreas</b></p> <p>In Theaters May 29</p> <p>Dwayne "The Rock" Johnson, Carla Gugino</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="border-bottom: 1px solid #ccc; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Aloha</b></p> <p>In Theaters May 29</p> <p>Bradley Cooper, Emma Stone</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="border-bottom: 1px solid #ccc; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Survivor</b></p> <p>In Theaters May 29</p> <p>Milla Jovovich, Emma Thompson</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Barely Lethal</b></p> <p>In Theaters May 29</p> <p>Sophie Turner, Jaime King</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div>	<p>In theaters this week</p> <div style="border-bottom: 1px solid #444; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>San Andreas</b></p> <p>In Theaters May 29</p> <p>Dwayne "The Rock" Johnson, Carla Gugino</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="border-bottom: 1px solid #444; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Aloha</b></p> <p>In Theaters May 29</p> <p>Bradley Cooper, Emma Stone</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="border-bottom: 1px solid #444; padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Survivor</b></p> <p>In Theaters May 29</p> <p>Milla Jovovich, Emma Thompson</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div> <div style="padding: 5px 0;">  <div style="display: flex; justify-content: space-between;"> <div> <p><b>Barely Lethal</b></p> <p>In Theaters May 29</p> <p>Sophie Turner, Jaime King</p> </div> <div style="text-align: right;"> <p><a href="#">View More &gt;</a></p> </div> </div> </div>

- **Design for iOS but also run on other platforms** - Although the Framework7 design is based on iOS you can build also native apps for other platforms like Android, Blackberry and Windows Phone! Your native app will just looks like an iOS app everywhere!

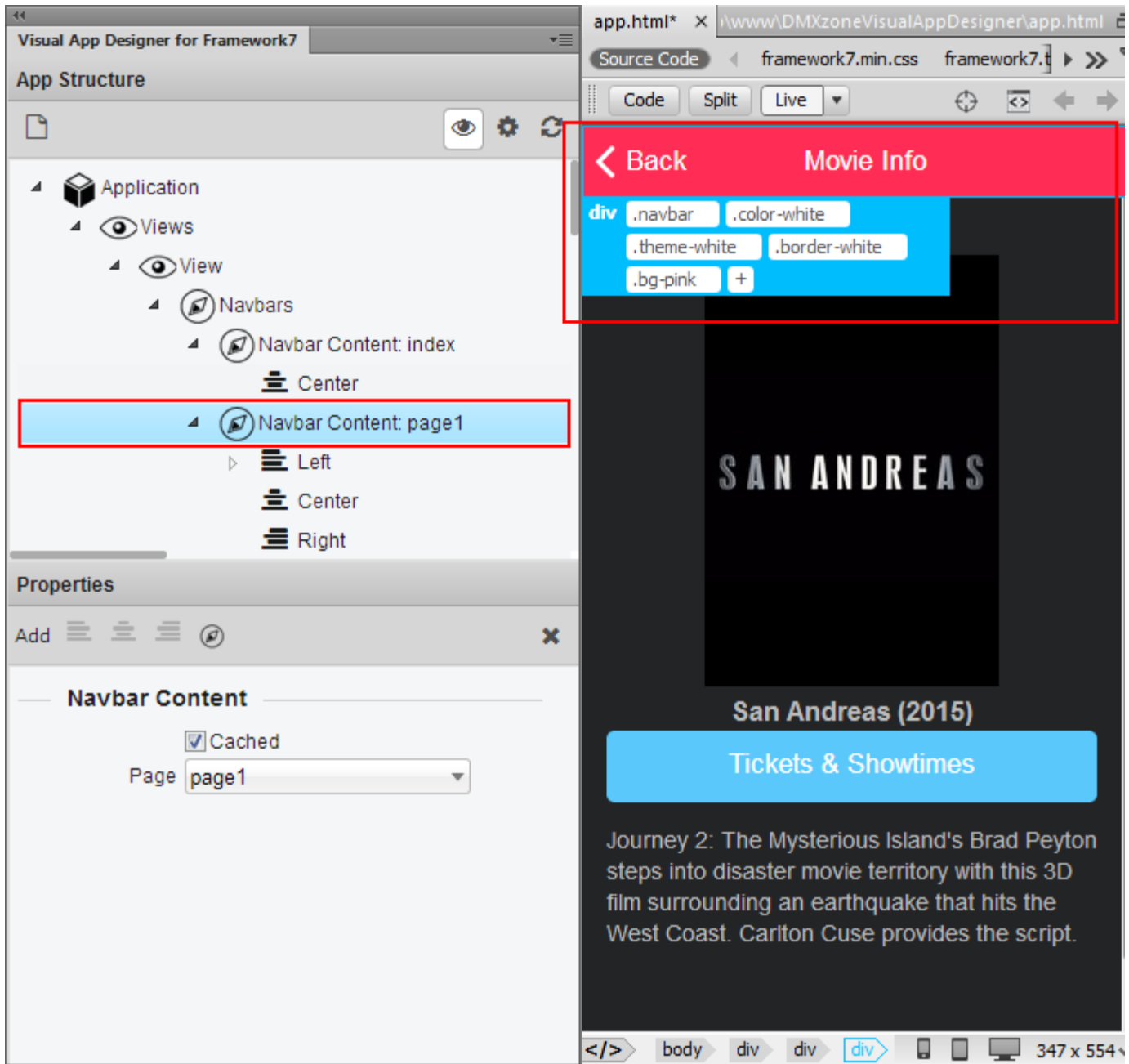


## Components in Visual App Designer

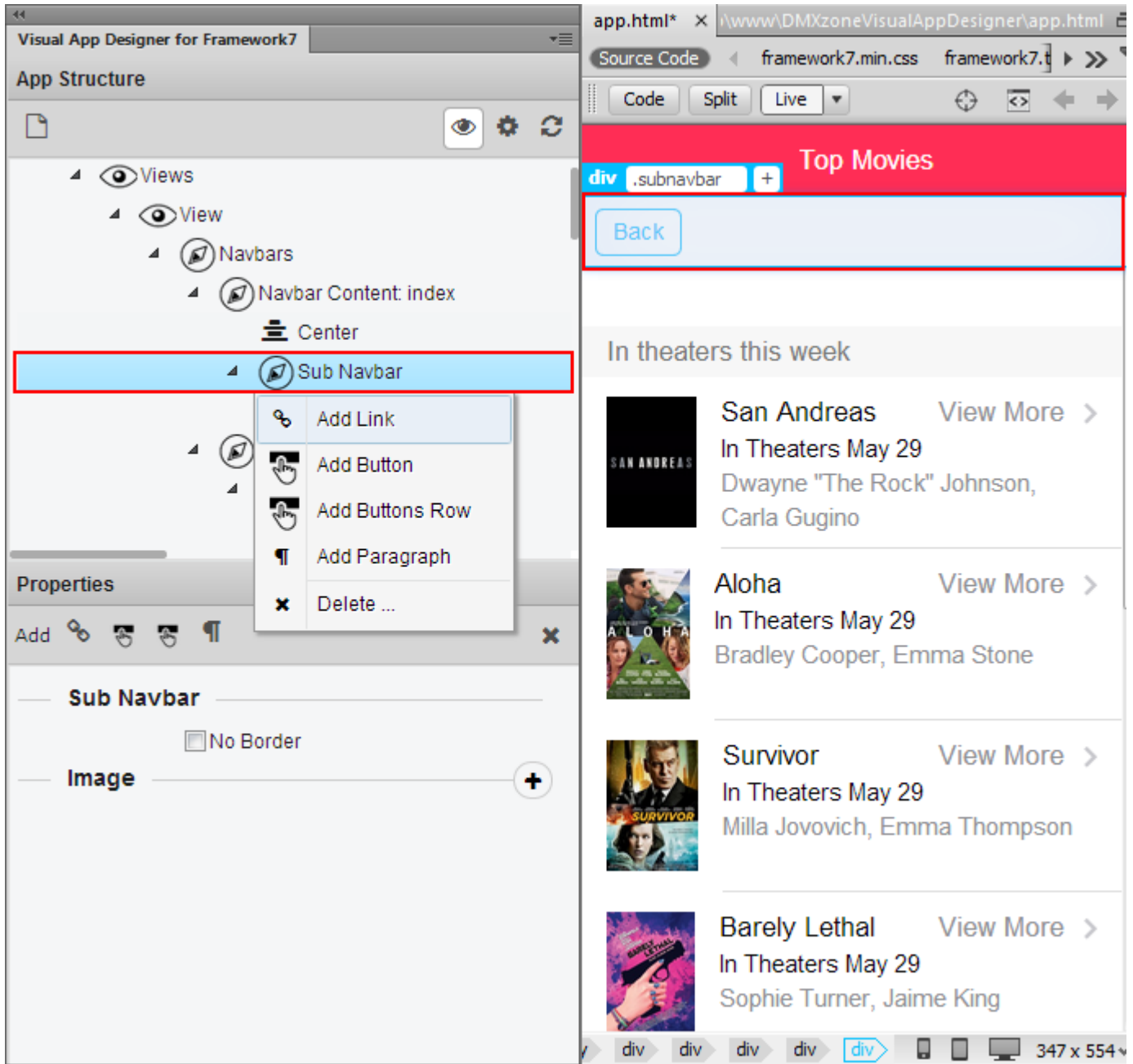
- **Navbar** - Add awesome navigation to your app. You can fix it on the top of the screen with page title and additional navigation elements. It features 3 main parts: left (designed for "back link"), center (for page title) and right (same as left).



- **Dynamic navbar** - Add additional navbar for each page with different navbar elements such as text and links. The layout remains the same as the navbar but you can choose between fade or slide transition effects.

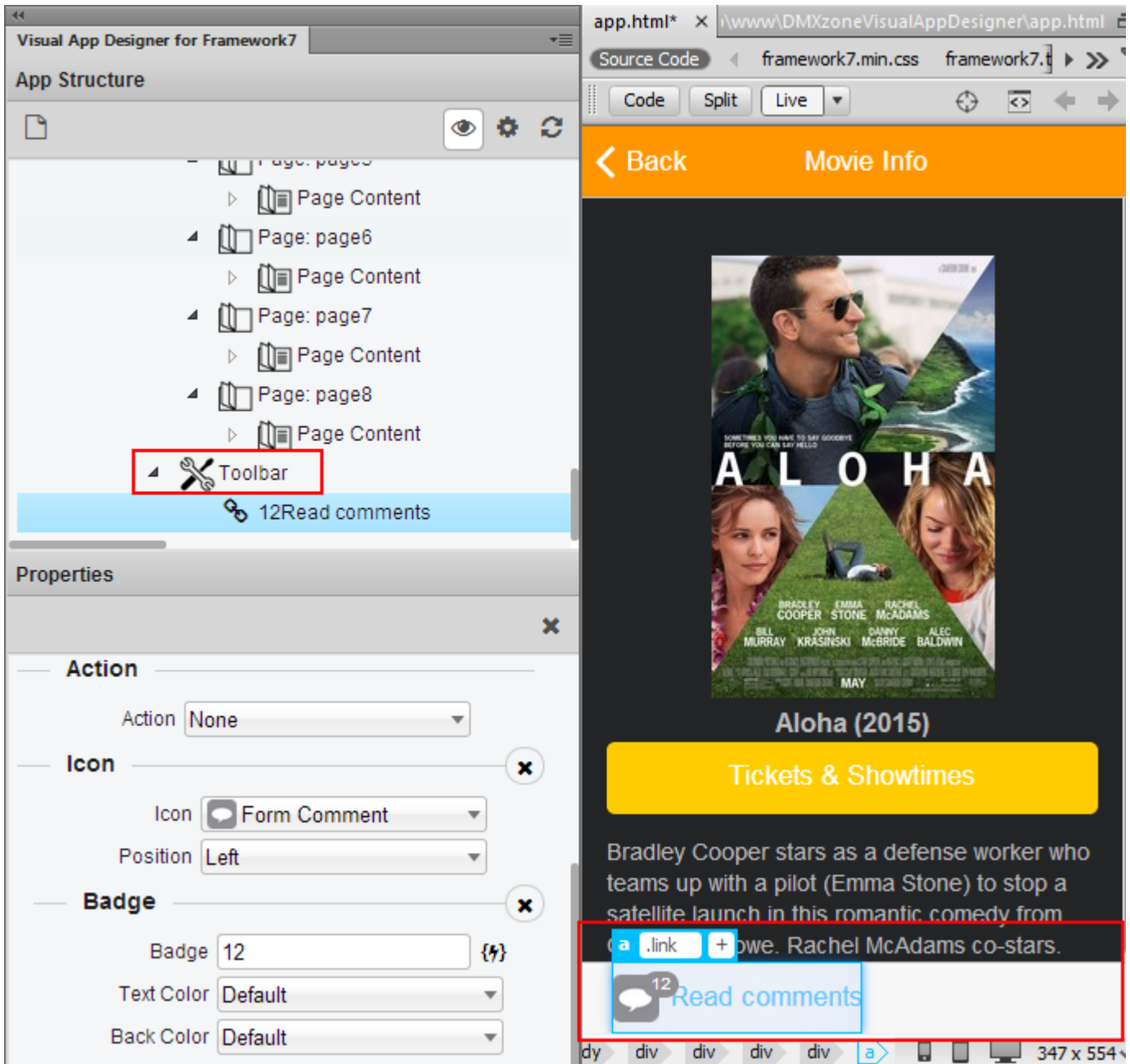


- **Sub navbar** - Sub Navbar is useful when you need to put any additional elements into the navbar, like buttons. It also remains visible when navbar hidden. Sub navbar supports sliding transitions when used with dynamic navbar.

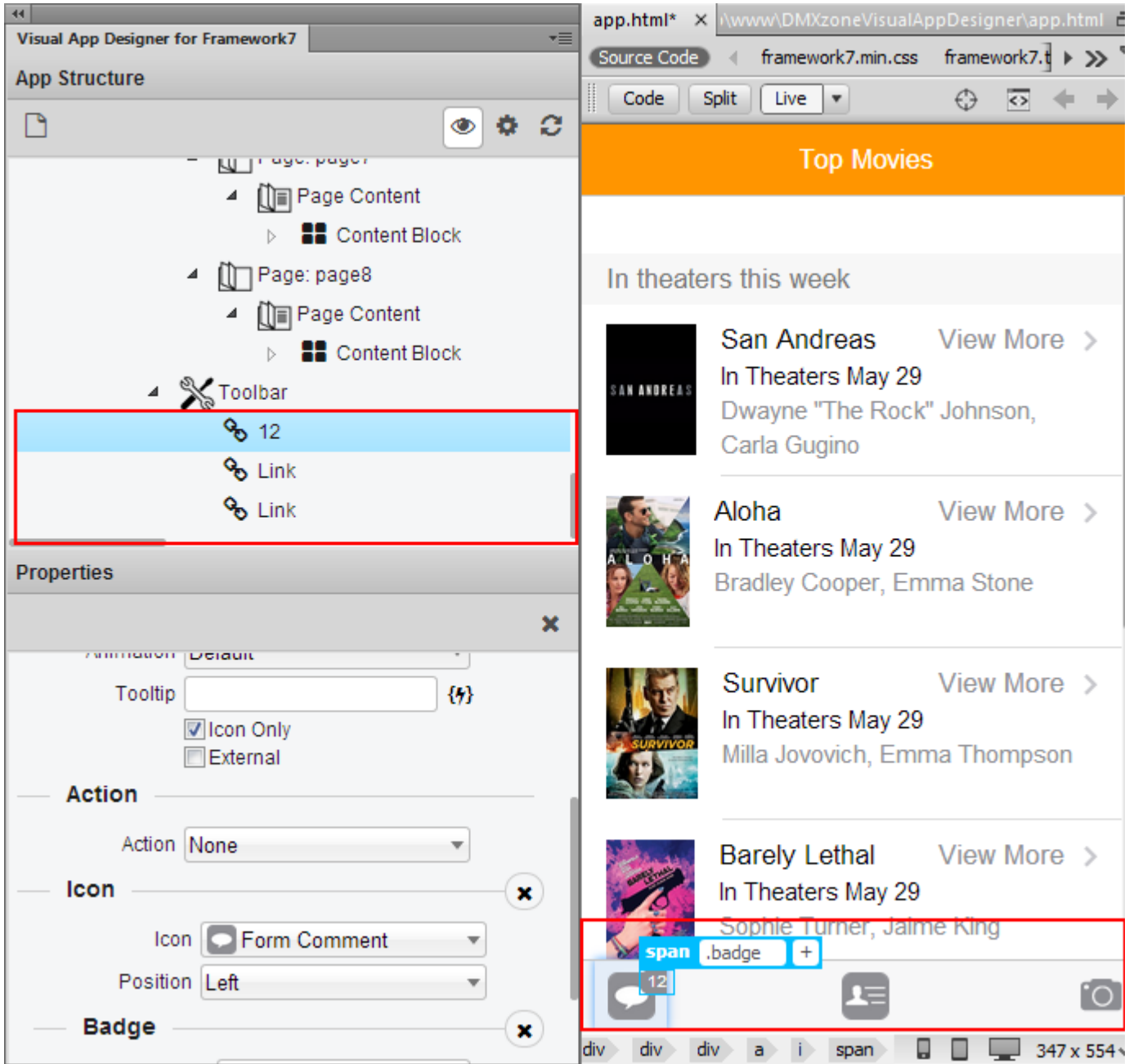




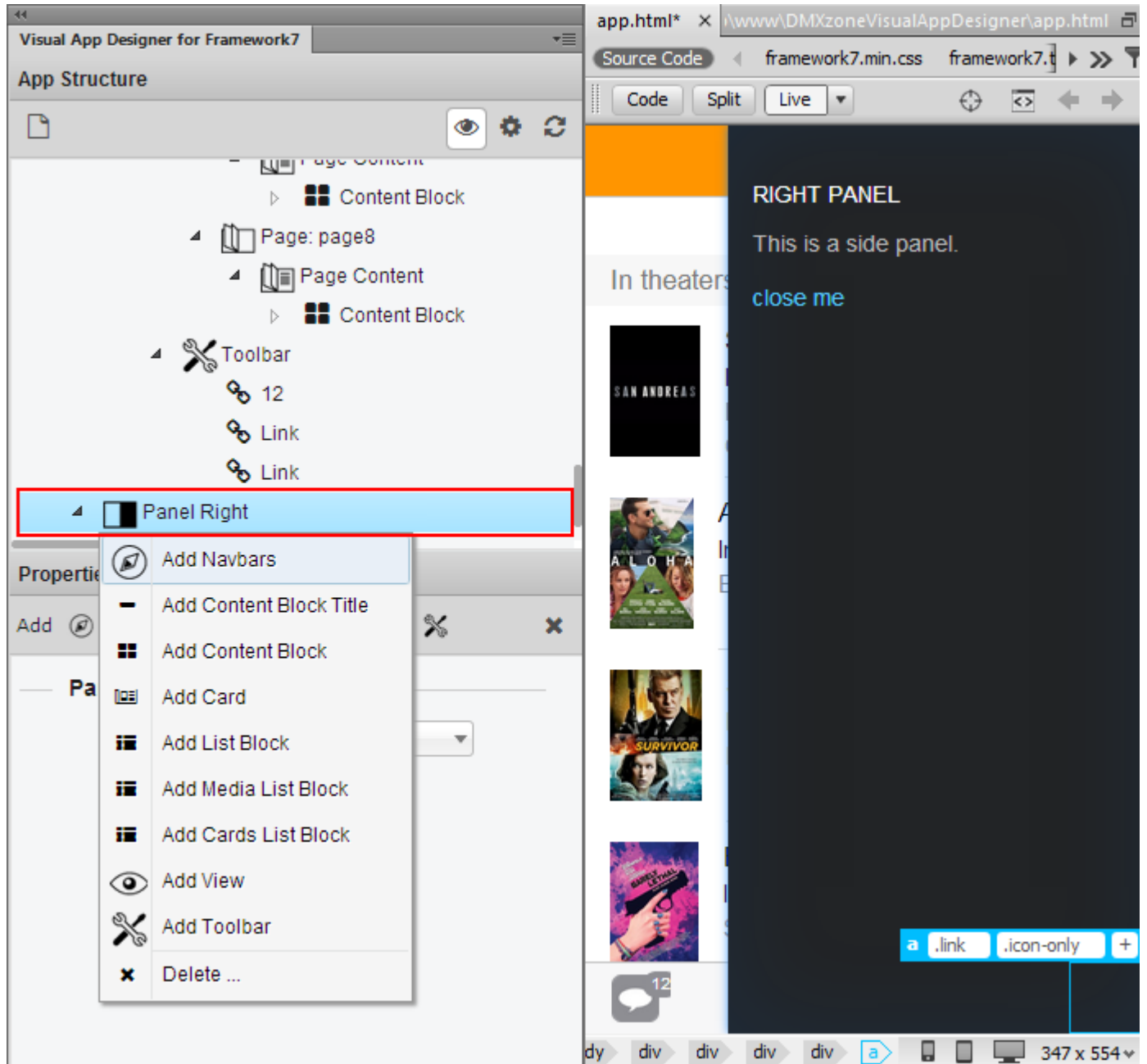
- **Toolbar** - The toolbar is fixed area at the bottom of the screen, where you can add navigation elements, icons, badges and more. All elements are equally spaced along the toolbar with equal space between them.



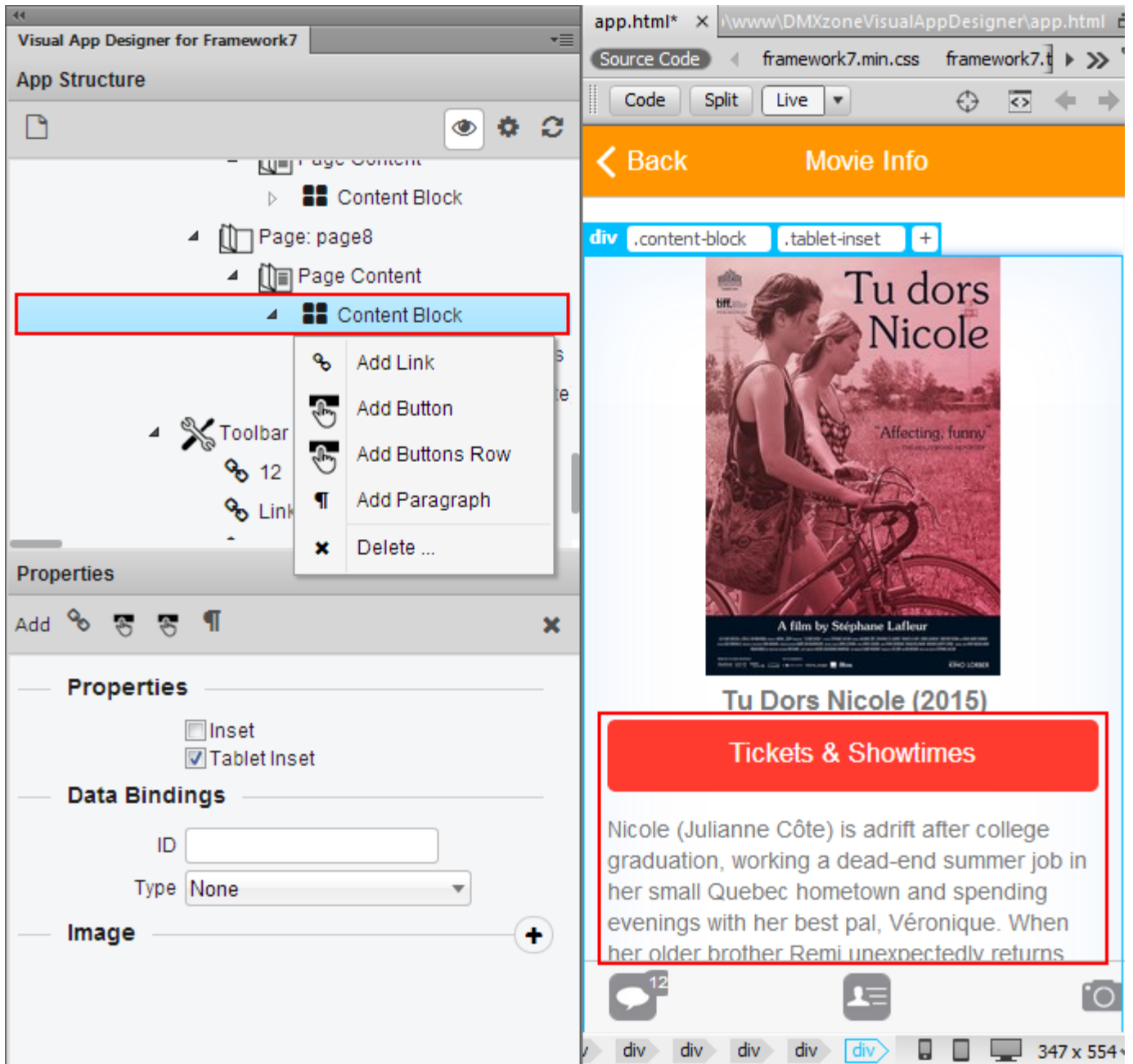
- **Tab bar** - Tab bar is a particular case of toolbar, but it contains icons or icons with labels, instead of plain links.



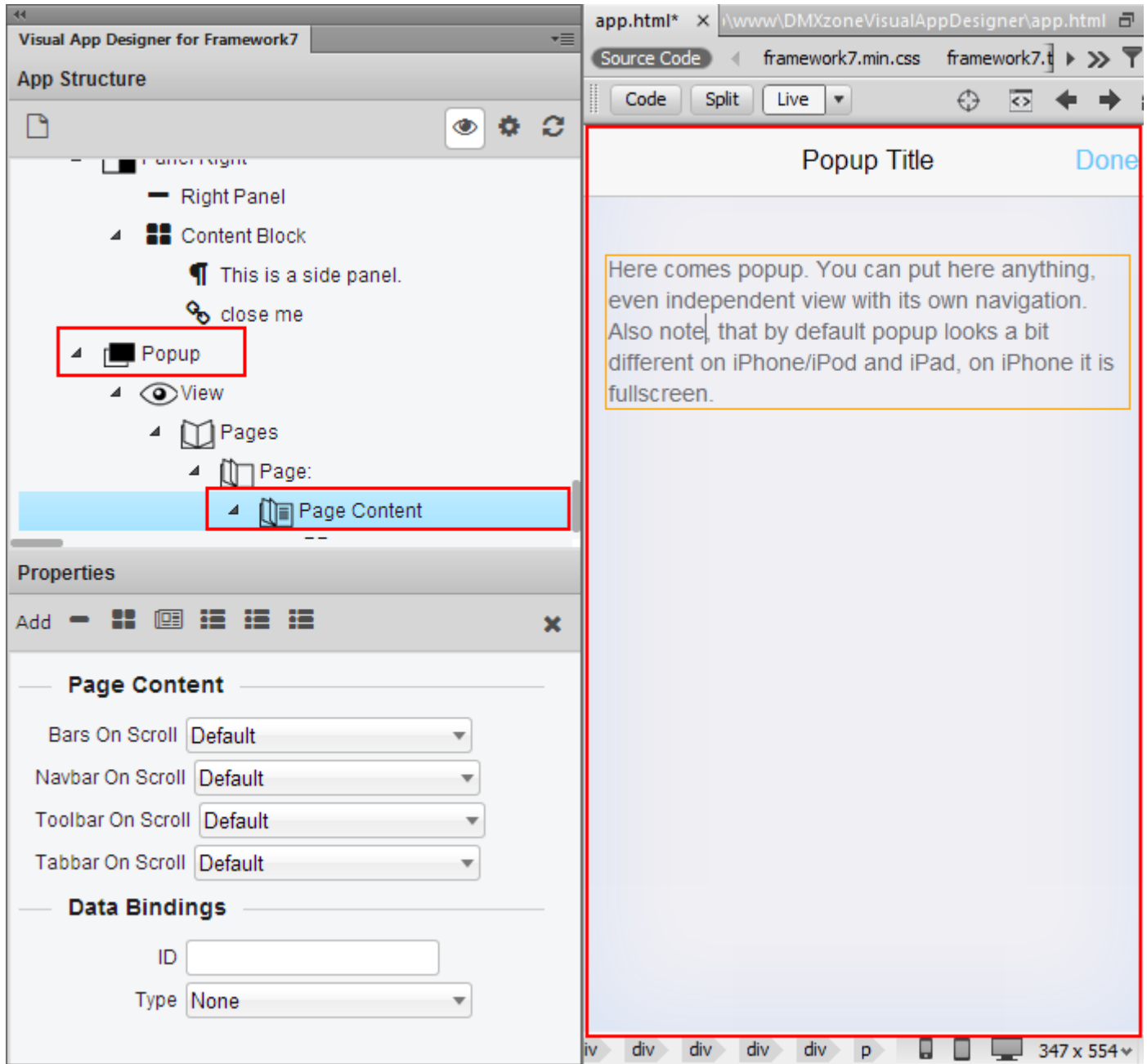
- **Side panels** - You can choose between a right or left side panel for your app. Within the panel you can add any page component.



- **Content block** - This is the place for your content, which can be text, images, buttons and links.



- **Popup** - The popup window pops up over the app's main content. You can use it for example when a user is filling out a form and needs to confirm the input info with a confirm button, text agreement and other.



- **Popover** - Popover component is used to manage the presentation of content in a popover. You can use popovers to present information temporarily in an awesome tooltip-like area. The popover remains visible until the user taps outside of the popover window or you explicitly dismiss it.

The screenshot displays the DMXzone Visual App Designer interface for Framework 7. On the left, the 'App Structure' panel shows a tree view of the application components. A 'Popover' component is highlighted with a red box. A context menu is open over the 'Popover' component, listing options: 'Add Link', 'Add Button', 'Add Buttons Row', 'Add Paragraph', and 'Delete ...'. Below the context menu, the 'Properties' panel for the 'Popover' component is visible, showing 'ID popover1' and a checked 'Popover Links' property.

On the right, the 'Live' preview window shows a mobile application interface. The top section is titled 'Top Movies' and features a list of movies under the heading 'In theaters this week'. The first movie listed is 'San Andreas', with the text 'In Theaters May 29' and 'Dwayne "The Rock" Johnson, ...'. A blue popover is overlaid on the 'San Andreas' movie card, containing the text 'Here you can add additional info'. The popover has a blue header with the text 'div .popover .modal-in #popover1' and a plus sign icon. The background of the application is dimmed to show the popover clearly.

- List view** - List views are versatile and powerful user interface components frequently found in iOS apps. A list view presents data in a scrollable list of multiple rows that may be divided into sections/groups. You can use list view to let your users navigate through hierarchically structured data, present an indexed list of items, display detail information and controls in visually distinct groupings or present a selectable list of options.

The screenshot displays the Visual App Designer interface for Framework 7. On the left, the 'App Structure' pane shows a tree view with 'Sub View' containing 'Navbars', 'Pages', and 'Page: list-view'. Under 'Page: list-view', there is 'Page Content' which includes 'Content Block', 'Data list, with icons', 'List Block', and 'List Elements'. The 'Properties' pane below shows settings for the 'Page' (Name: list-view, Theme: Default, Layout: Default) and 'Data Bindings' (ID: ).

The right side shows a live preview of the 'List View' in a mobile browser. The page has a 'Back' button and a title 'List View'. A text box contains the text: 'Framework7 allows you to be flexible with list views (table views). You can make them as navigation menus, you can use there icons, inputs, and any elements inside of the list, and even make them nested:'. Below this, there are two sections: 'DATA LIST, WITH ICONS' and 'LINKS'. The 'DATA LIST, WITH ICONS' section contains three rows: 'Ivan Petrov CEO', 'John Doe 5', and 'Jenna Smith'. The 'LINKS' section contains three rows: 'Ivan Petrov CEO >', 'John Doe Cleaner >', and 'Jenna Smith >'. The bottom status bar shows the device dimensions as 347 x 554.



- **Media list view** - Media list view is extended case of list view where you can add additional elements such as images. It's useful for comment threads or articles lists.

The screenshot displays the Visual App Designer interface for Framework 7. On the left, the 'App Structure' pane shows a tree view where the 'Media List Block' is selected and highlighted with a red box. Below it, the 'Properties' pane shows configuration options for the 'Media List Block', including checkboxes for 'Inset' and 'Tablet Inset', and dropdown menus for 'Theme' (set to 'Default') and 'Layout' (set to 'Default').

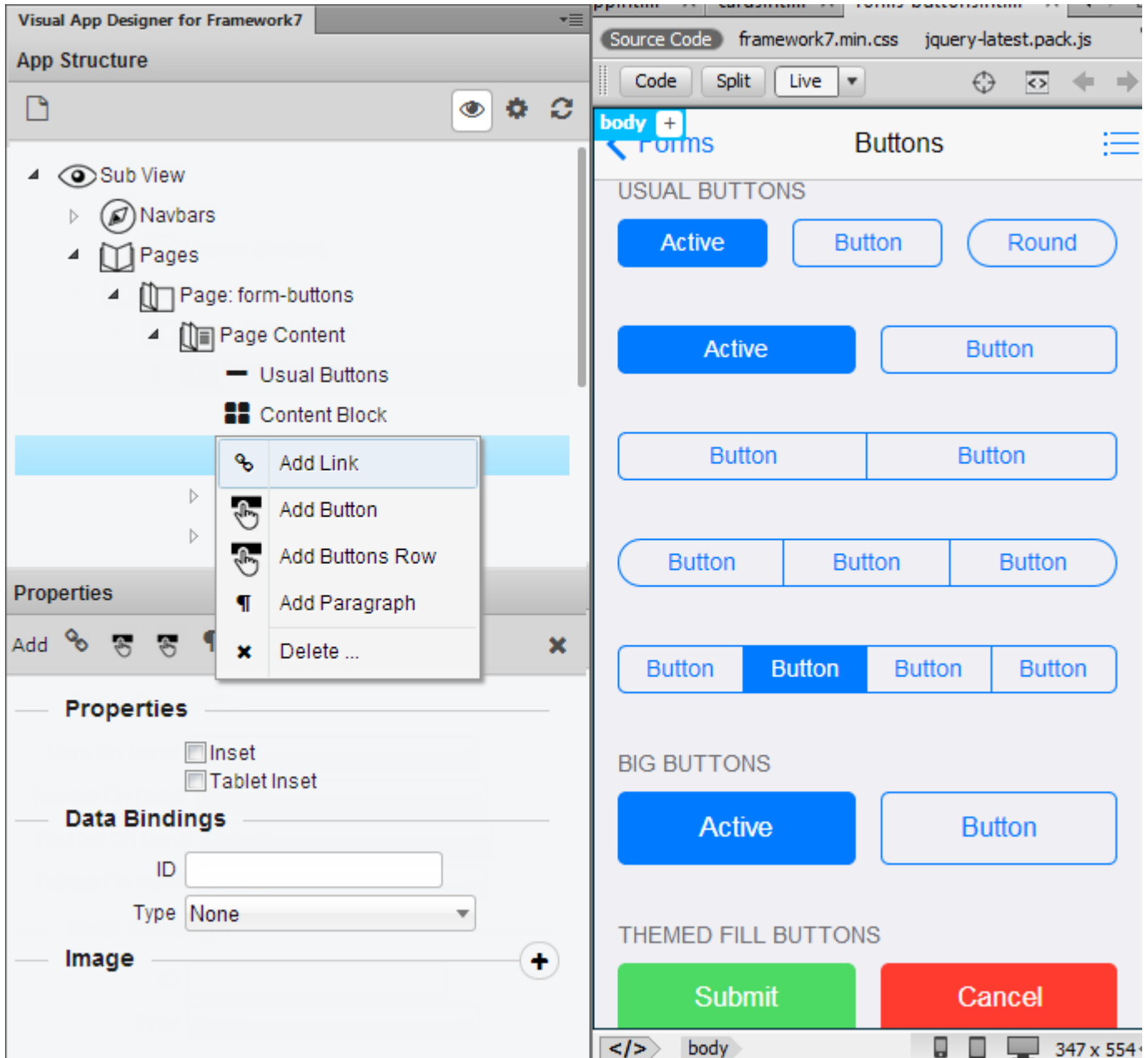
On the right, a browser window shows the rendered app view. The page has a blue header 'Top Movies' and a section titled 'In theaters this week'. A red box highlights the first movie entry: 'San Andreas', featuring a movie poster, the title, the release date 'In Theaters May 29', and the cast 'Dwayne "The Rock" Johnson, Carla Gugino'. Below this, other movie entries are visible, including 'Aloha', 'Survivor', and 'Barely Lethal', each with a poster, title, release date, and cast.



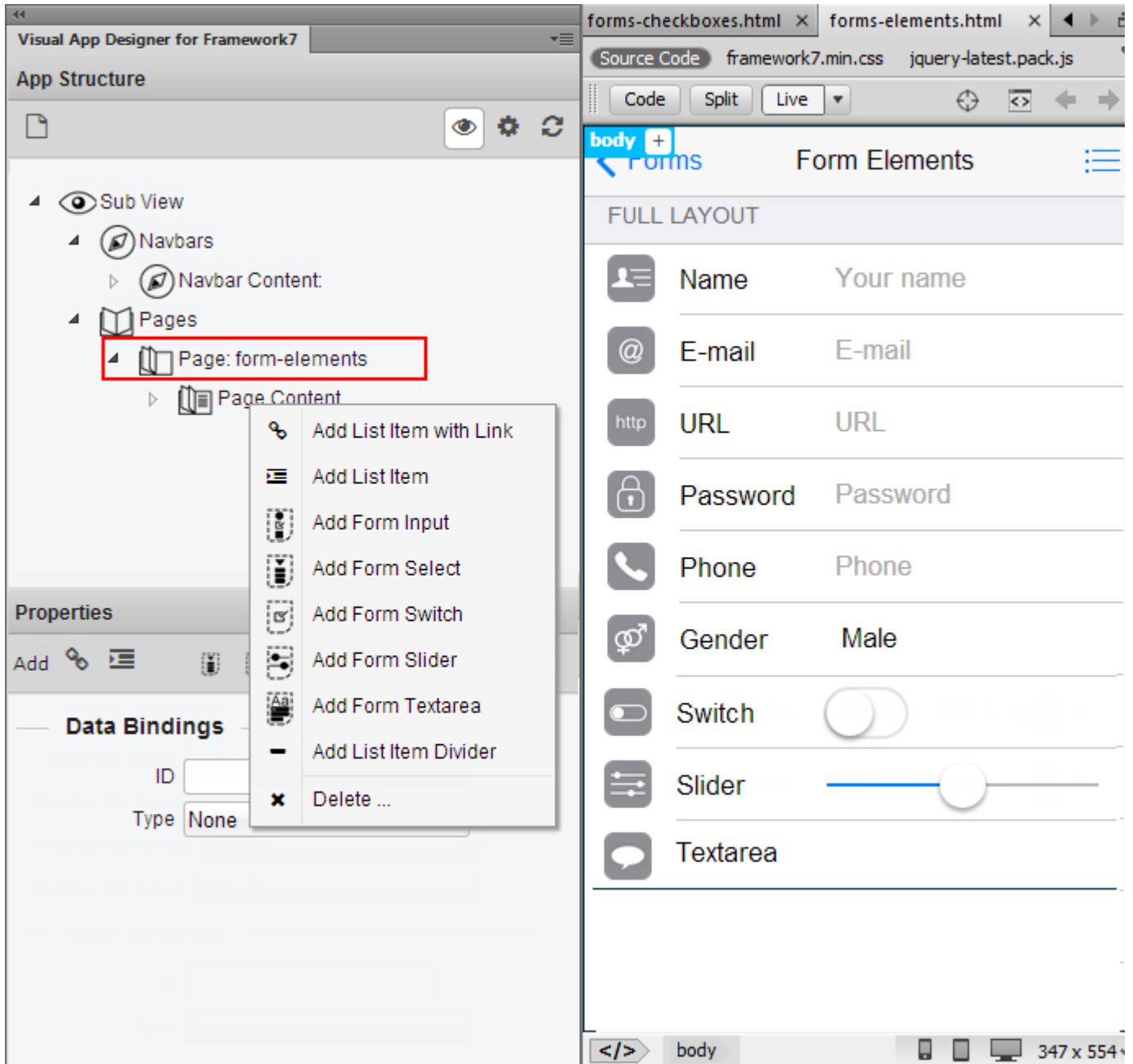
- **Cards** - Cards, along with list view, is a one more great way to contain and organize your information. They are typically an entry point to more complex and detailed information. Along with the inner area you can add header and footer.

The screenshot displays the Visual App Designer for Framework 7 interface. On the left, the **App Structure** pane shows a tree view with the following hierarchy: Pages > Page: cards > Page Content > Content Block > Cards, along with List View, is a one more great way to contain and organize your information. Below this are Simple Cards, Card, Card Content, Card, and Card. The **Properties** pane below it shows settings for Page Content (Bars On Scroll, Navbar On Scroll, Toolbar On Scroll, Tabbar On Scroll) and Data Bindings (ID, Type). The right side shows a live preview of the 'Cards' page, which includes a 'Back' button, a title bar, and a content area with text and a 'SIMPLE CARDS' section containing a card with a header and footer.

- **Buttons** - You can add normal or big buttons, active or whole buttons row. From the properties you can choose to fill the button with color and the text color.



- **Forms** - The included form elements allow you to create flexible and beautiful form layouts within your app.



## **Before you begin**

Make sure to setup your site in Dreamweaver first before creating a page. If you don't know how to do that please [read this](#).

## **Installing the extension**

Read the tutorial at <http://www.dmxzone.com/go/4671/how-to-install-extensions-from-dmxzone/>

Login, download the product, and look up your serial at <http://www.dmxzone.com/user/products>

Get the latest DMXzone Extension Manager at <http://www.dmxzone.com/go/22670/dmxzone-extension-manager-for-dreamweaver>

Open DMXzone Extension Manager and go to your Purchased products tab and click install. If you have a subscription, the extension is to be found and installed in the Explore tab.

Get the latest extension manager from Adobe at [http://www.adobe.com/exchange/em\\_download/](http://www.adobe.com/exchange/em_download/)

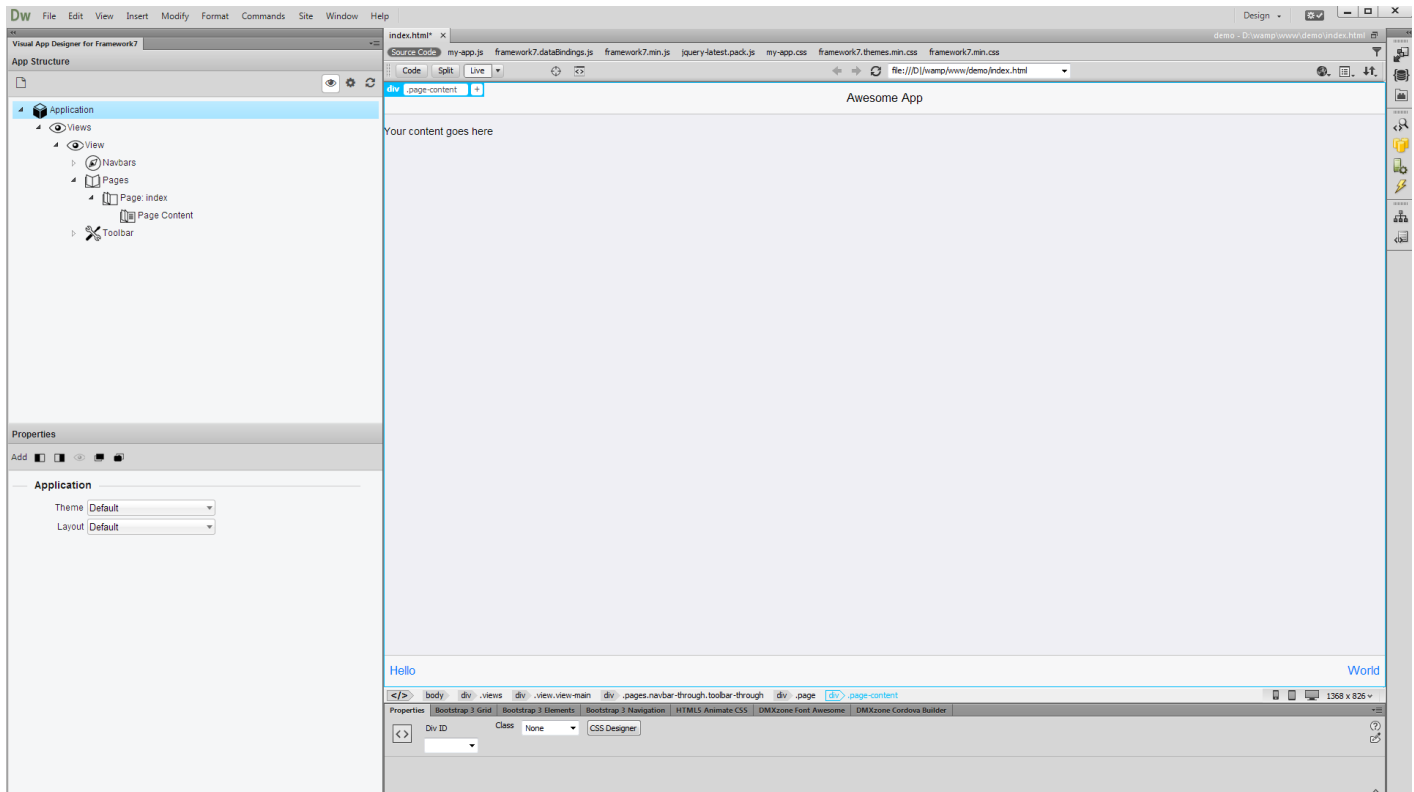
Open the .zxp file if you're using Dreamweaver CC or the .mxd file for Dreamweaver CS+ with Adobe Extension manager and install the extension.

**If you experience any problems contact our Live Support!**

## The Basics: Setting Up Visual App Designer

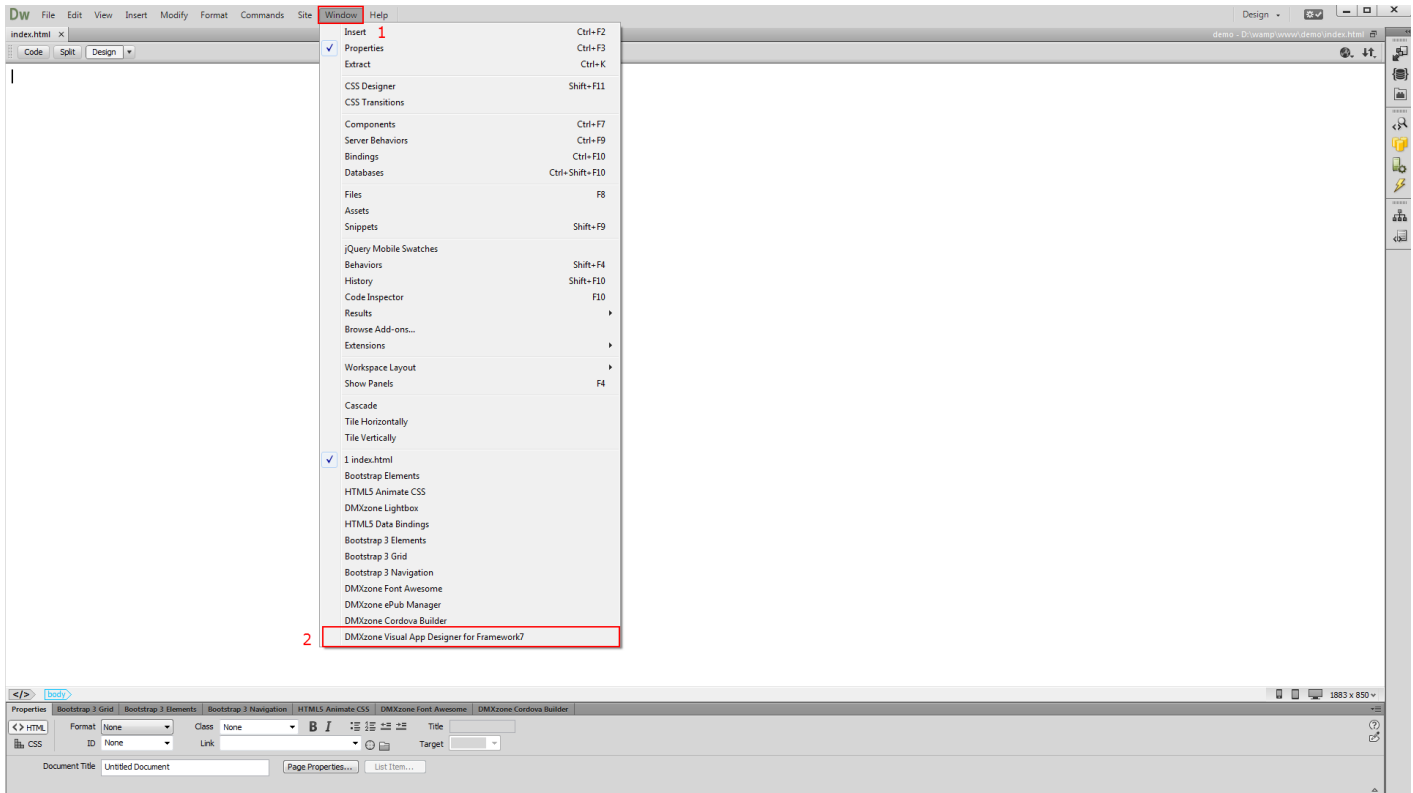
In this tutorial we will show you how to setup [DMXzone Visual App Designer for Framework 7](#) and how to insert the default app layout on your page.

\*You can also check out the [video](#).

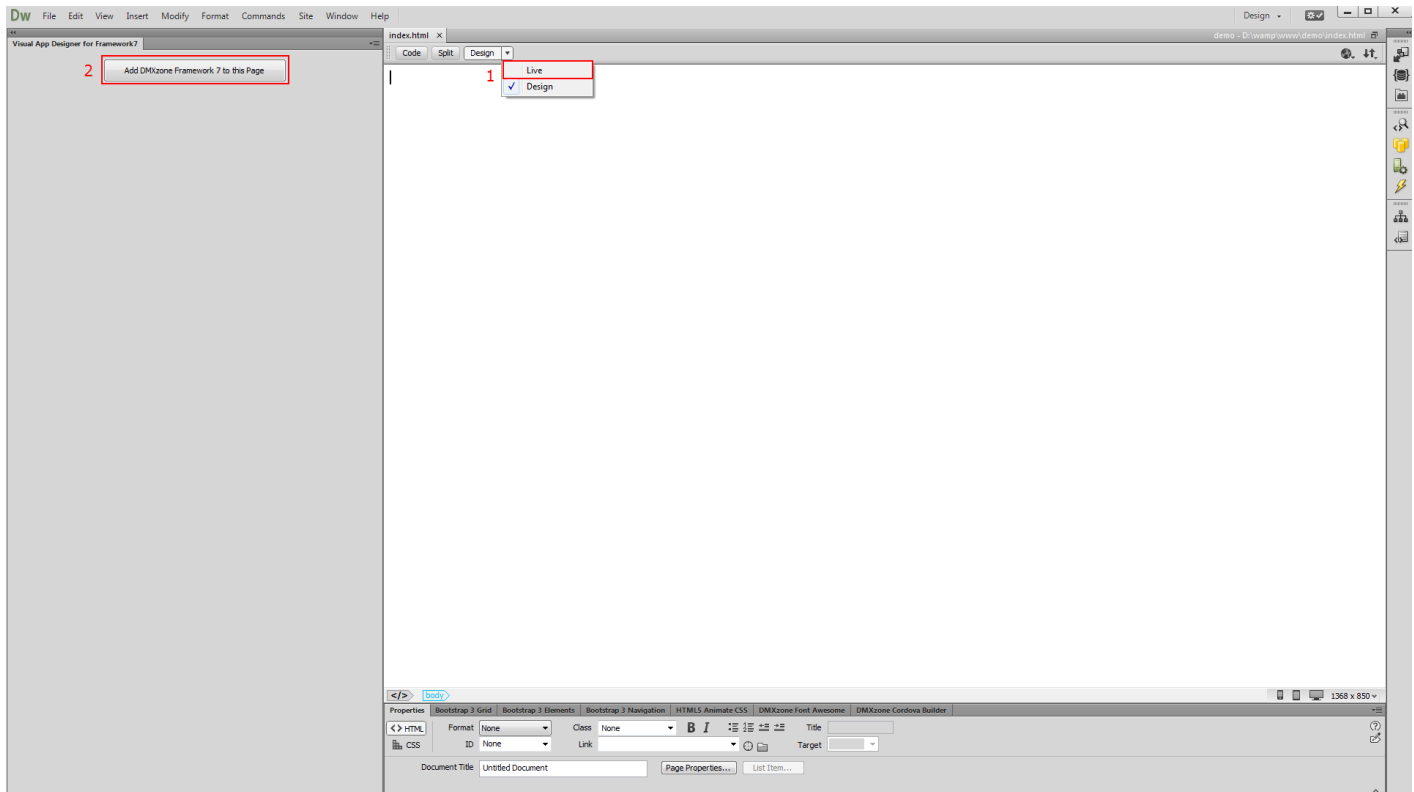


## How to do it

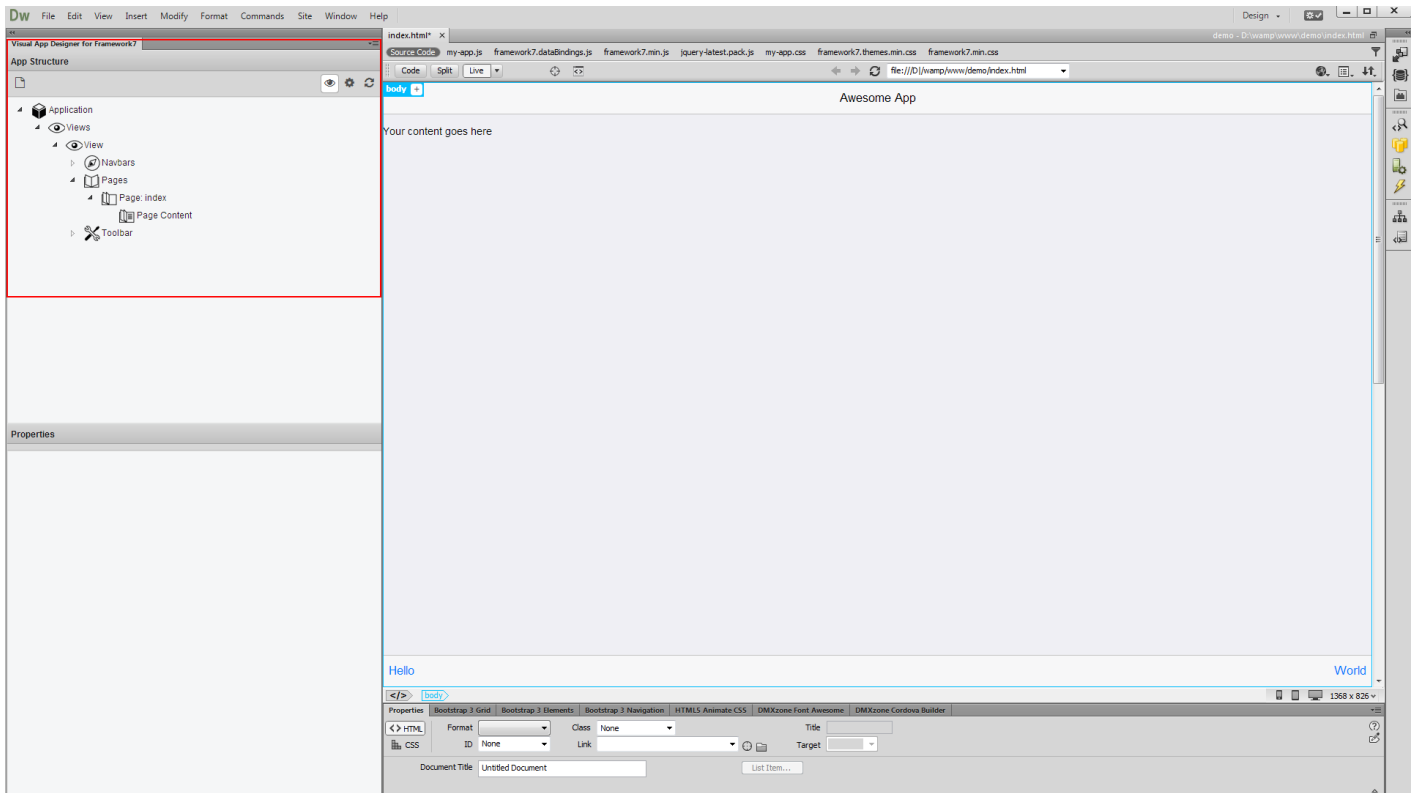
1. From the **window menu (1)** we open the **DMXzone Visual App Designer (2)**.



2. We pin the app designer panel to the left side of our Dreamweaver window. Now, it's more comfortable to build apps. Switch your Dreamweaver to **live view (1)** and click **Add DMXzone Framework7 to this page (2)**.



3. Framework 7 has been successfully added to this page. The visual app designer adds a default application layout. You can inspect its structure in the **Visual App Designer panel**. Also, you can preview your app and make changes directly in your Dreamweaver Live View. That's how easy it is to setup DMXzone Visual App Designer for Framework7 and insert the default app layout on your page.

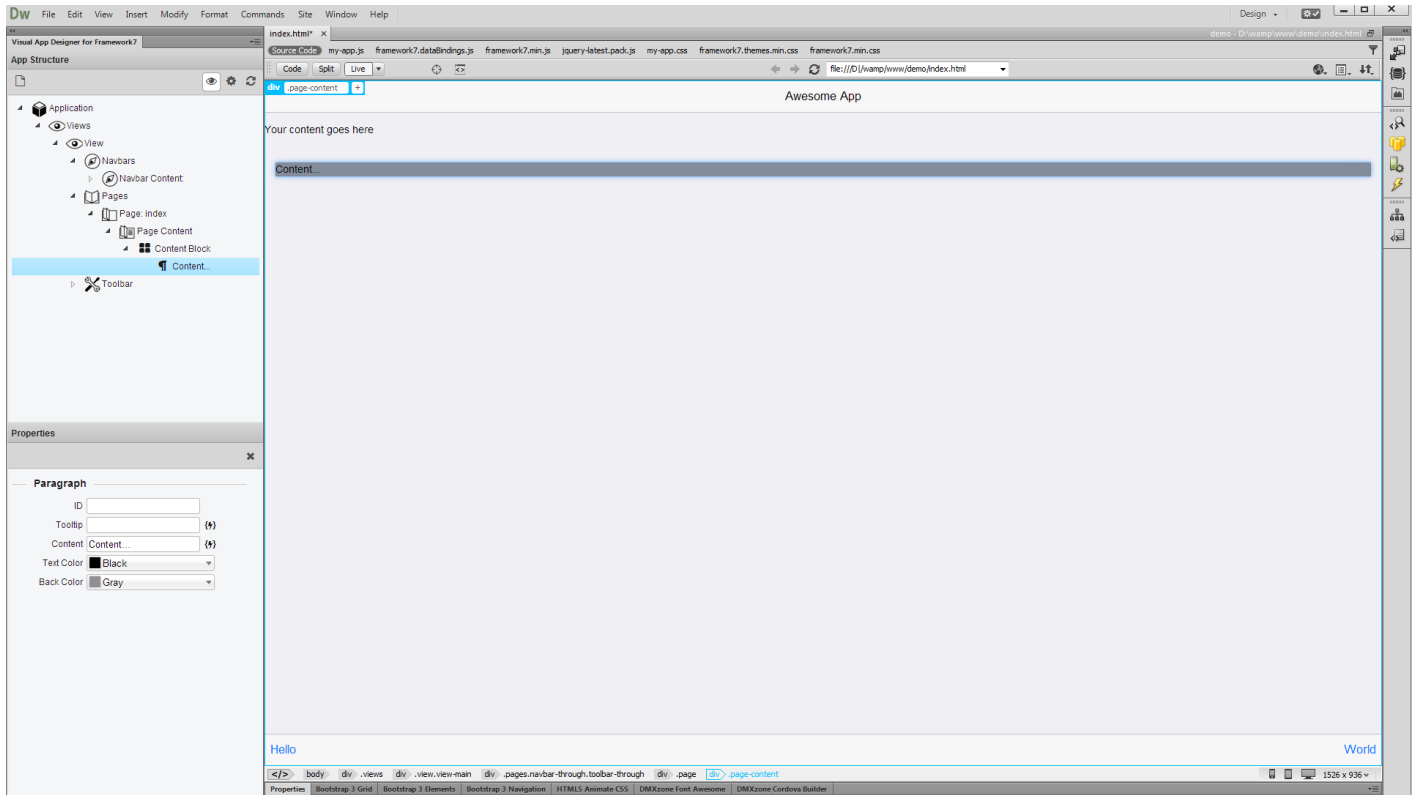




## Advanced: Manage Pages and Content of your App

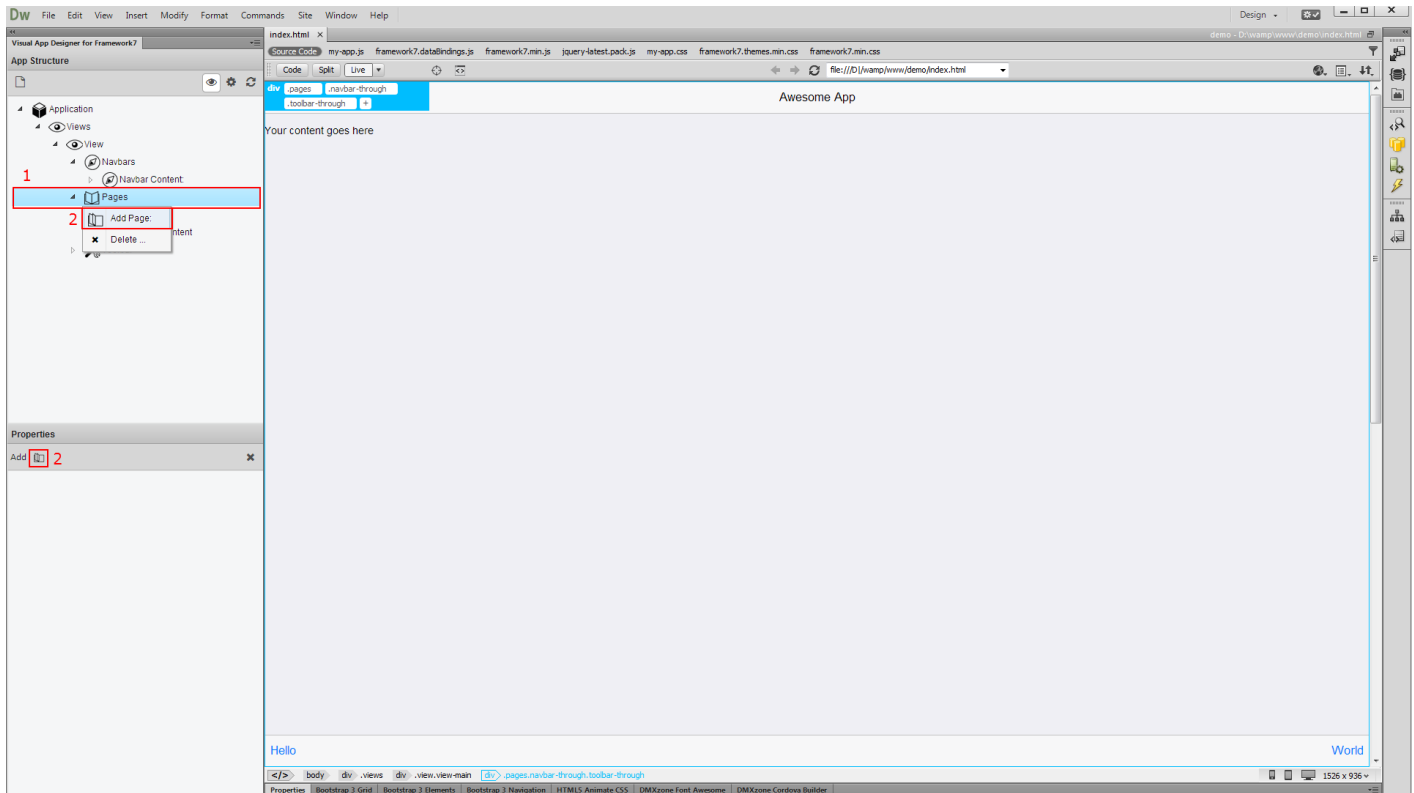
In this tutorial we will show you how to manage the pages of your app, using [DMXzone Visual App Designer for Framework 7](#).

\*You can also check out the [video](#).



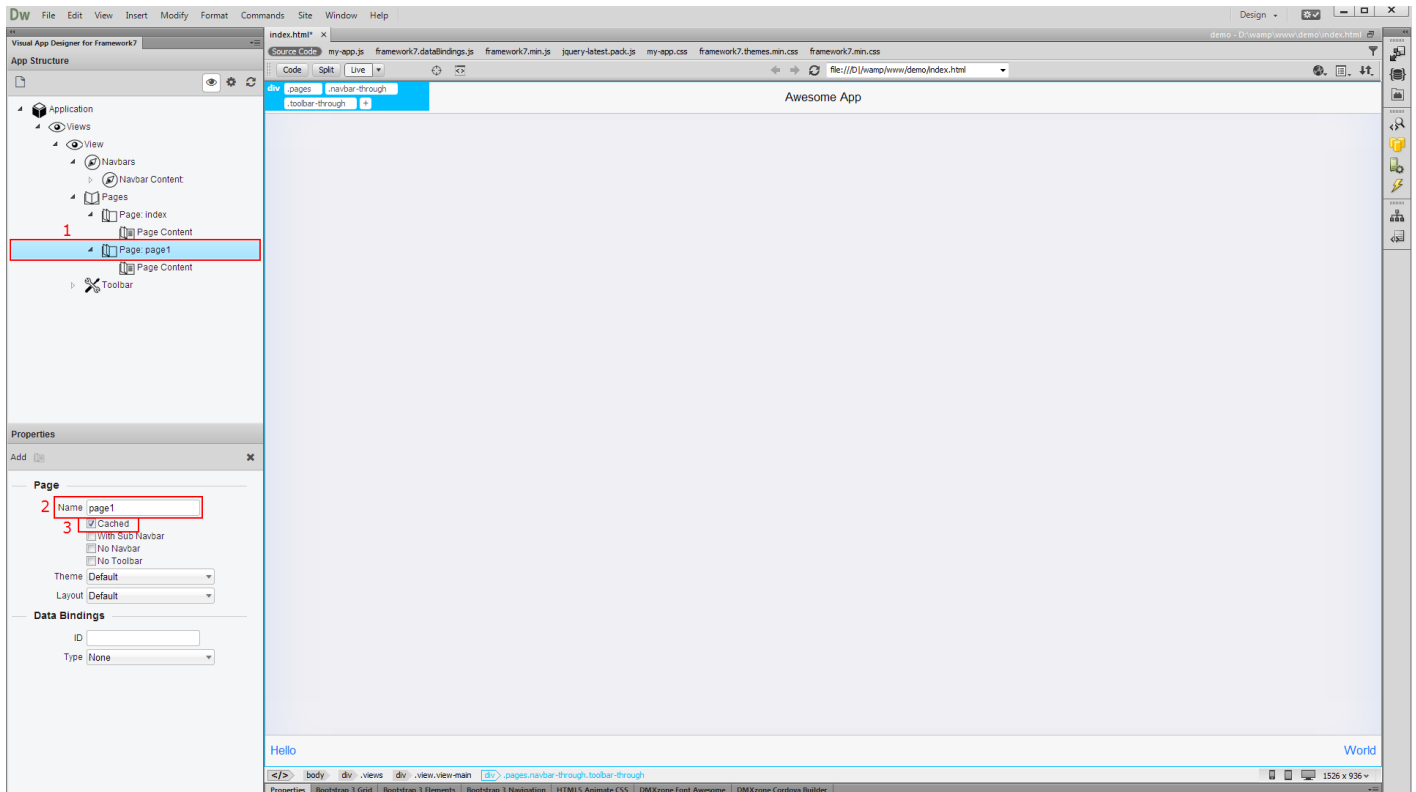
## How to do it

1. In the previous tutorial we added the Framework 7 with the default App layout on our page. The default layout contains a view with a nav bar, toolbar and index page. Let's add another page in our app. Select **pages (1)** in the app structure tree. You can add a new page, by using the **properties toolbar (2)**, or you can **right click and select add page (2)**.

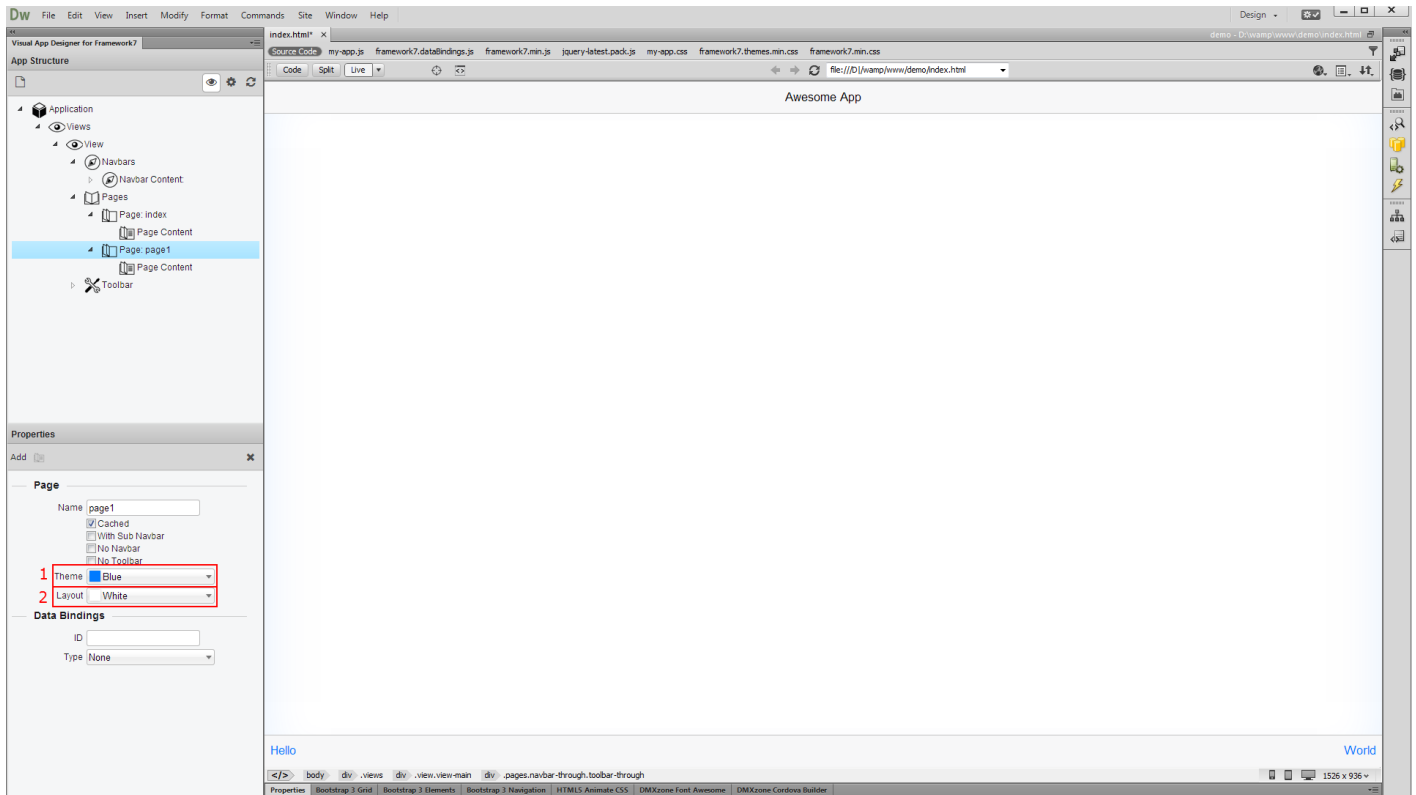


- Our new page has been successfully added. **Select it (1)** in order to get its properties. Here, you can **add a page name (2)**. The page name is used for identification and linking. All of the additional pages that you add are **marked as "cached" (3)**. Choose, whether to show the nav bar, toolbar and sub nav bar on each of the pages, that you add. We leave them to their defaults.

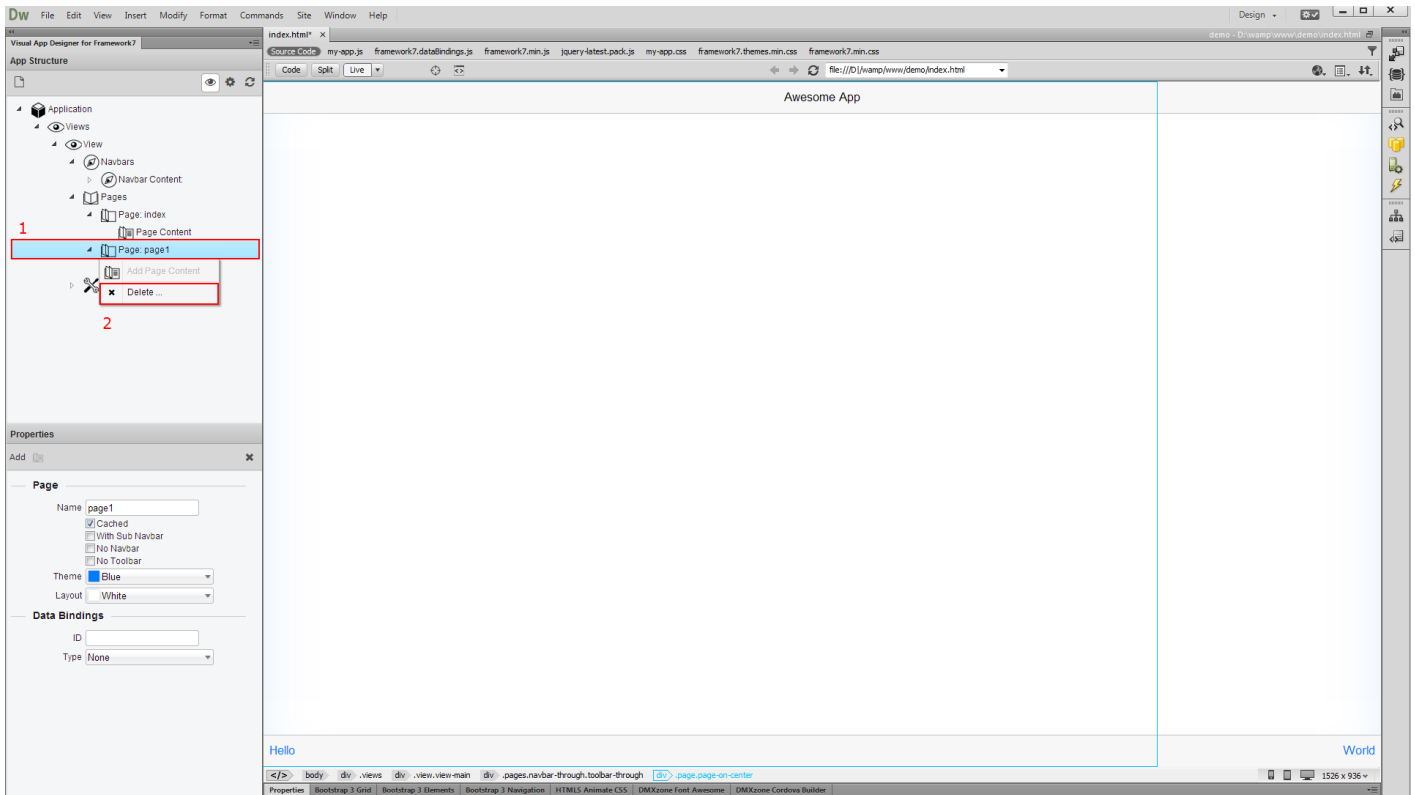
**Note:** Cached pages are loaded in the DOM when the app loads, and they are initially hidden. The index page is not cached by default, as we need it to be visible, when the app loads. You can link to the cached pages, using their names.



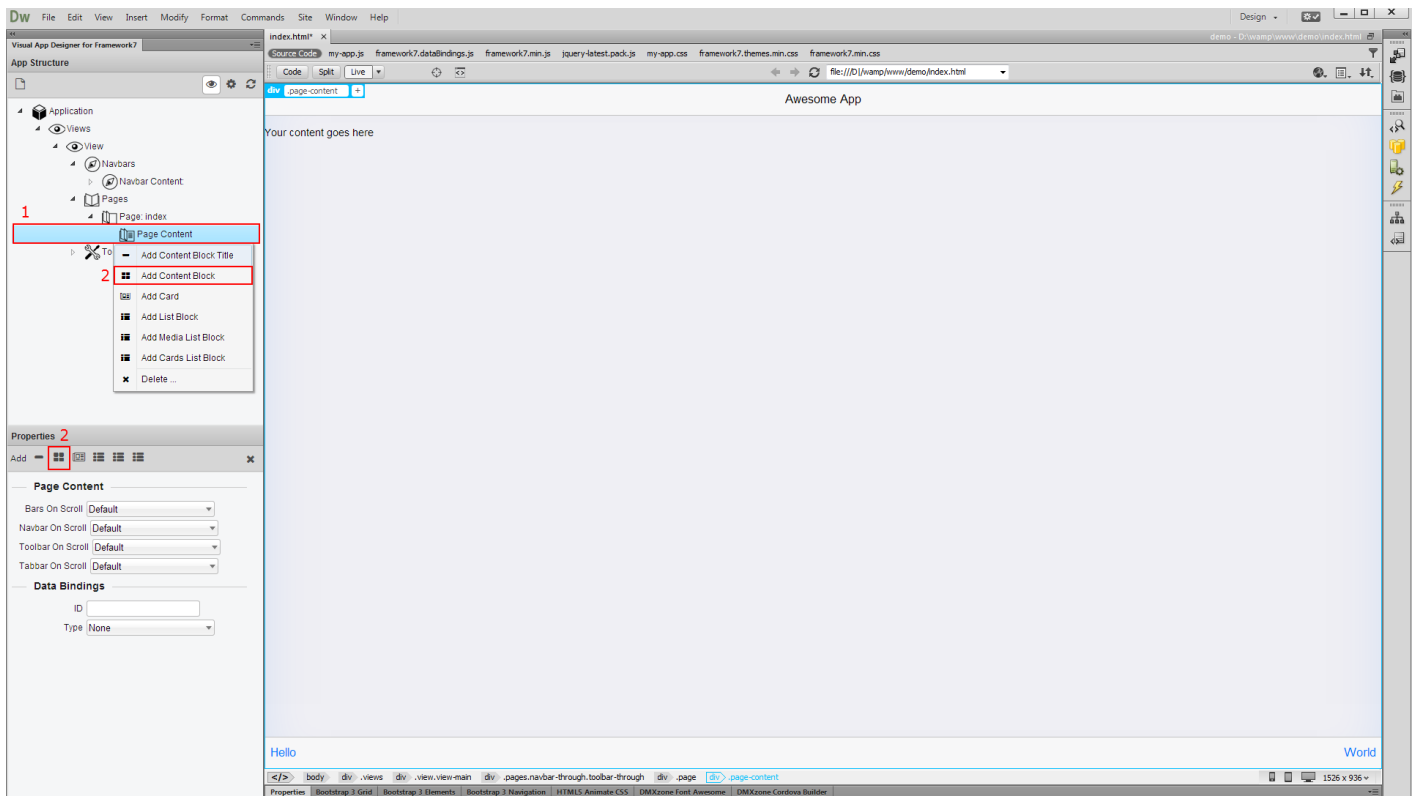
3. Select one of the ready to use default color themes. The applied color theme affects only the interactive elements such as links, buttons, form elements, icons. It doesn't change the basic text color or background colors on other blocks. We select the **blue theme (1)**. Also you can select the page layout color. We choose **white (2)**. You can see the changed that you make, directly in Dreamweaver.



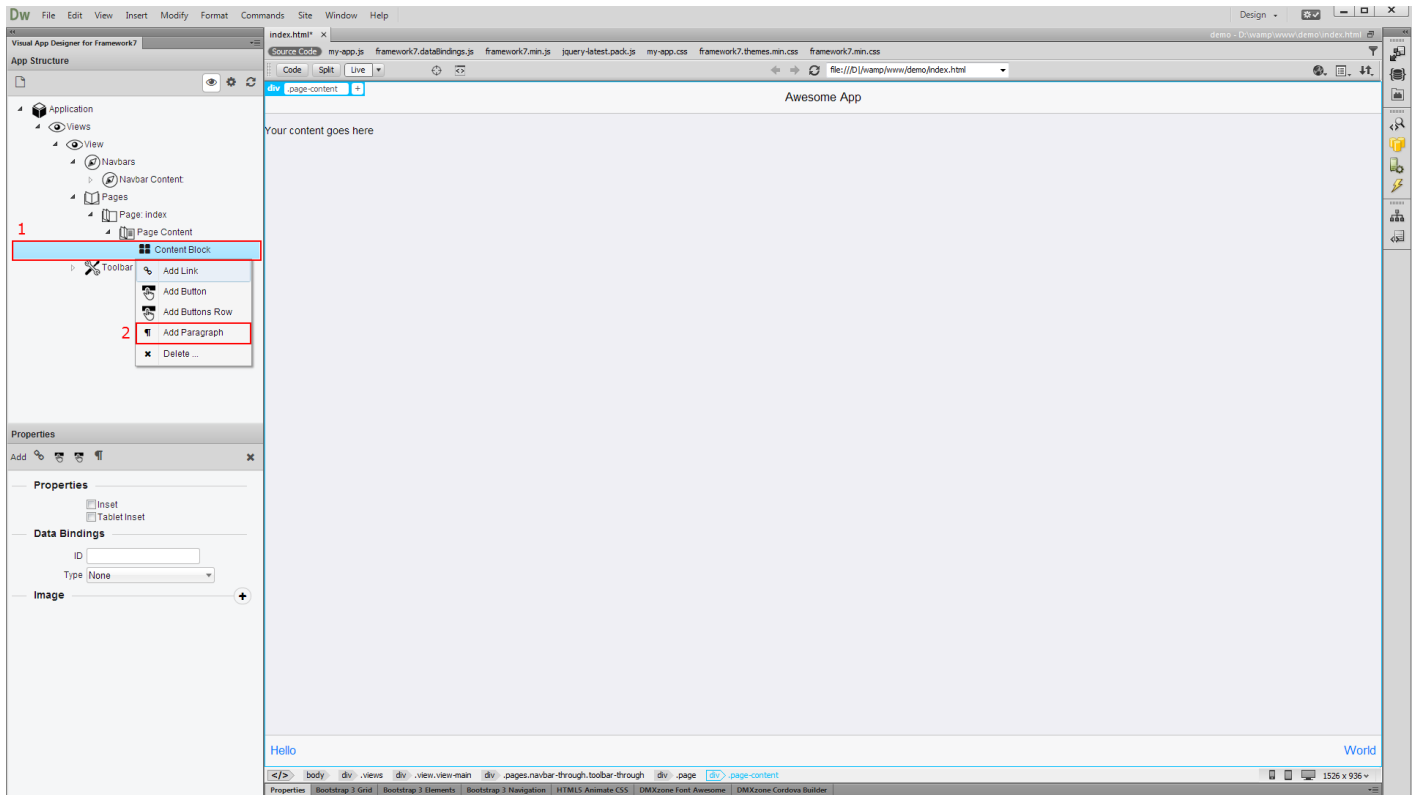
- You can switch between your pages, by clicking on them in the app structure. You can remove any of the pages. Just **right click the one (1)** you want to remove, and select **delete (2)**.



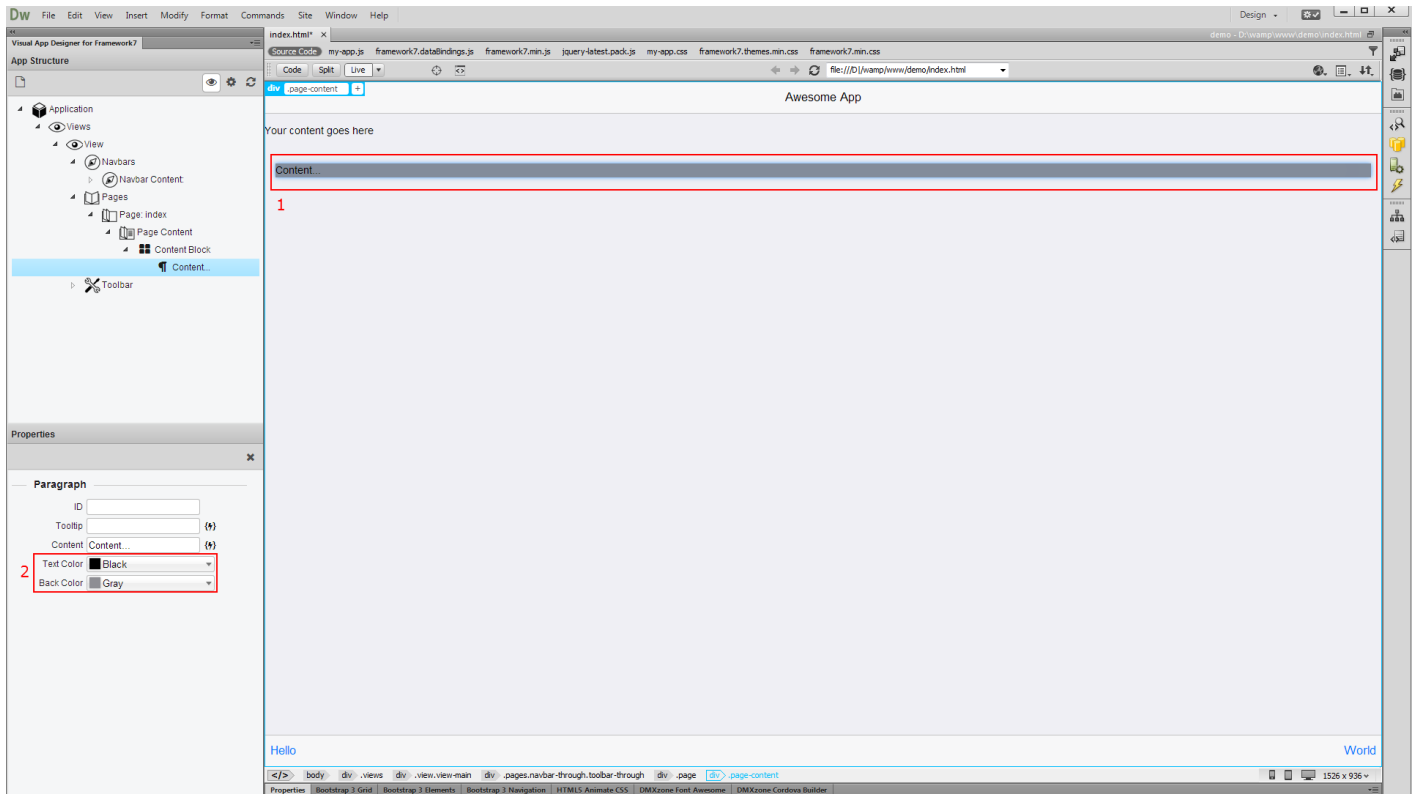
5. In order to add some content to your page, first select its **page content (1)** in the app structure panel. Then select any of the content types in the properties toolbar (2) or **right click the page content** in the app structure and select the content type, in our case it's content block (2).



- Let's add our content within the content block. **Right click on the content block (1)** and select **paragraph (2)**.



7. This adds a paragraph with some default text inside. You can **change its content directly in Dreamweaver live view (1)**. Also, you can change the **text color and background color** of this paragraph **(2)**. We choose black text color and grey back color. That's how easy it is, to manage the pages of your app, using DMXzone Visual App Designer for Framework 7.

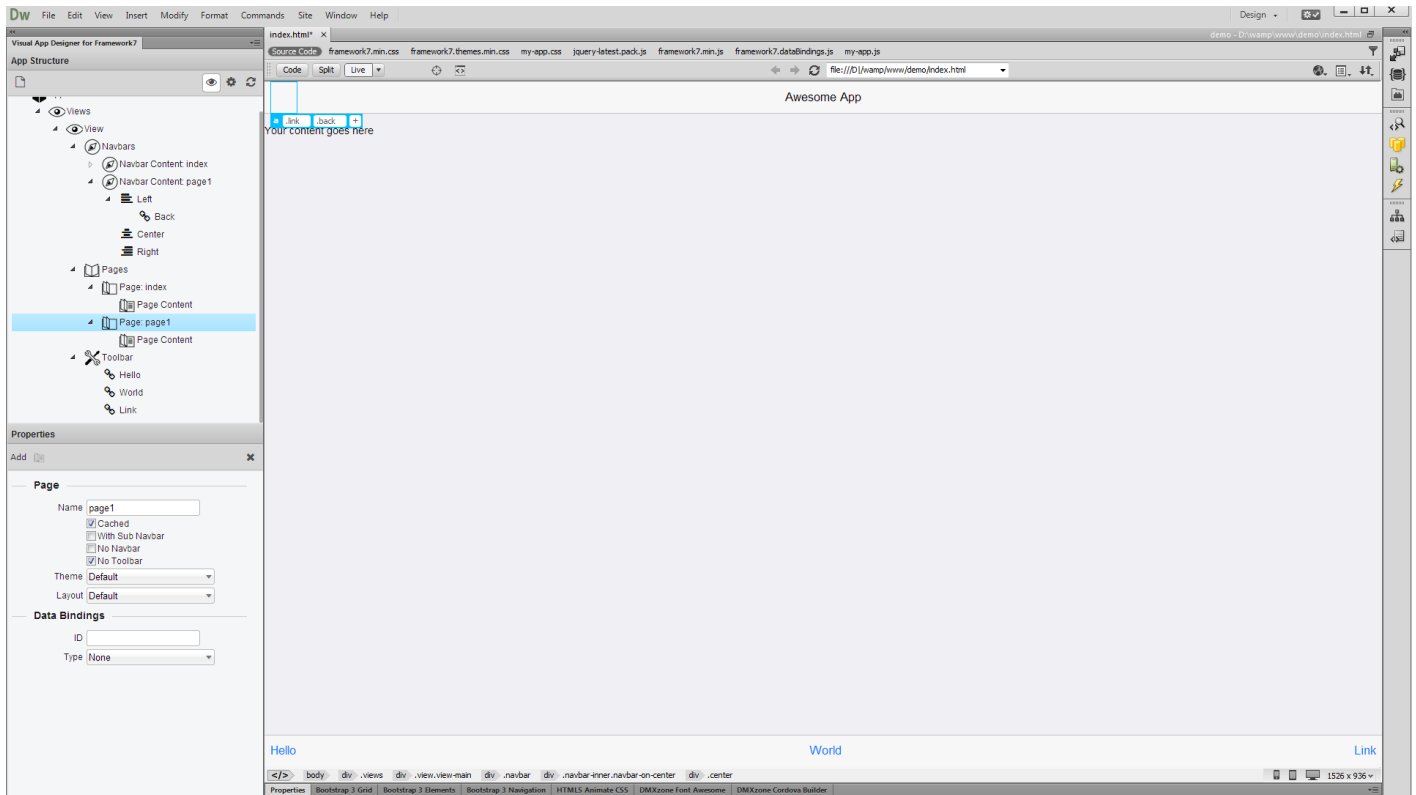




# Advanced: Using Navbars and Toolbar in Visual App Designer

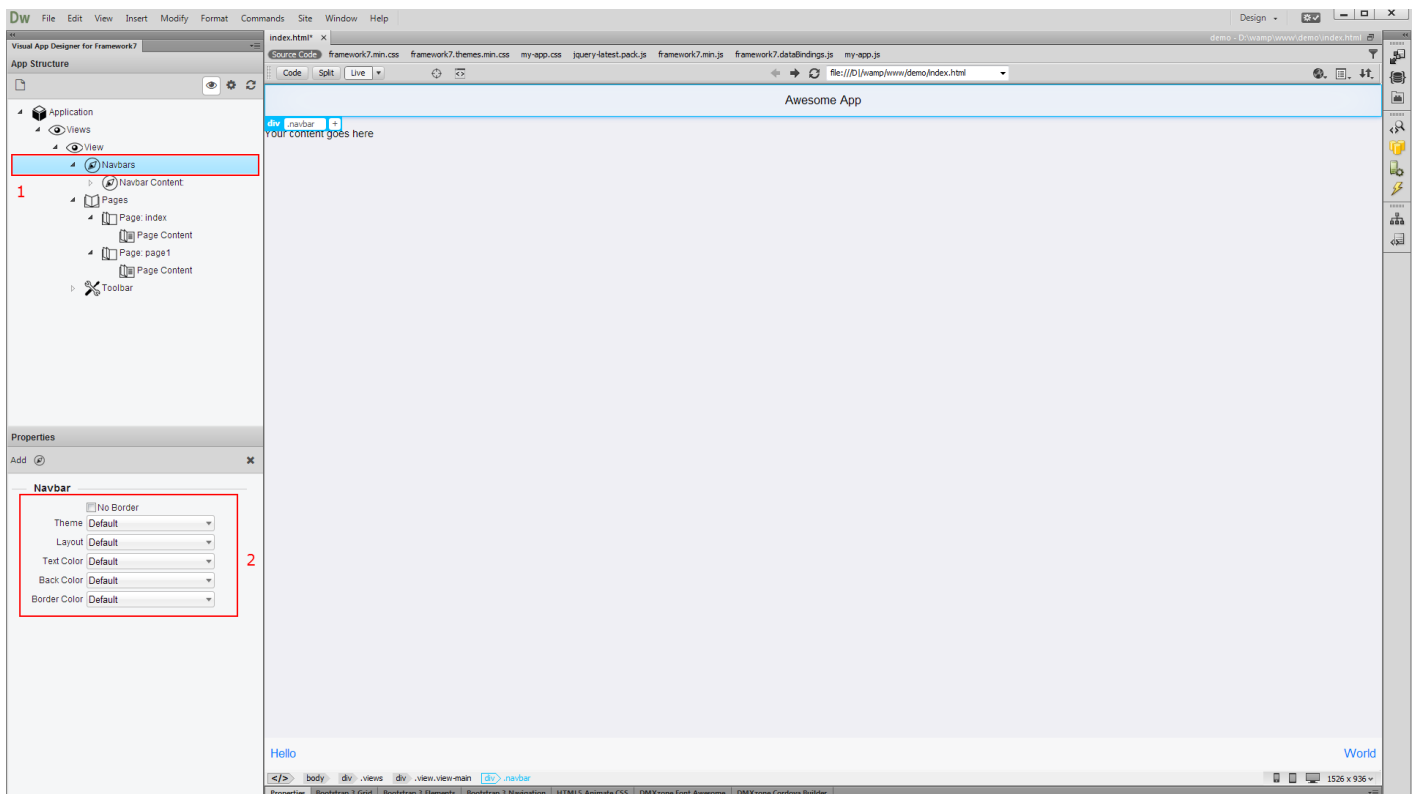
In this tutorial we will show you how to use Framework 7 nav bars and toolbar in your app.

\*You can also check out the [video](#).



## How to do it

1. In our previous tutorials we've inserted Framework 7 default app layout on our page. We also added an additional cached page. The default Framework 7 app layout contains **a nav bar**. Select it in order to see its properties **(1)**. Here, you can customize the **nav bar appearance - and colors**. Also you can **show or hide the bottom border** of the nav bar **(2)**. We leave these settings to their default values.



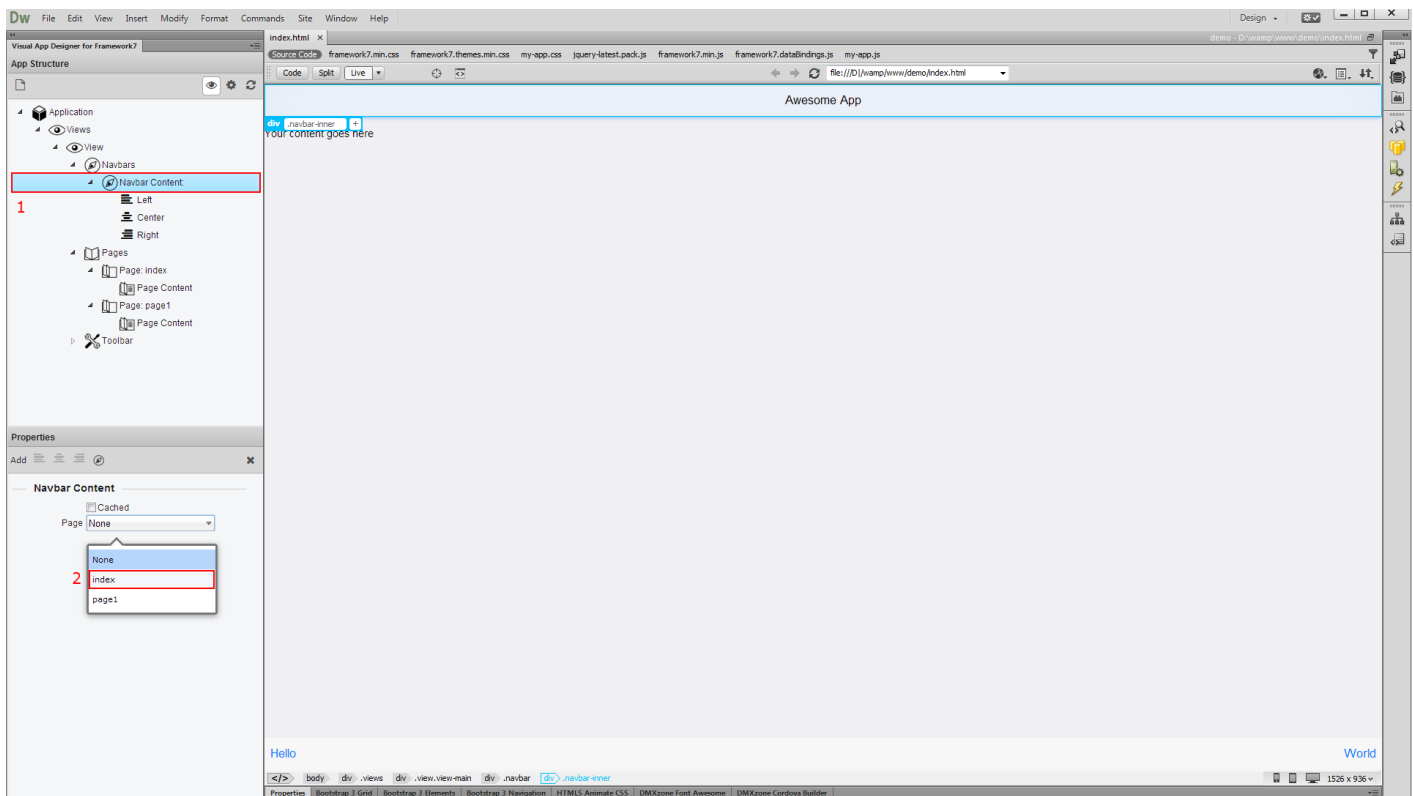
2. Select **navbar content** in the app structure tree **(1)**. The nav bar has 3 main parts: left, center and right. Each of the nav bar contents that you add can be assigned to a specific page. If you do not assign the nav bar content to a page, then it will be displayed across all of the app pages. We want to display this nav bar content only to the index page so we **open the page menu and select index (2)**.

**Note:** Each part may contain any HTML content but it is recommended to use them in the following way:

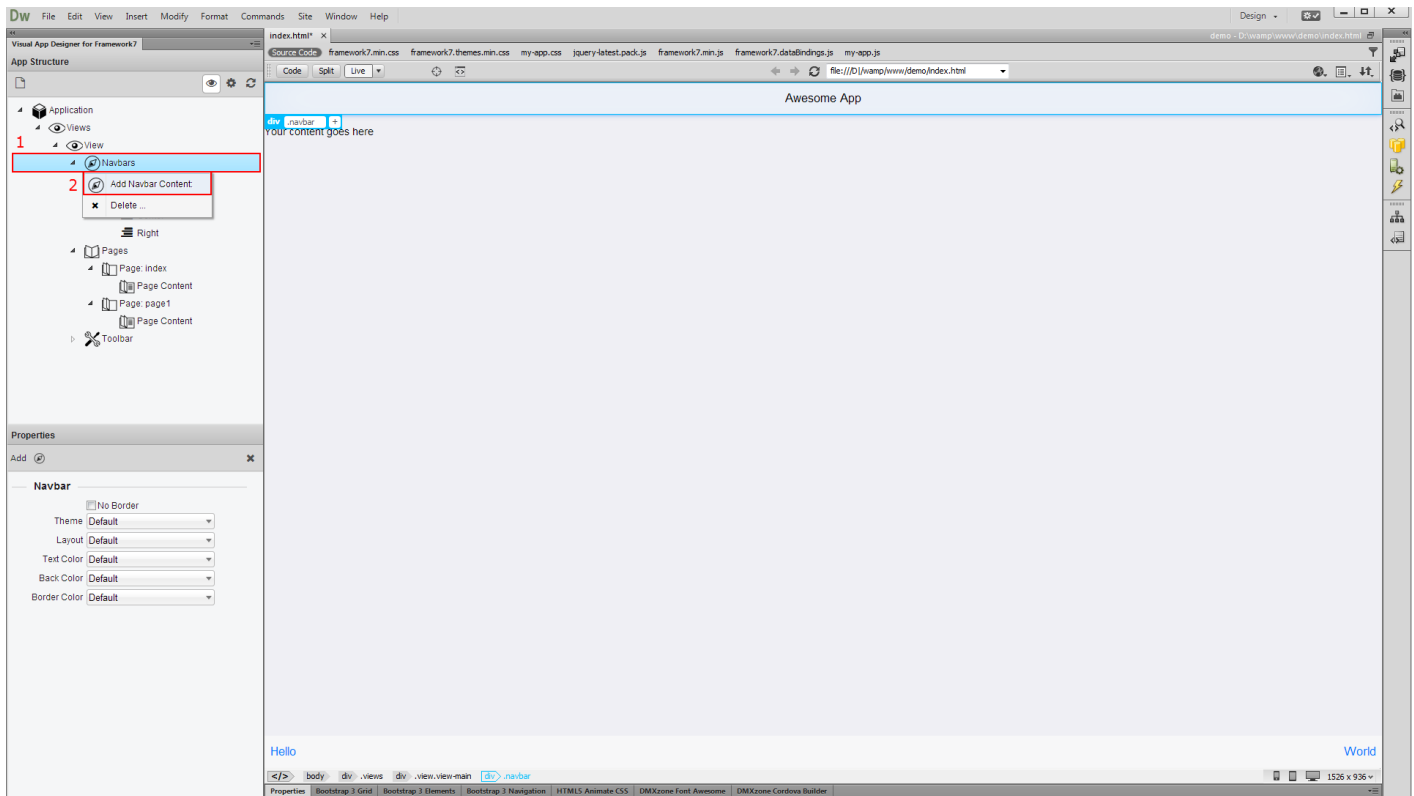
The **left part** is designed to be used with “back link”, icons or with single text link.

The **center part** is used to display page title or tab links.

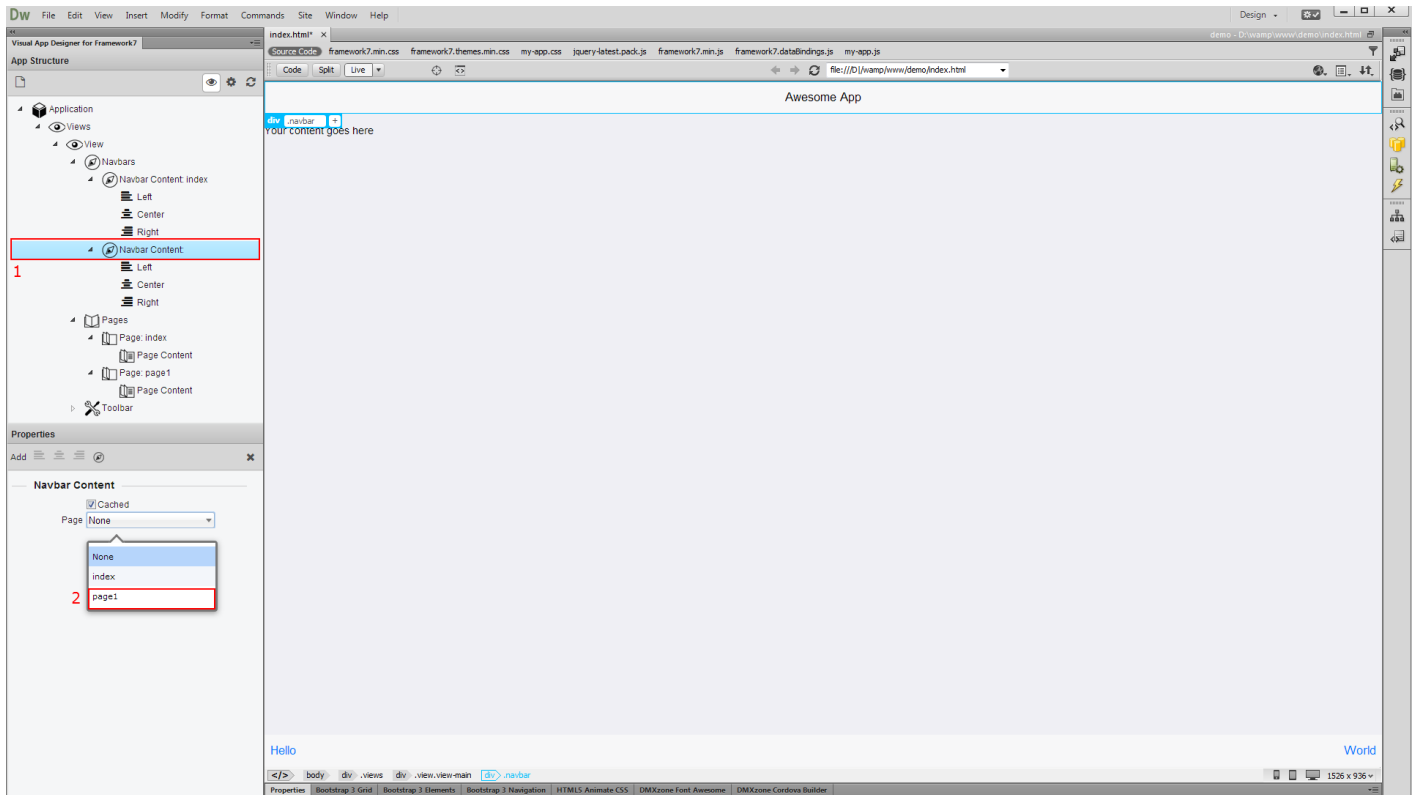
The **right part** is for the same as the left part.



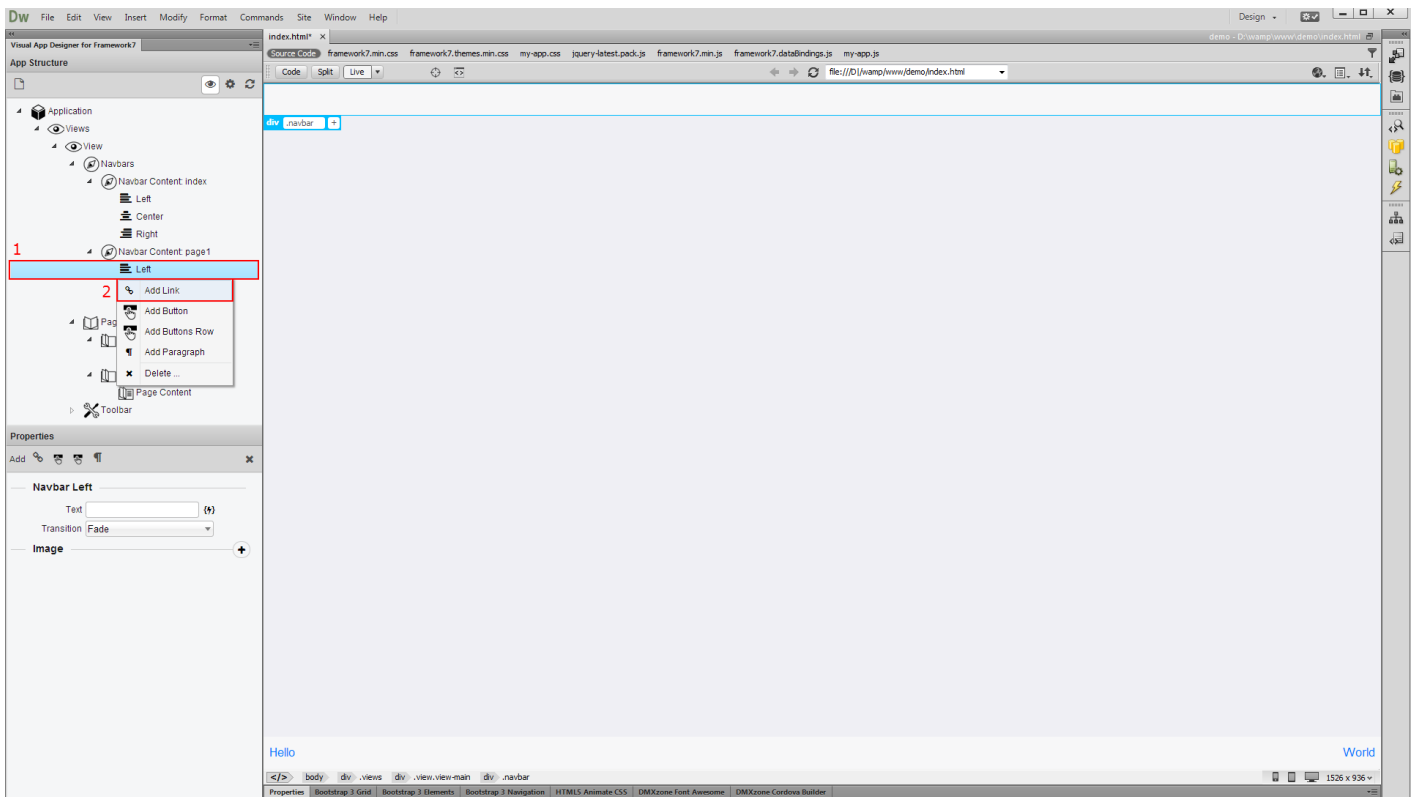
- Now, let's add another navbar content that will be displayed only on our cached page named page 1. **Right-click on the navbars** in the app structure **(1)** and select **add navbar content (2)**.



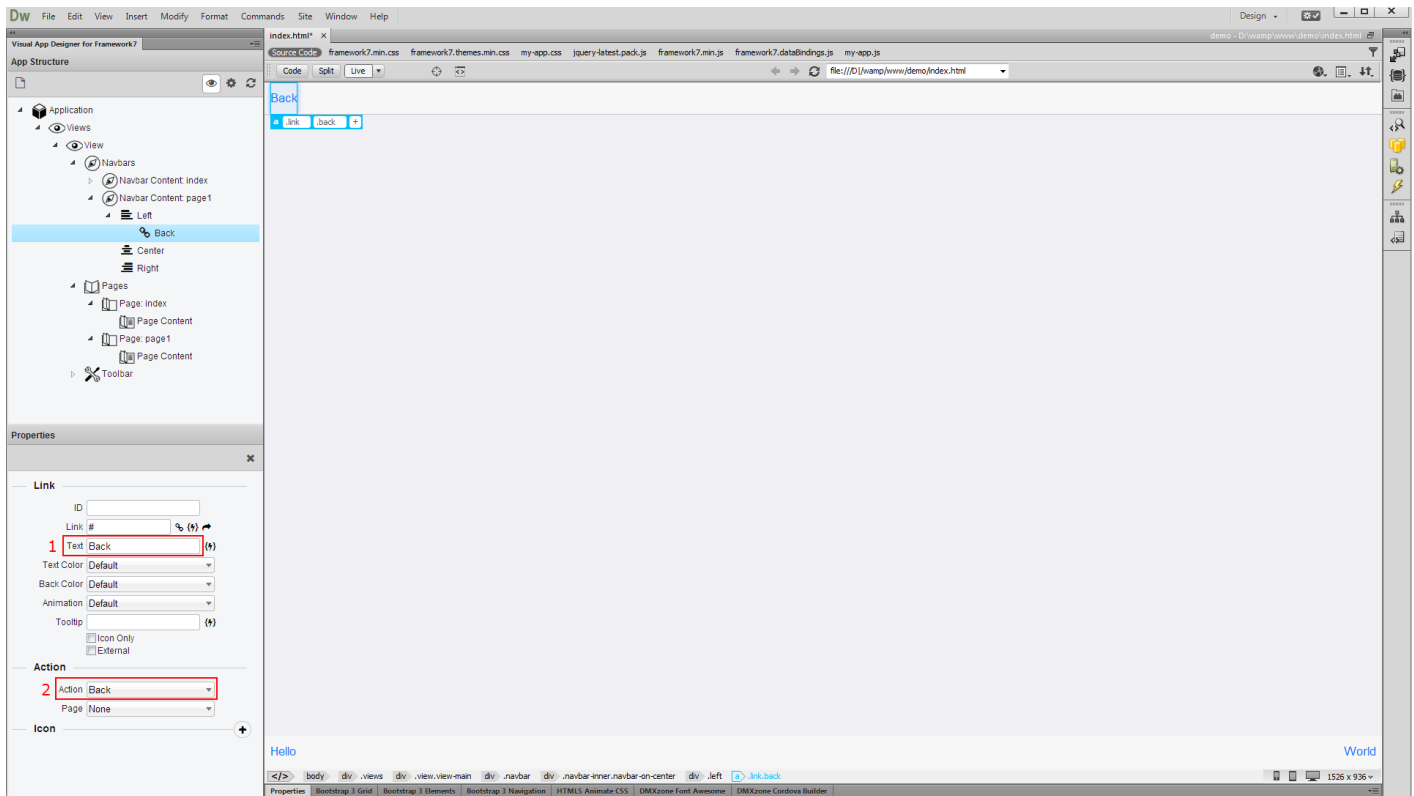
- The new navbar content has been successfully added. Now let's assign it to page 1. Select the **navbar content (1)** and from the page menu **select page 1 (2)**.



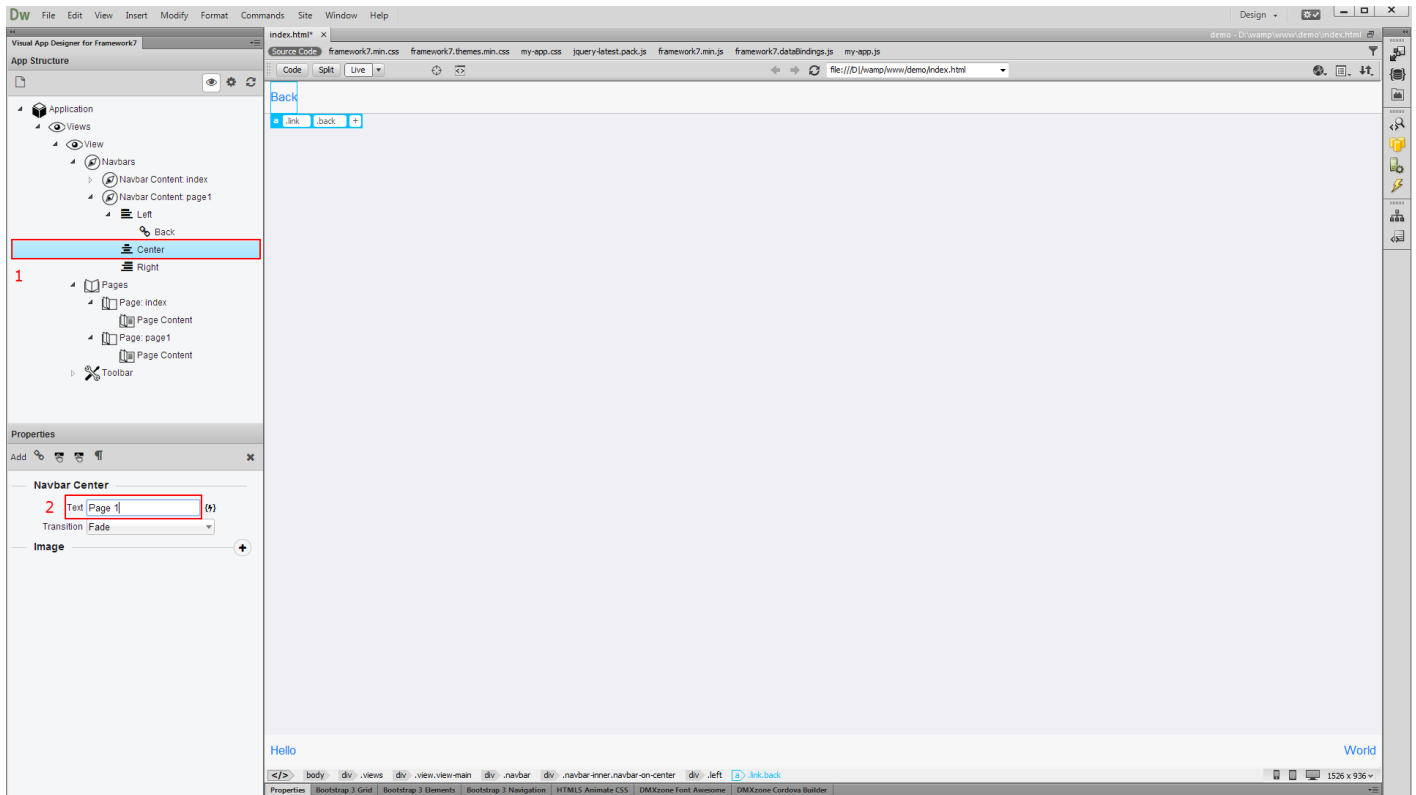
- Now, let's add a back link, to the page 1 navbar. **Right click the left part of the navbar content (1)** and select **add link (2)**.



6. We change the default link text to: **back (1)**. From the action menu **select back (2)**.

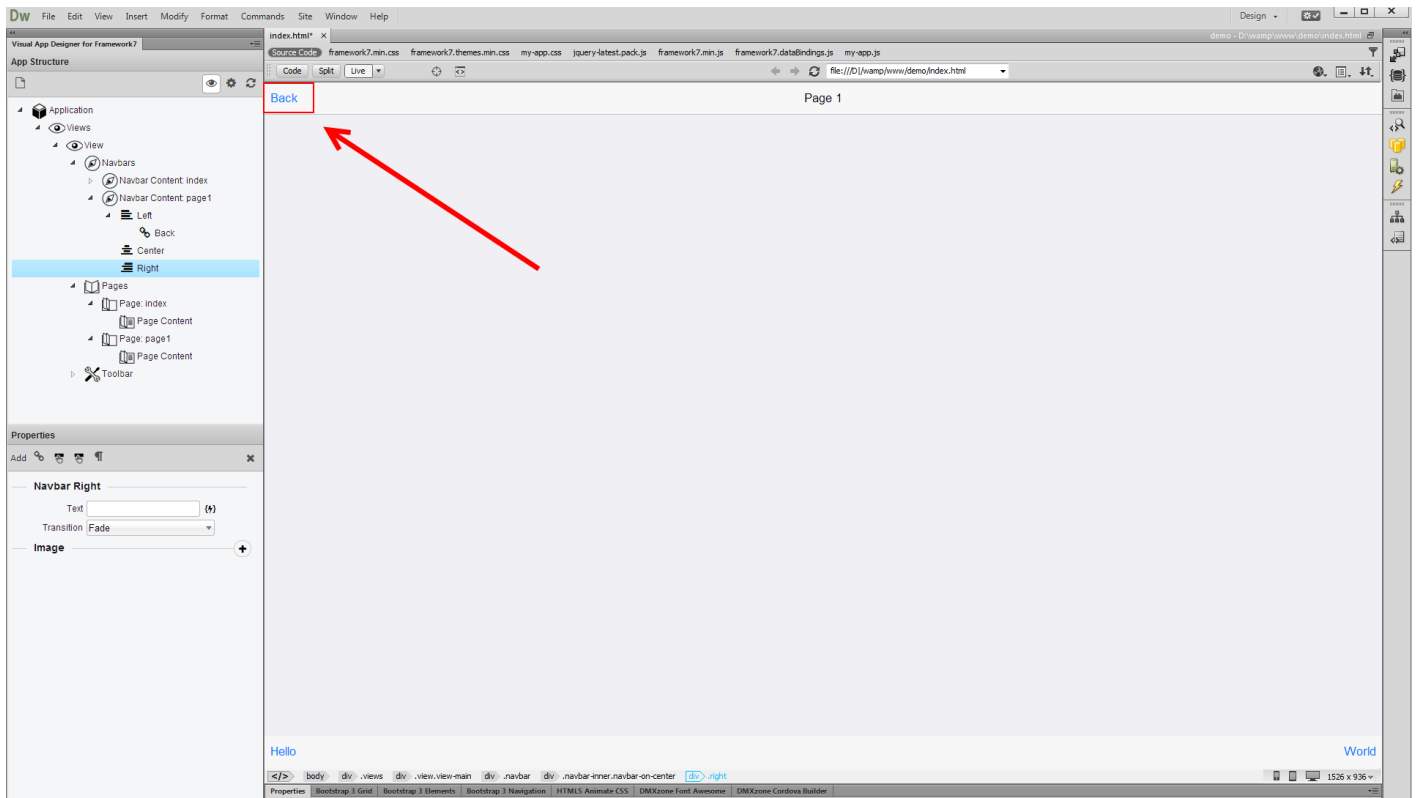


- Now, let's add the page title into the center part of the navbar content. Select **center (1)** and enter the **page title** into the text field **(2)**. We leave the right section empty, as we do not need it right now.



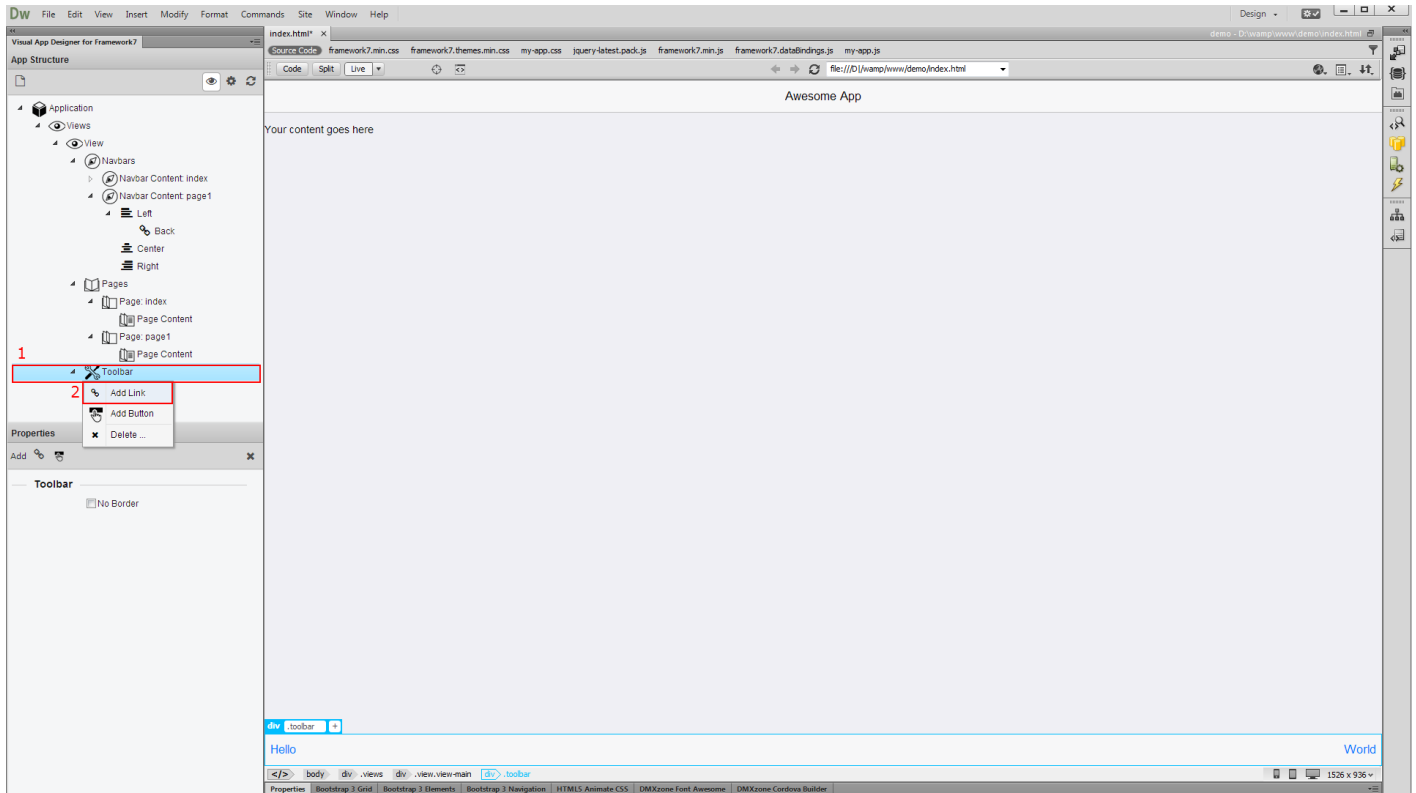


8. You can test the back button, by clicking it in Dreamweaver Live View.

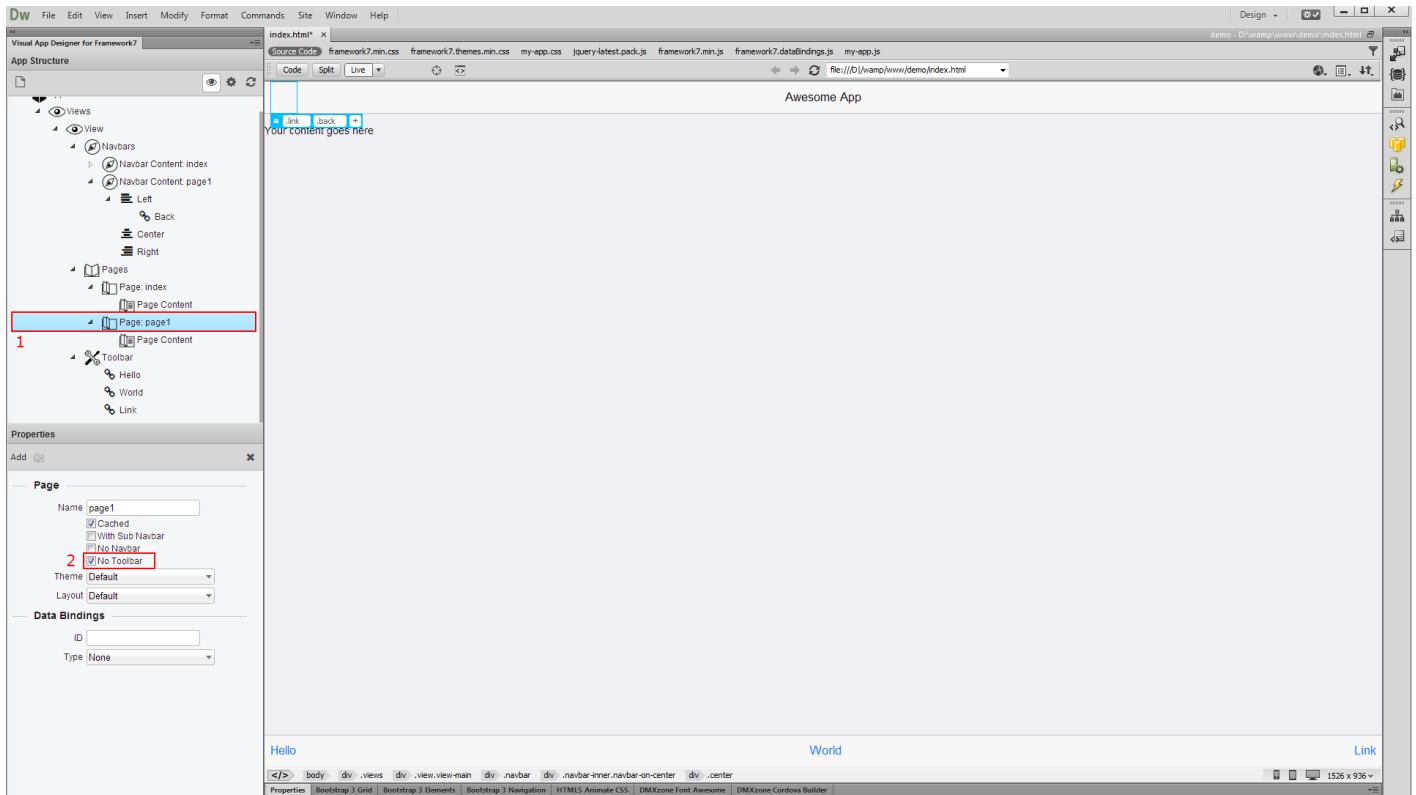


- The default Framework 7 app layout contains a toolbar. **Right click the toolbar (1)** and select **add link (2)**.

**Note:** The toolbar is a fixed area at the bottom of a screen that contains navigation elements. It does not have any parts, just plain links inside. By default, all of the Toolbar elements are equally spaced along toolbar - they have equal space between each other.



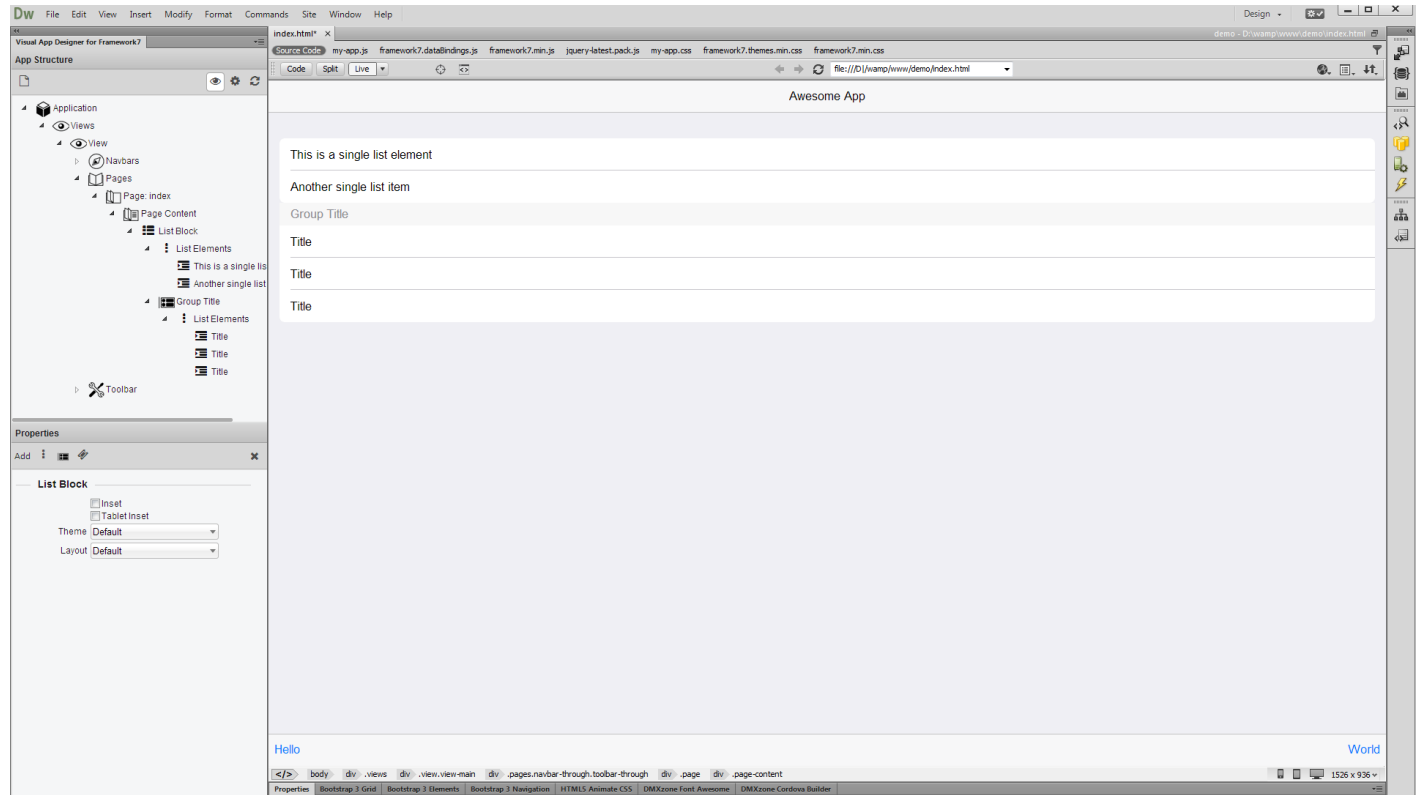
10. You can see that our link has been successfully added. You can apply some action to it or you can link it to a specific page. We do not want our toolbar to be displayed on page 1. So, we **select page 1 (1)** and **check no toolbar checkbox (1)**, located in the properties panel. This option hides the toolbar on the selected page. That's how easy it is to manage the navbars, navbar content and toolbar, using DMXzone Visual App Designer.



# Advanced: Insert List Blocks and List Items in Visual App Designer

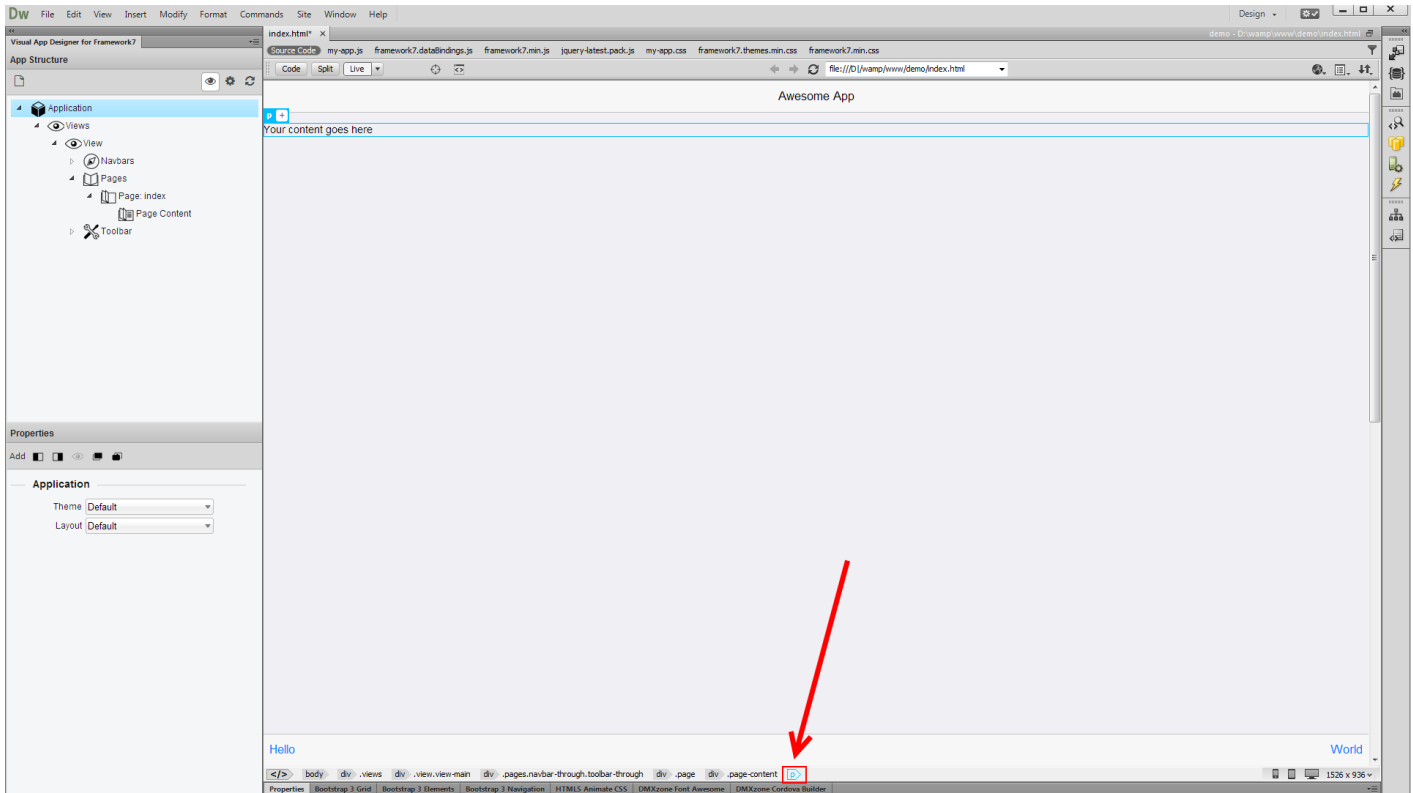
In this tutorial we will show you how to insert list blocks and list items into your app layout, using [DMXzone Visual App Designer for Framework 7](#).

\*You can also check out the [video](#).

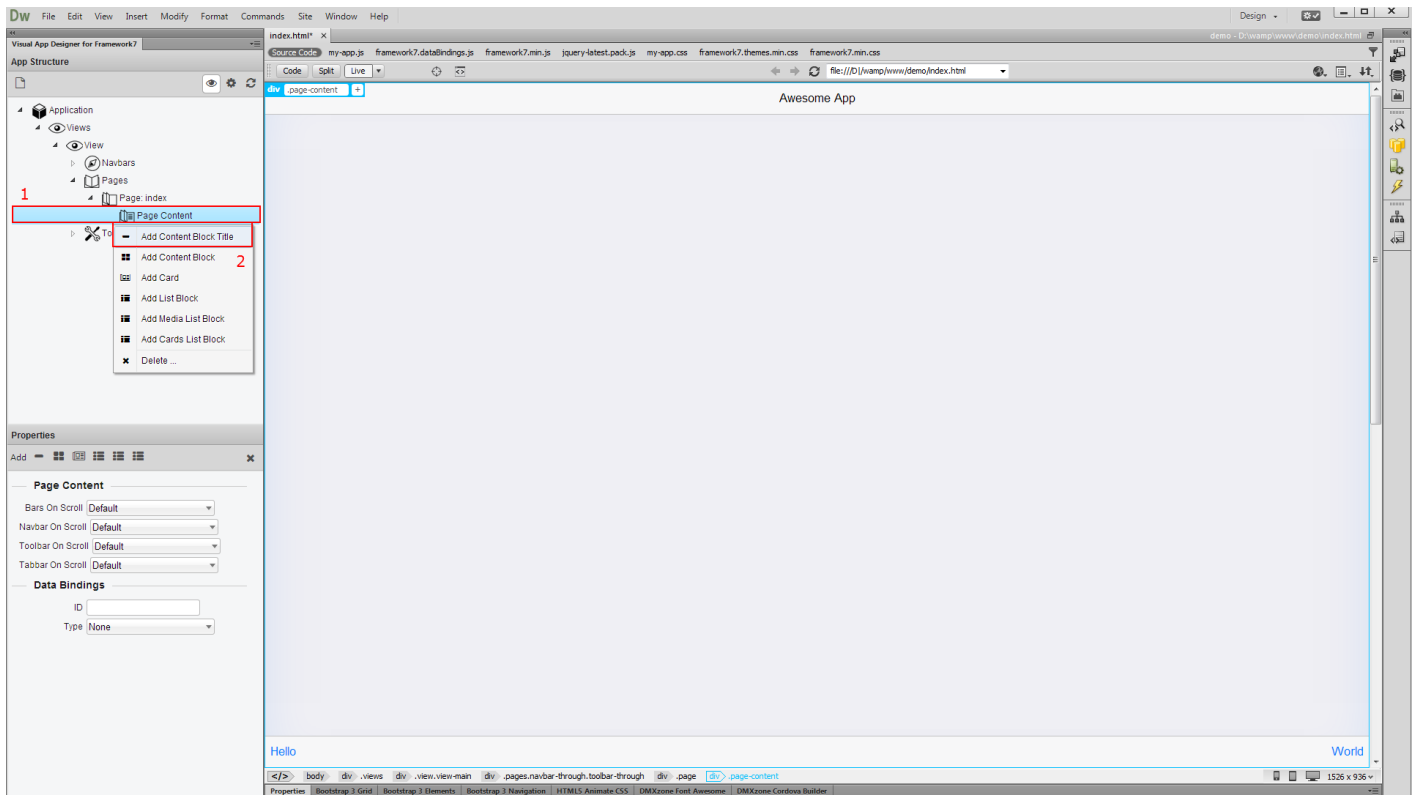


## How to do it

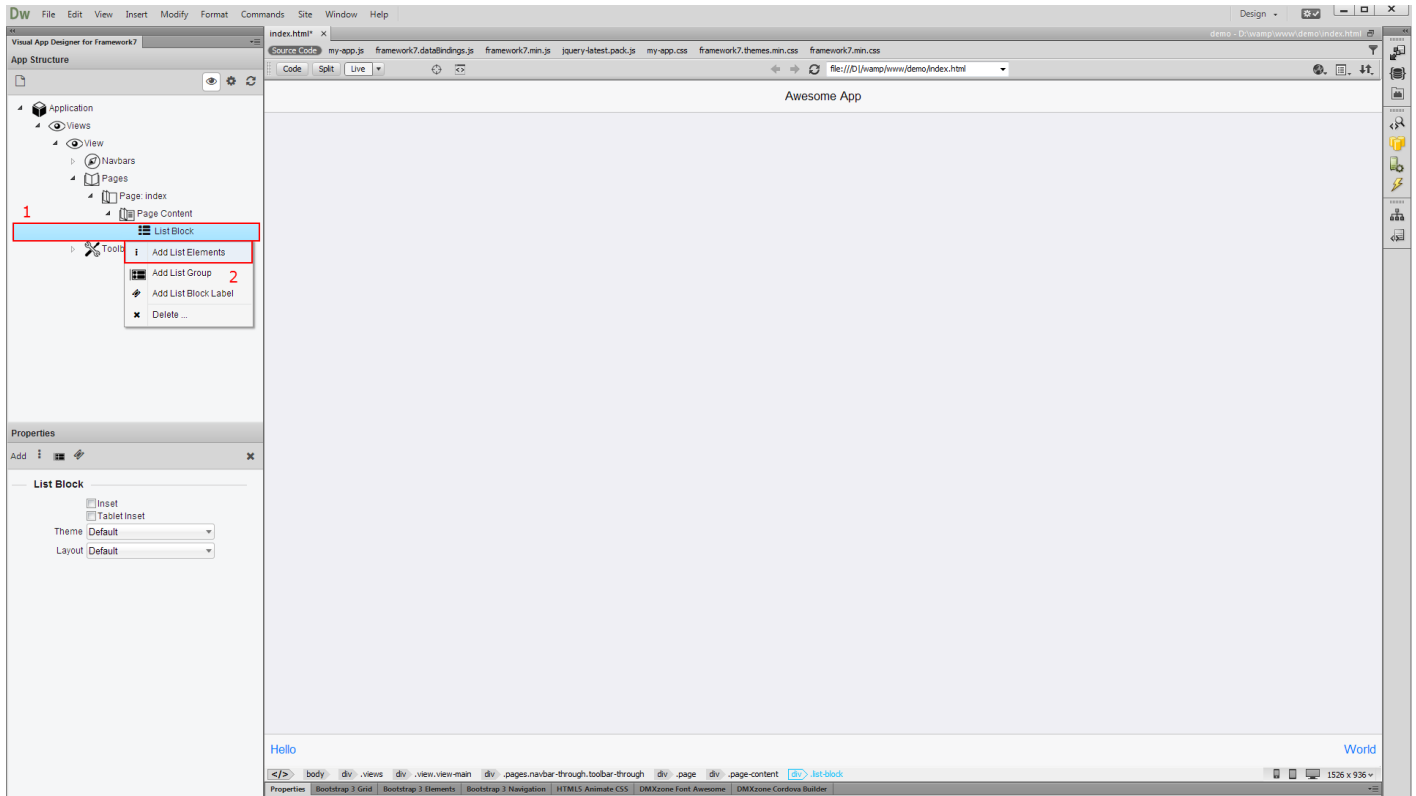
1. First, let's remove the default content from our page. Select the paragraph and delete it.



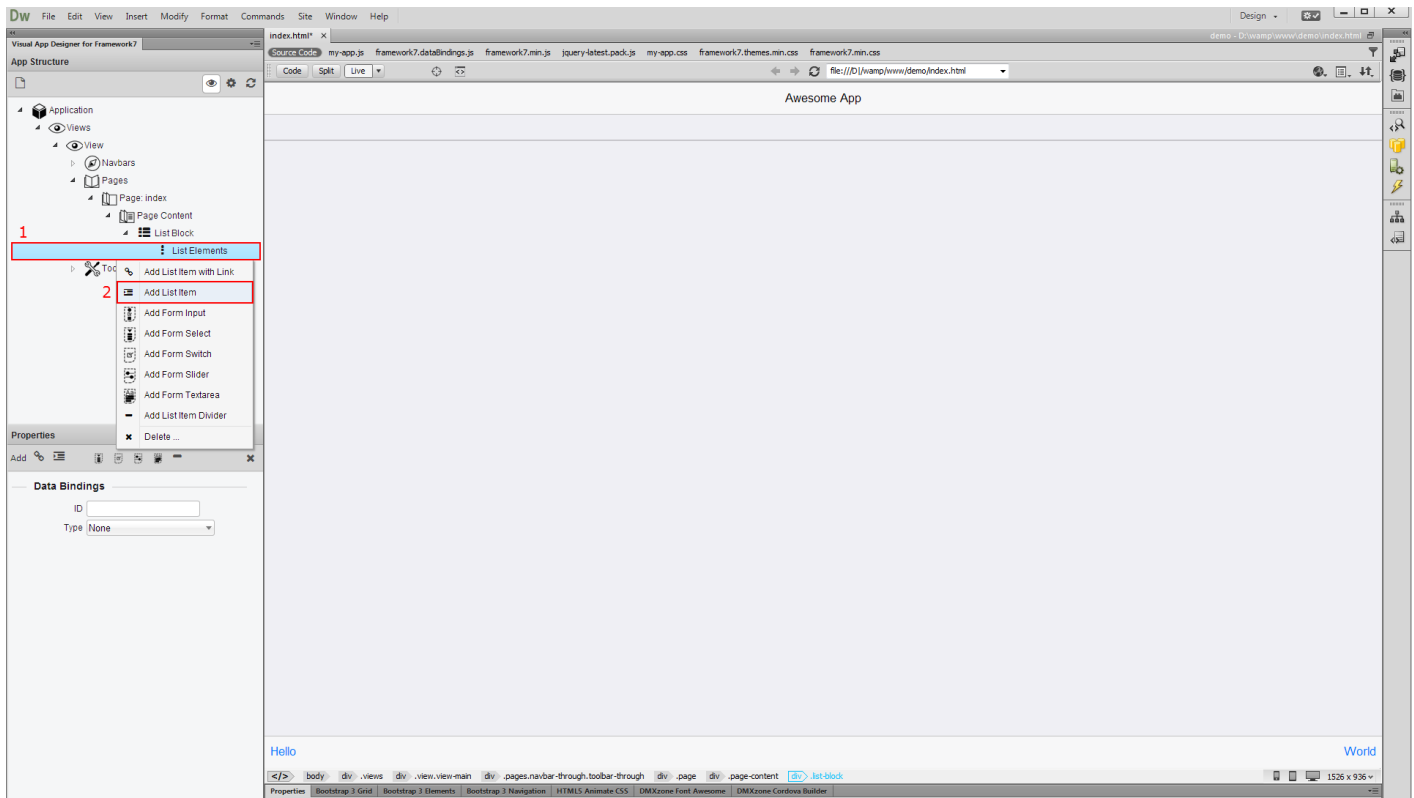
2. Right click the **page content** in the app structure (1) and select **add list block** (2). List block is your list elements wrapper.



3. Right click the list block that we've just inserted (1) and add a single list of elements (2).

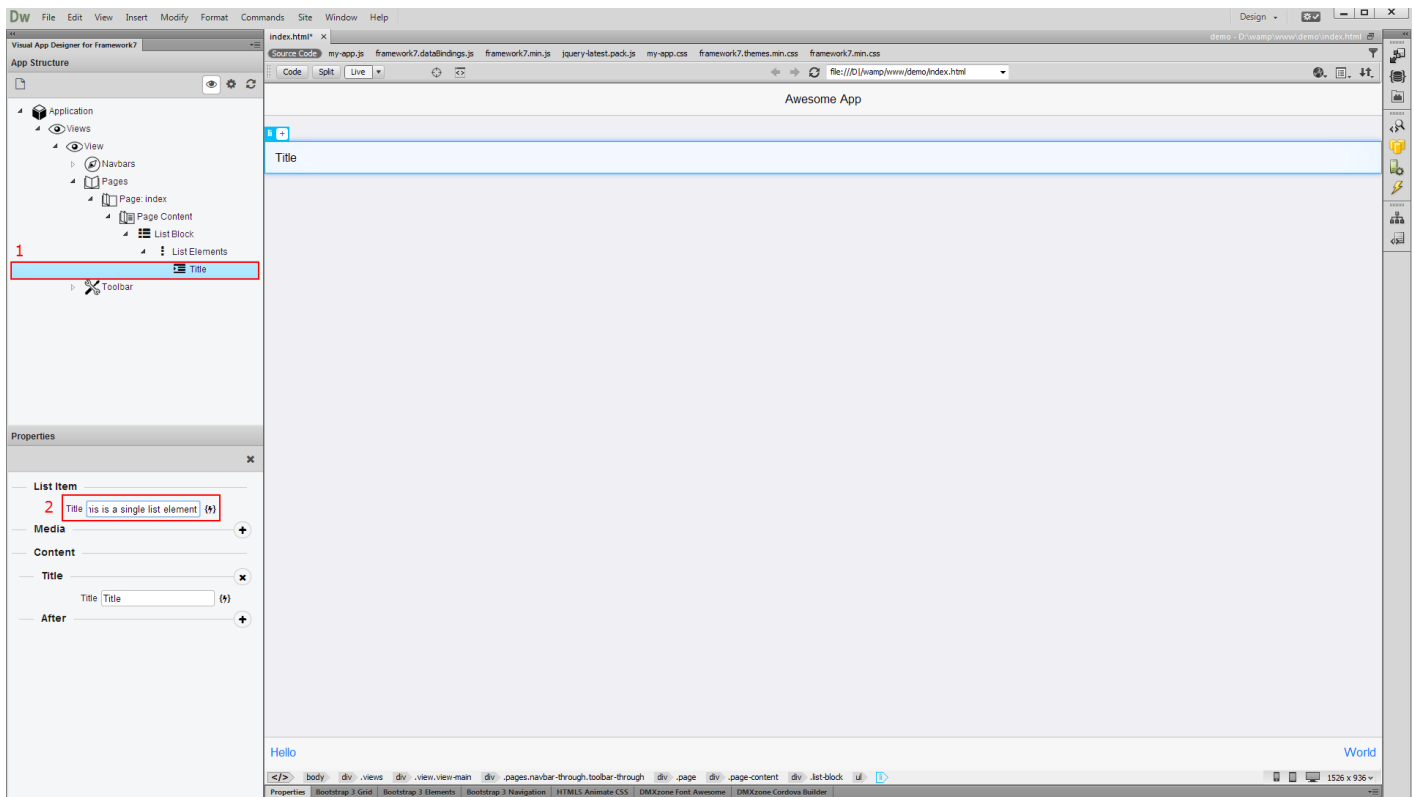


#### 4. Right click the list elements in the app structure (1) and select **add list item** (2).

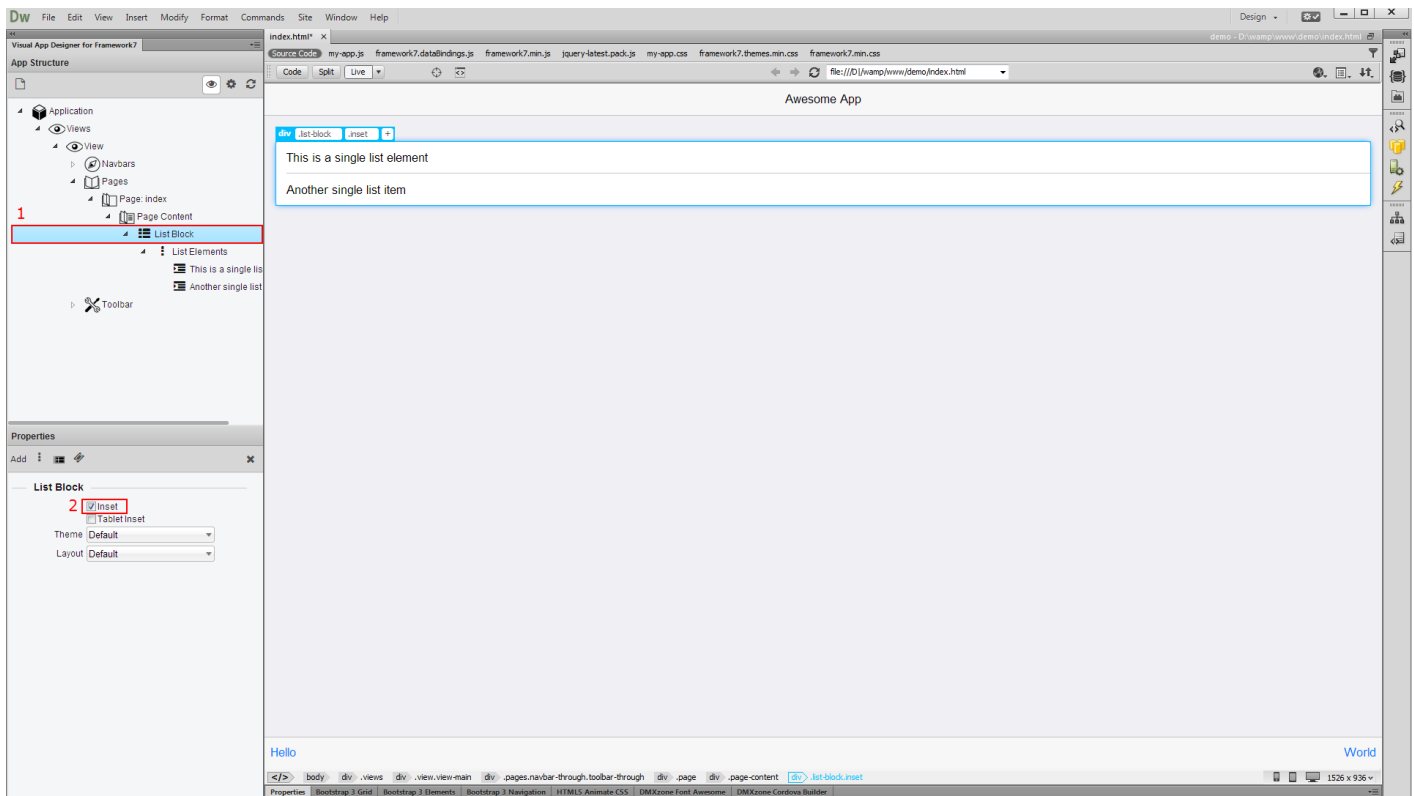




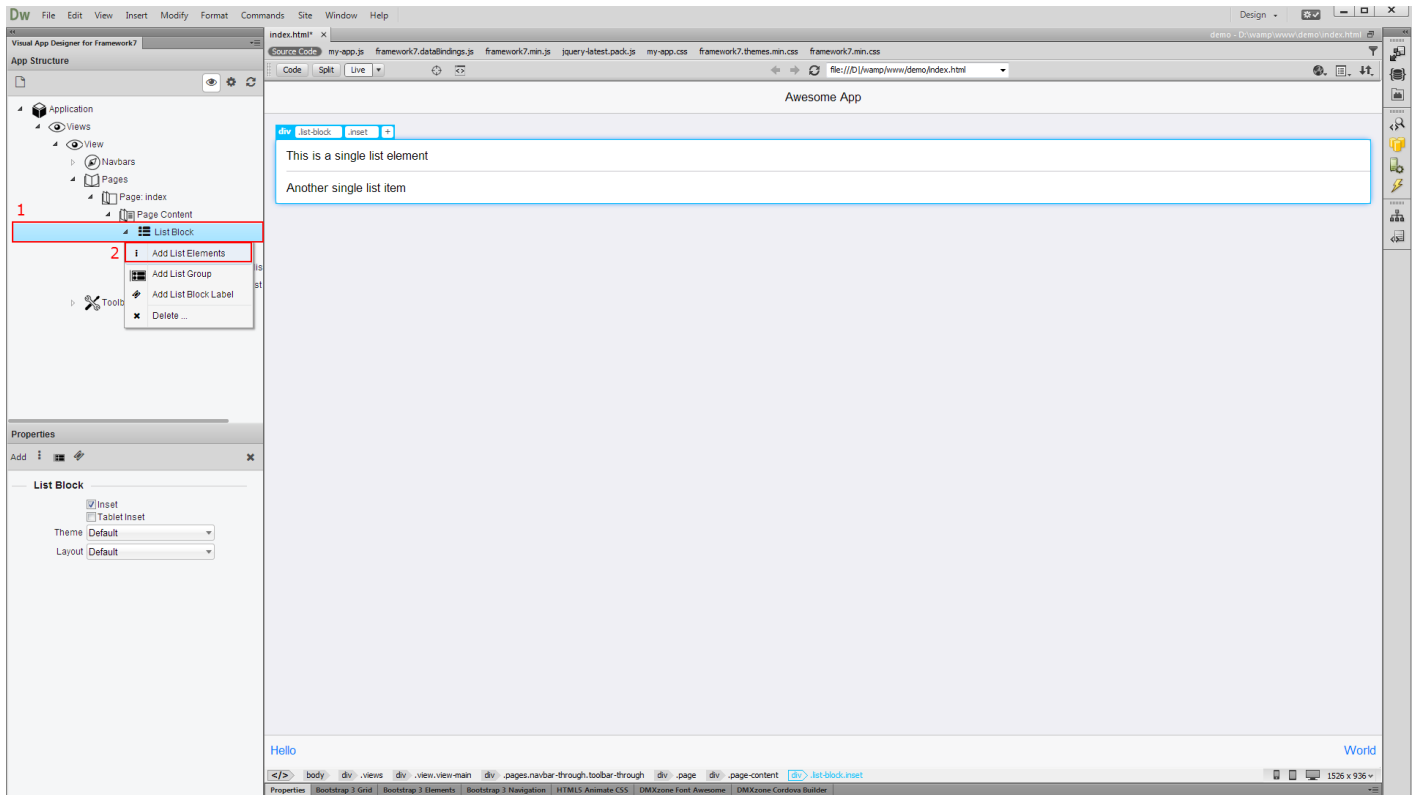
5. You can edit the list item properties, by **selecting it (1)**. Let's change **the list item name (2)**. You can add as many list items, as you need.



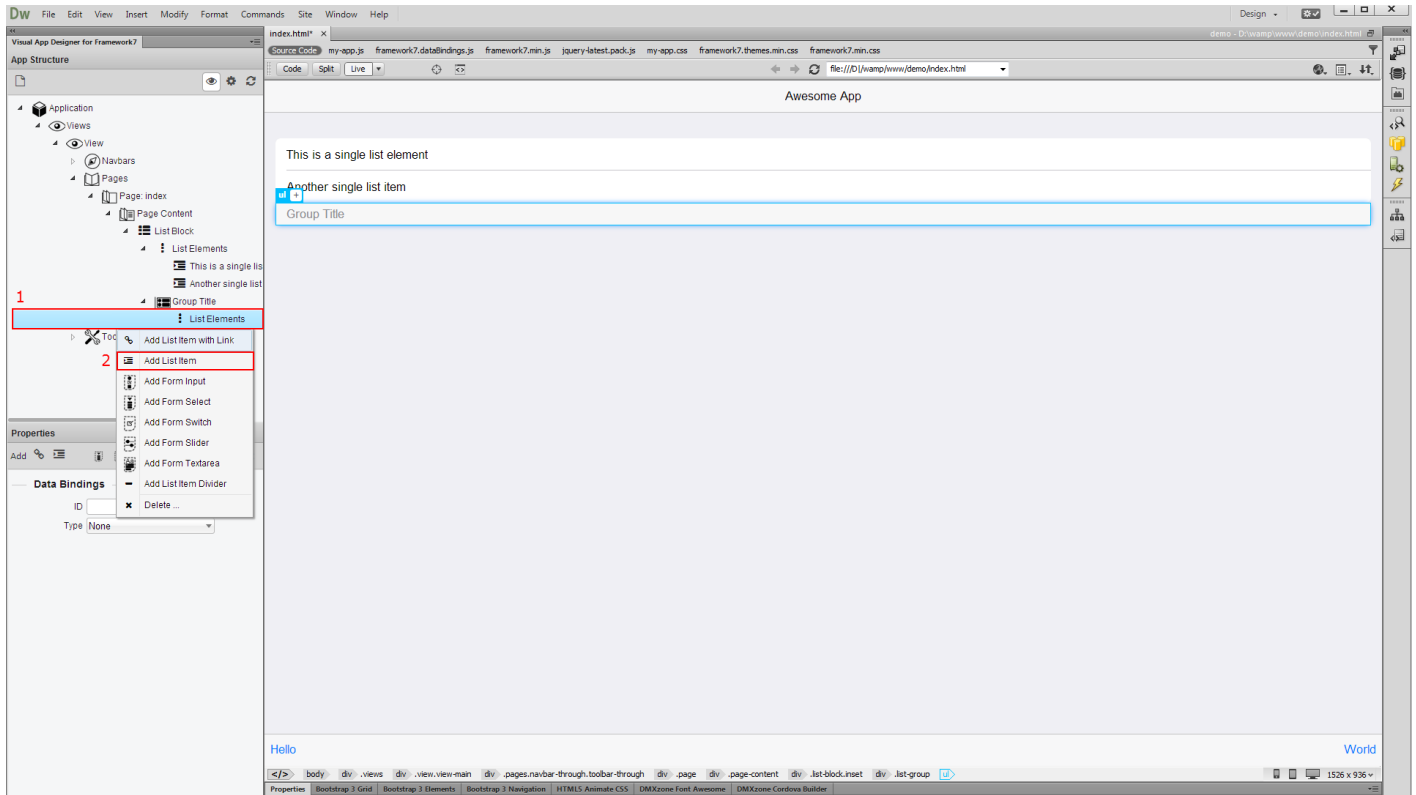
- When you are done adding list items, **select the list block (1)** and check the inset checkbox, if you need the **list block to be inset (2)**. You can select whether the list block should be inset only on tablets, by checking the tablet inset checkbox.



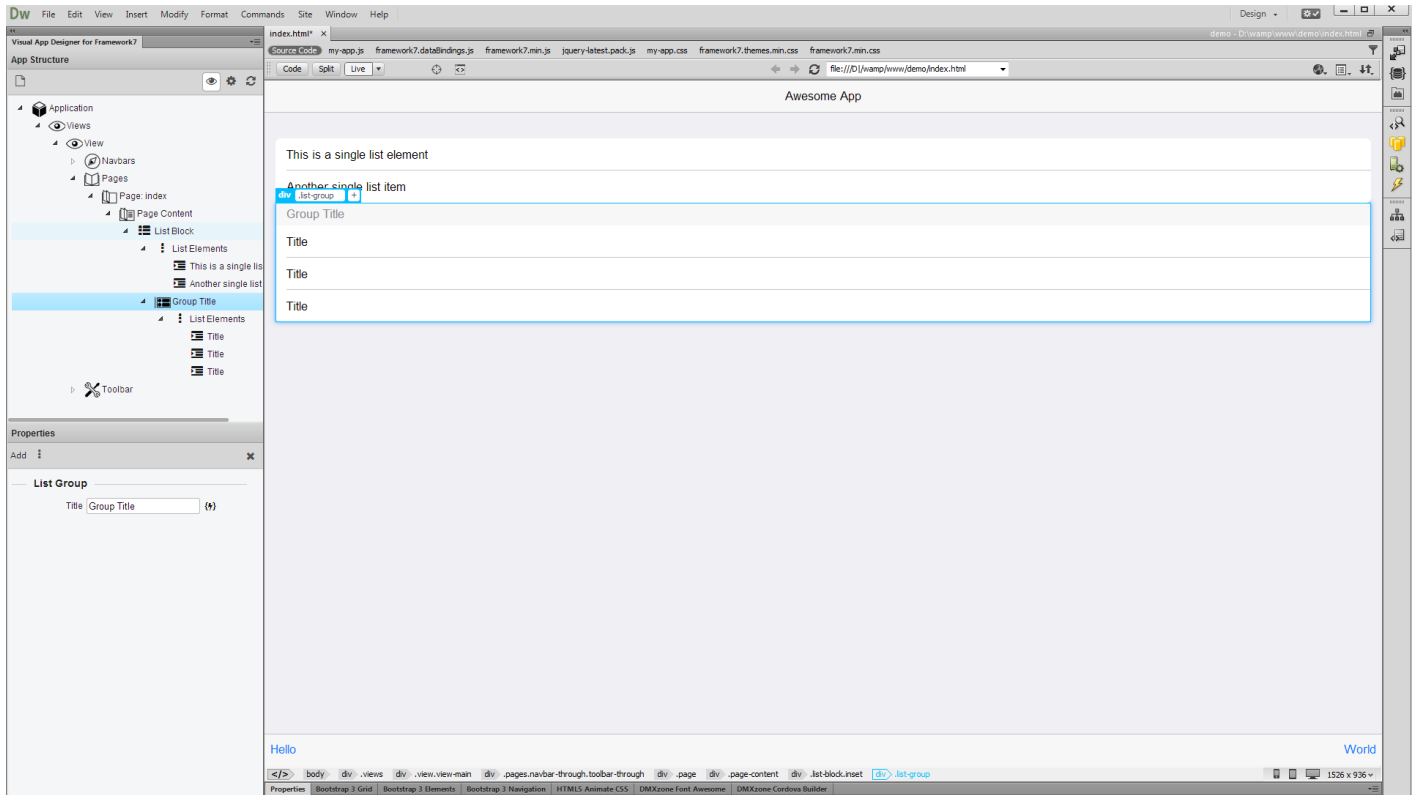
7. Now, let's add a list group into the same list block. **Right click the list block (1)** and select **add list group (2)**.



- The list group has a title and you can edit it in the properties inspector. Now let's add some list elements to this list group. **Right click the list elements block (1)** and select **add list item (2)**. You can add as many list items into the list group, as you wish.

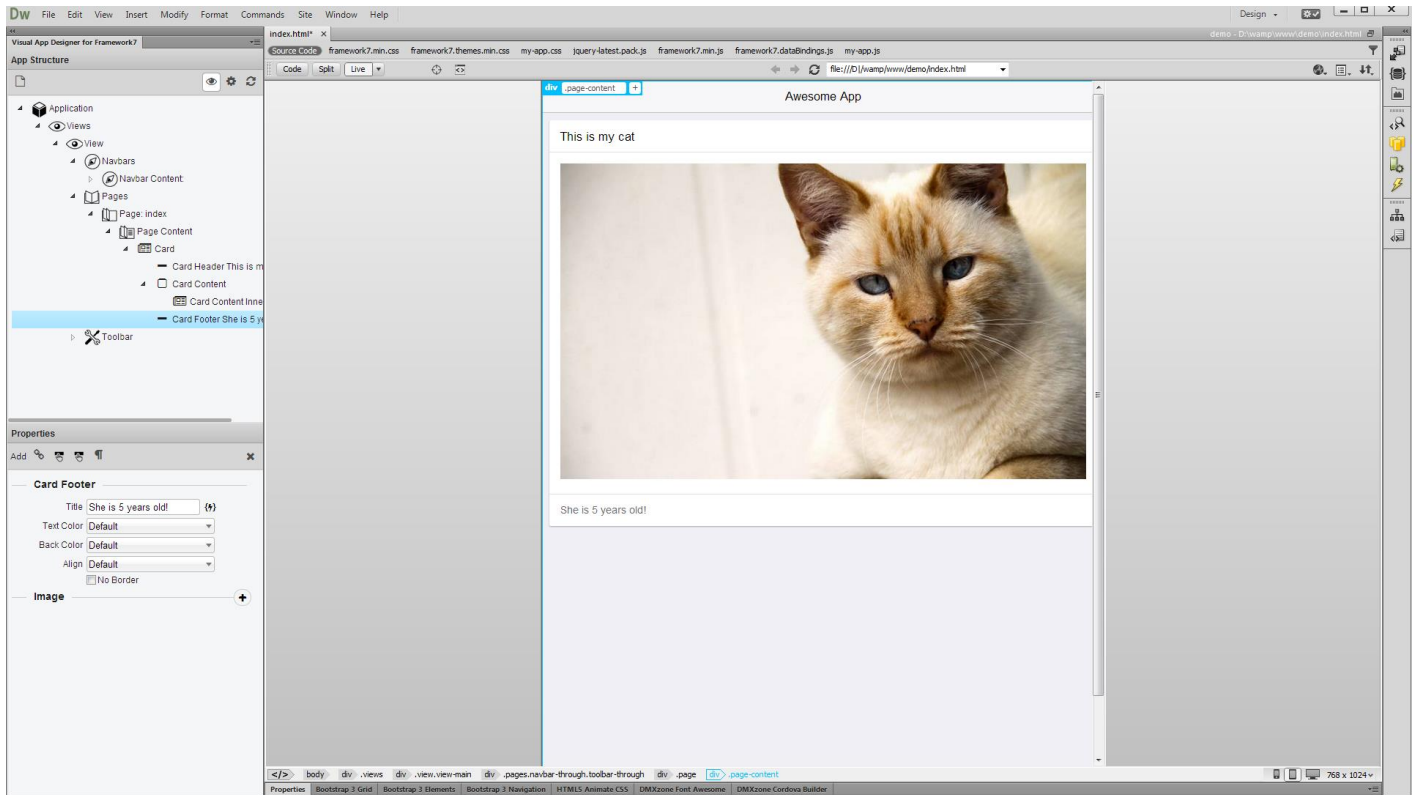


9. That's how easy it is to insert list blocks and list elements into your app layout, using DMXzone Visual App Designer for Framework 7.



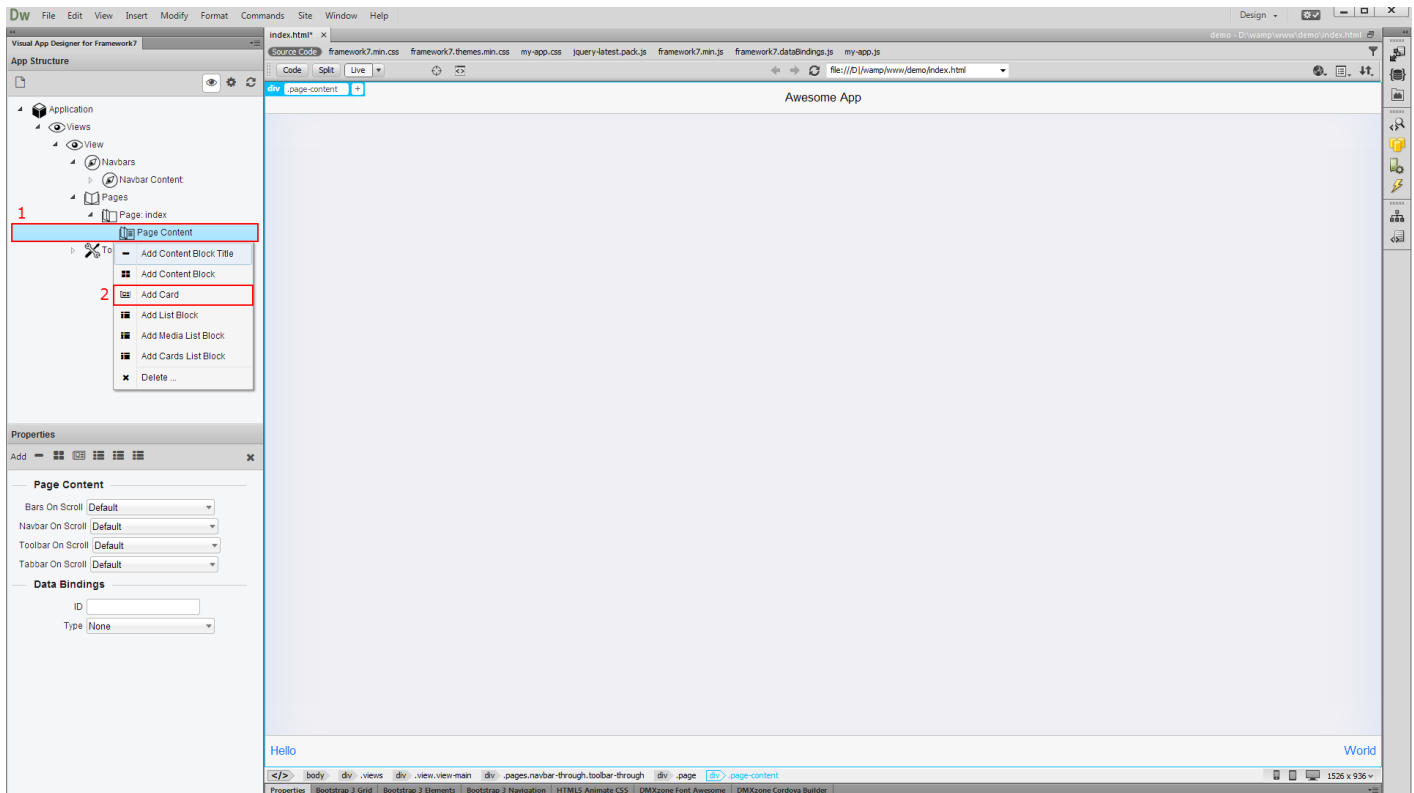
## Advanced: Insert Cards with Header, Footer and Content

In this tutorial we will show you how to use cards with [DMXzone Visual App Designer for Framework 7](#).  
\*You can also check out the [video](#).

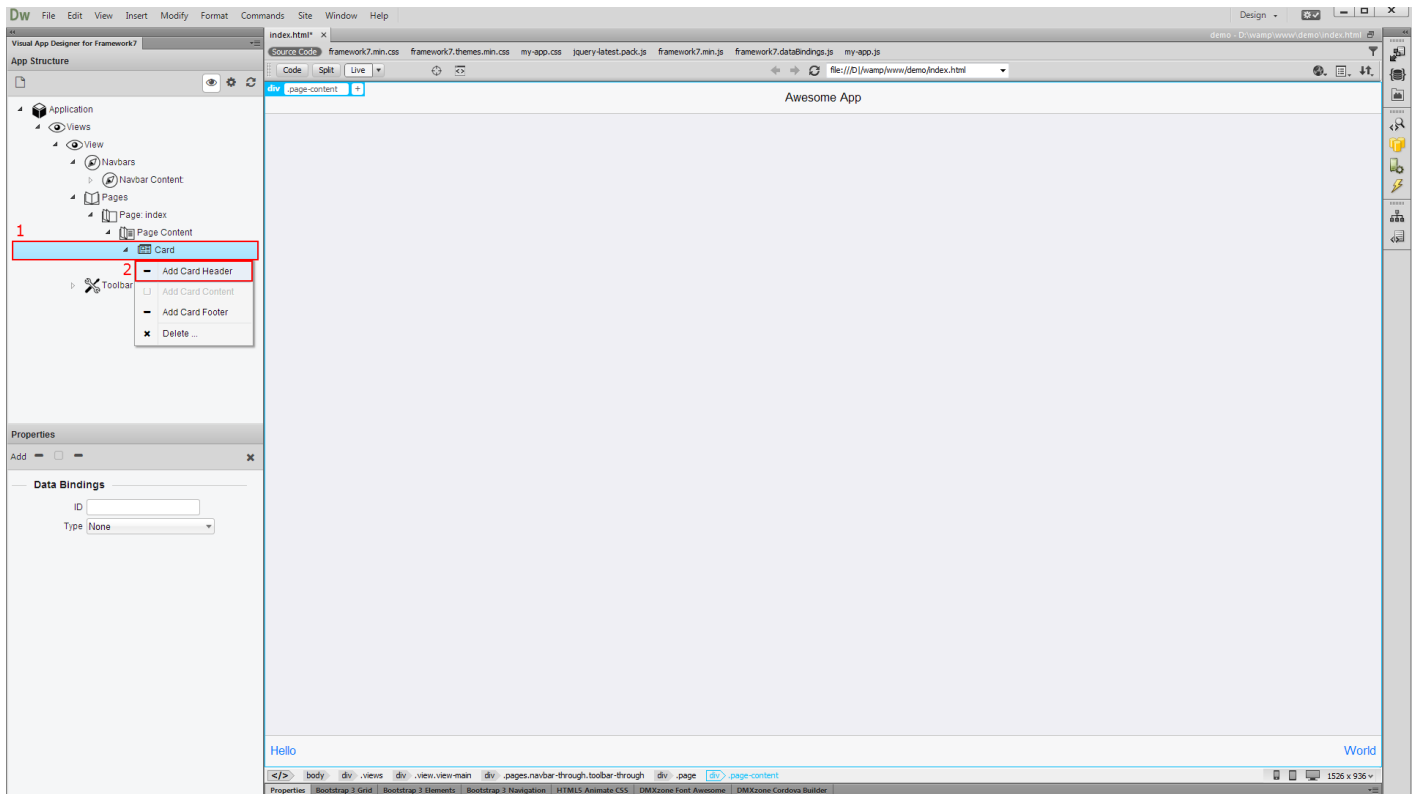


## How to do it

1. Cards are a great way to organize your information and contain unique related data, for example, a photo, text, and link about a single subject. In the app structure **right click your page content (1)** and select **add card (2)**.

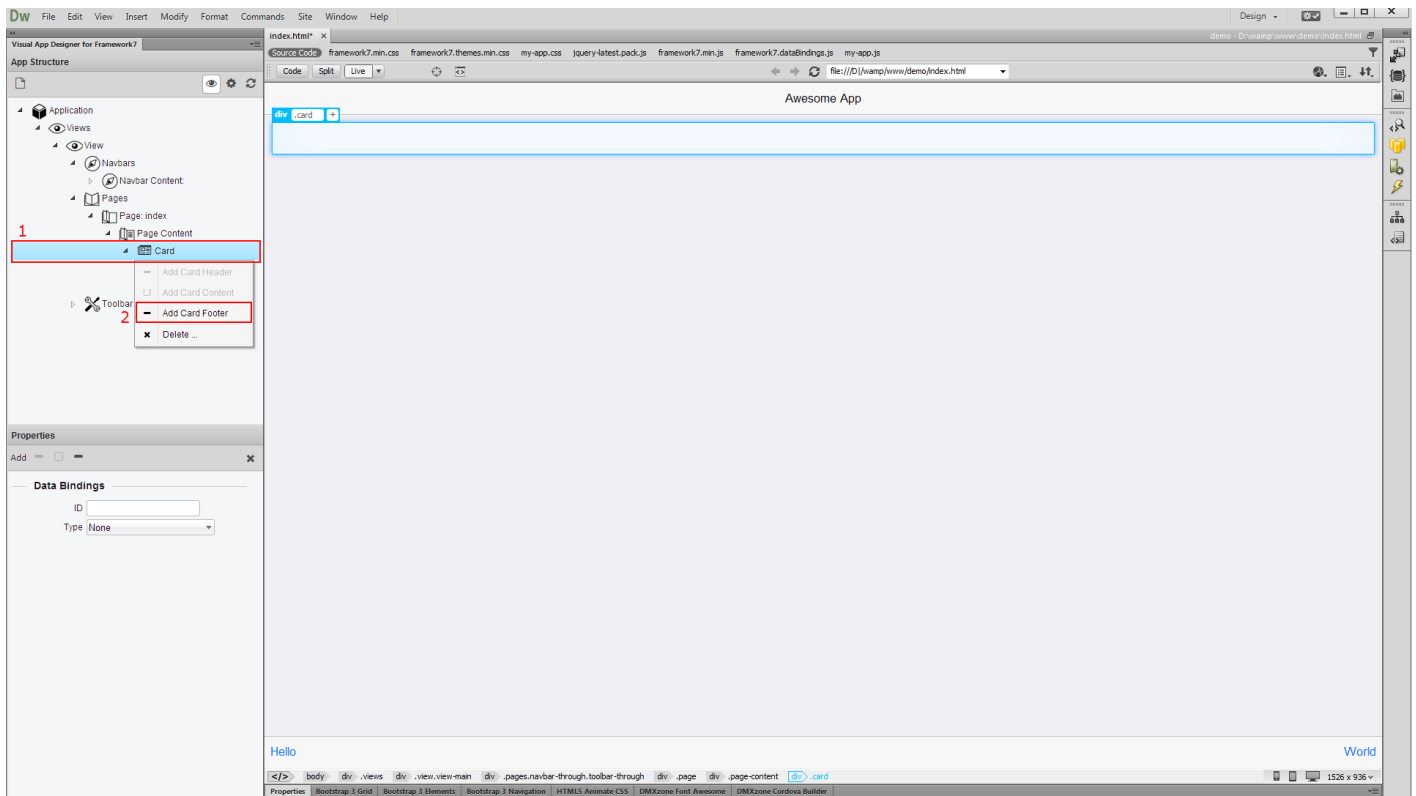


- The cards contain 3 main elements: card header, card content and card footer. Usually, the header is used to display the card title and the footer is used for some additional information or for custom actions or links. To add a header, **right click card** in the app structure (1) and select **add card header** (2).

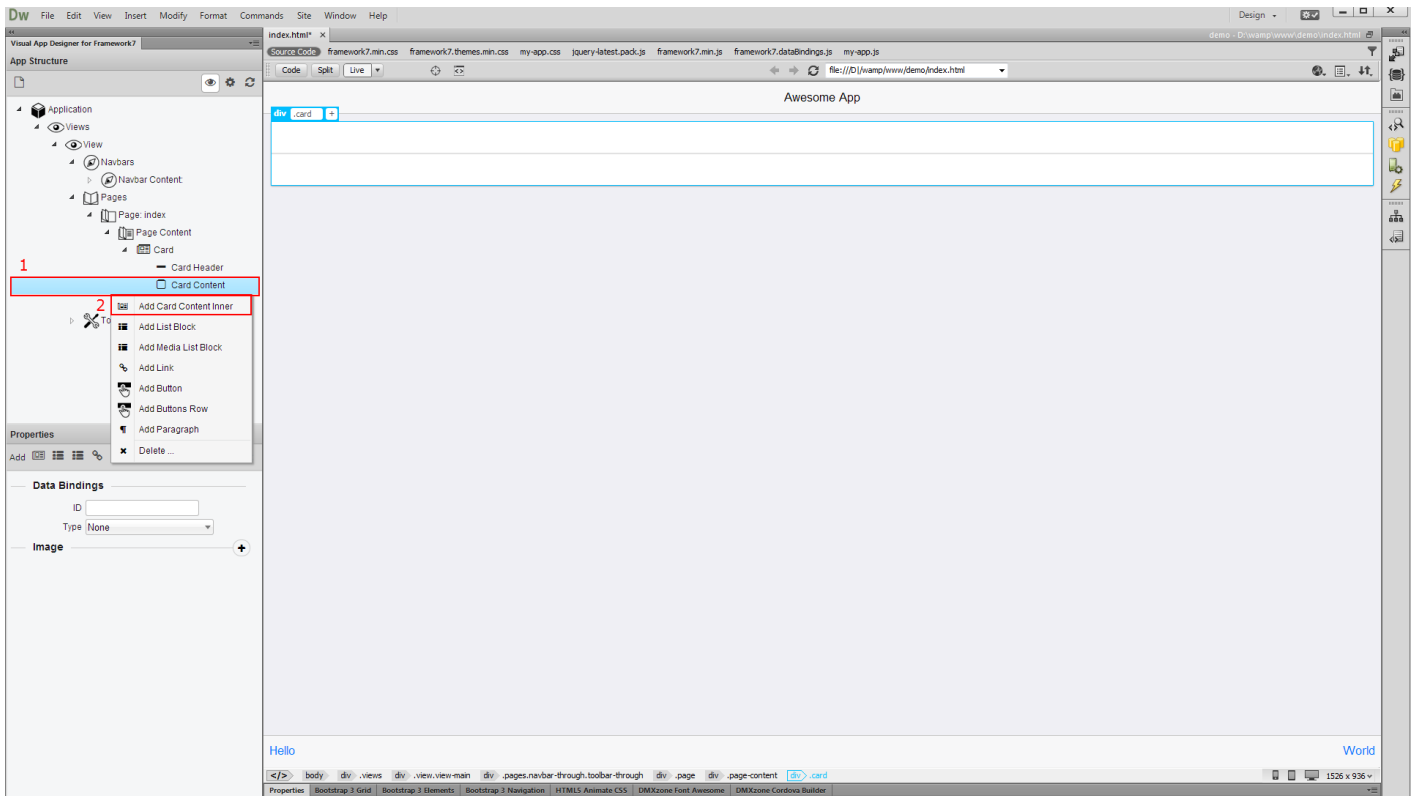




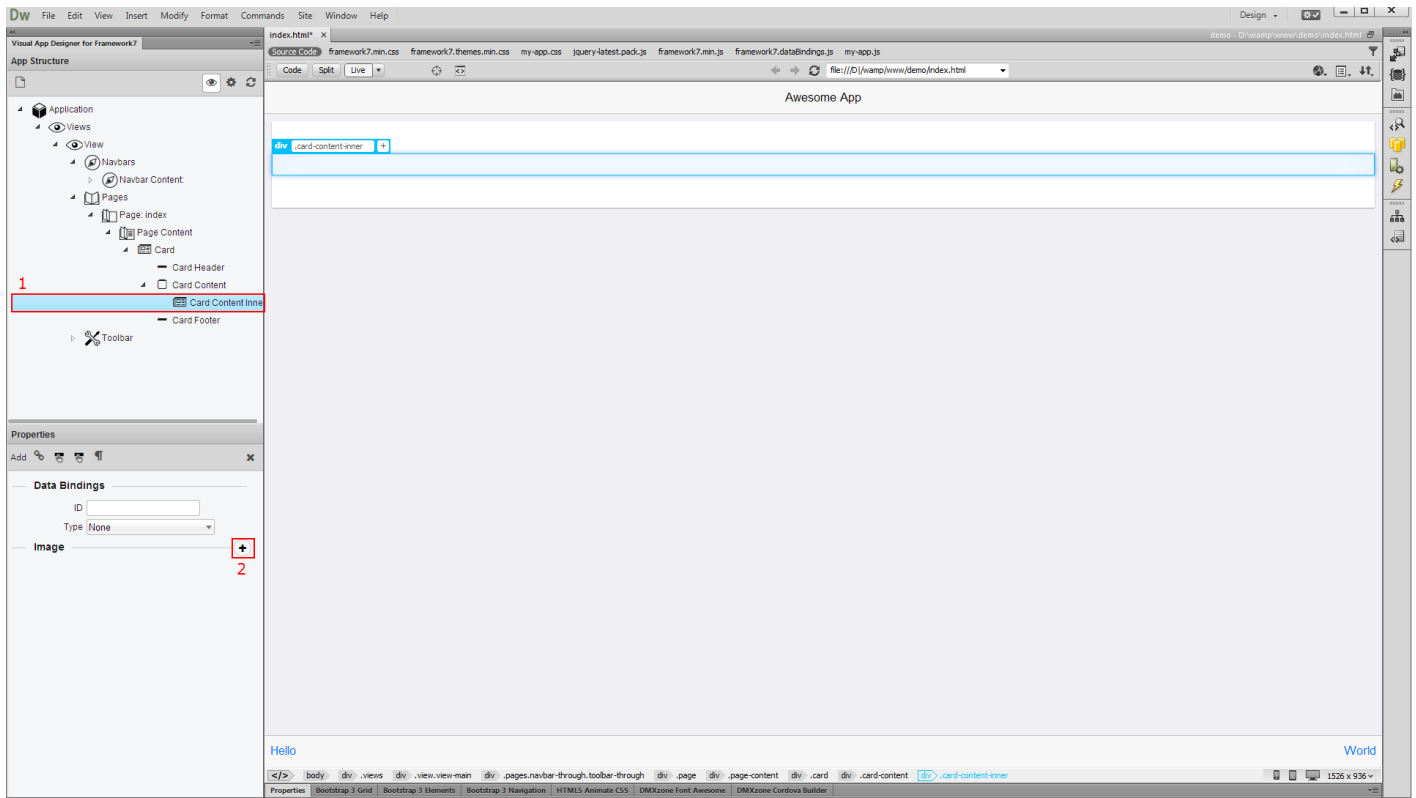
3. The same way we add footer, **right click card** in the app structure **(1)** and select **add card footer (2)**.



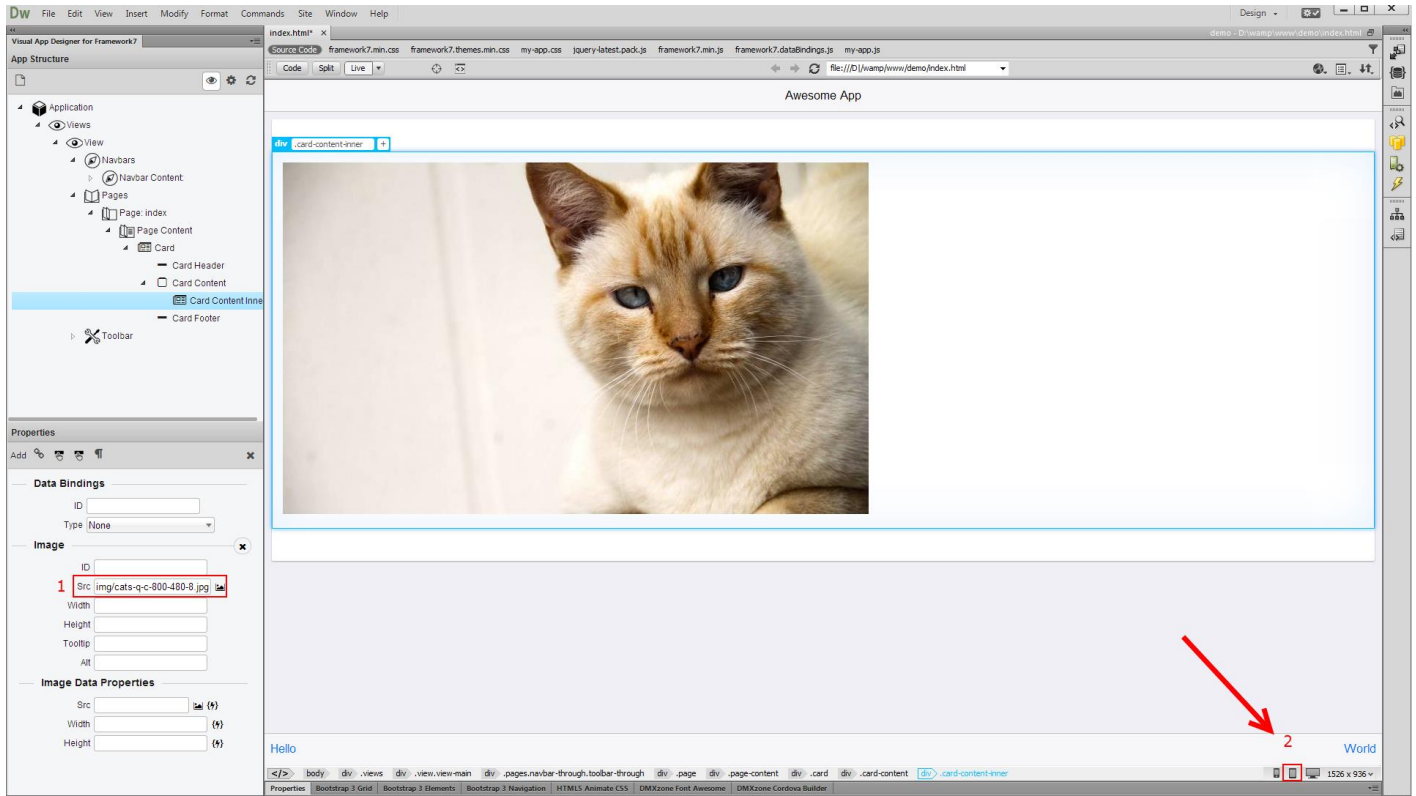
4. Right click the **card content wrapper (1)** and select **add card content inner (2)**.



5. Select **card content inner** (1) and in the properties click **add image button** (2).



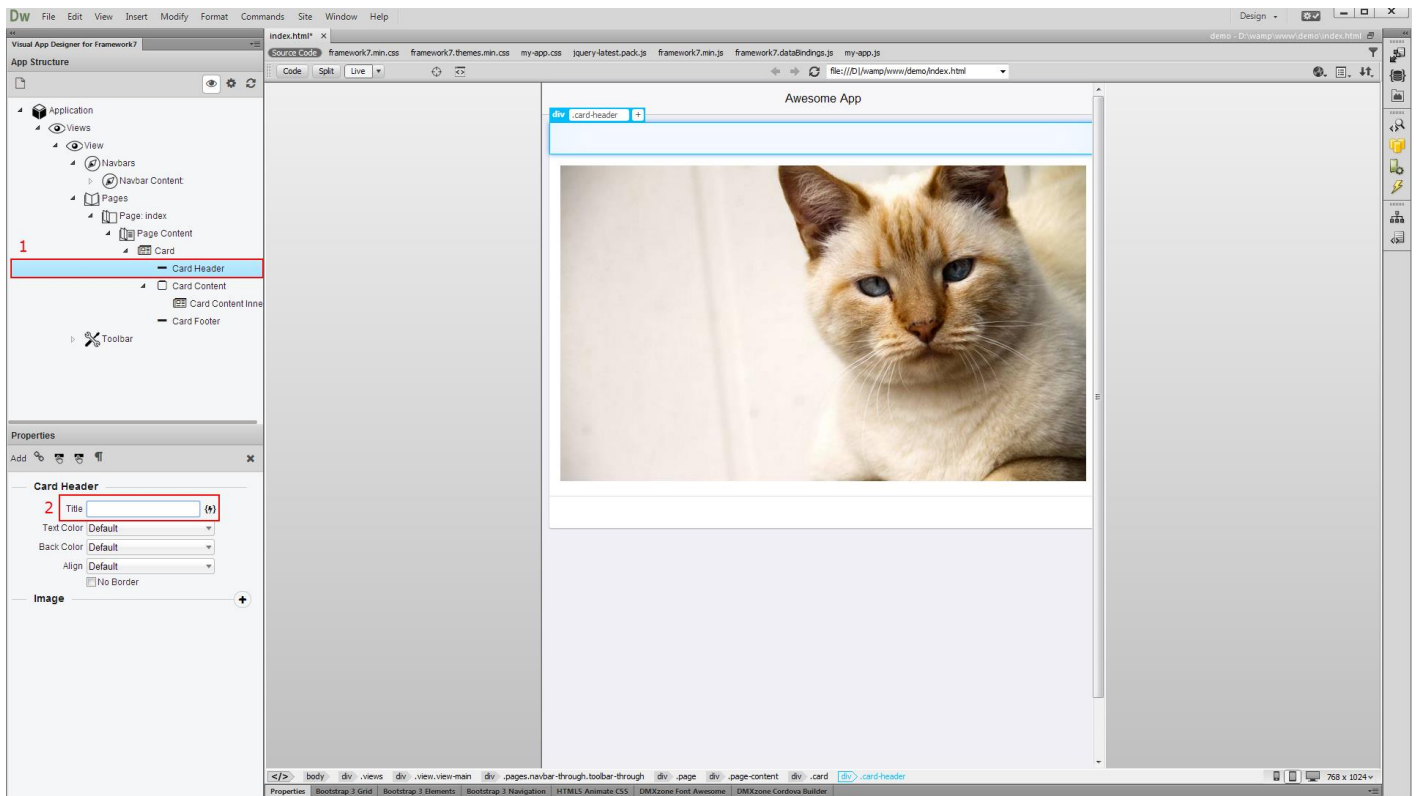
6. Select your **image source (1)** and switch to **tablet layout** for more realistic preview (2).



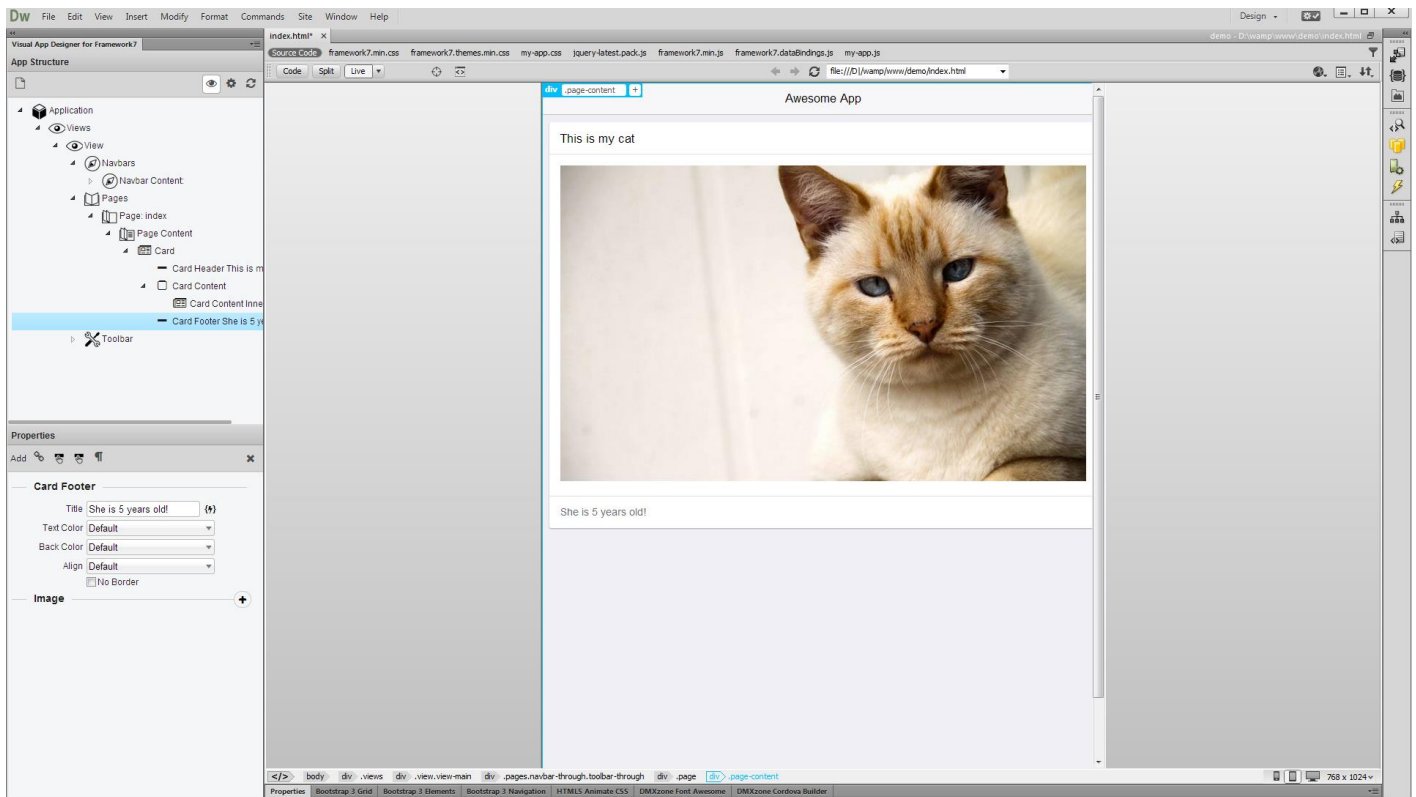
## 7. Enter 100% in the image width field.

The screenshot displays the DMXzone Visual App Designer interface. The main workspace shows a preview of an 'Awesome App' with a card containing a photo of a Siamese cat. The 'Properties' panel on the left is open to the 'Image' section. The 'Width' field is highlighted with a red box and contains the text '100%'. The 'Image Data Properties' section below it shows fields for 'Src', 'Width', and 'Height', each with a magnifying glass icon. The 'App Structure' panel on the left shows a tree view of the application components, with 'Card Content Inner' selected. The status bar at the bottom indicates the current page is 'card-content-inner' and the window size is 768 x 1024.

- Let's add a title for our card. Click the **card header (1)** and in the properties **enter the title (2)**. We do the same for the card footer.



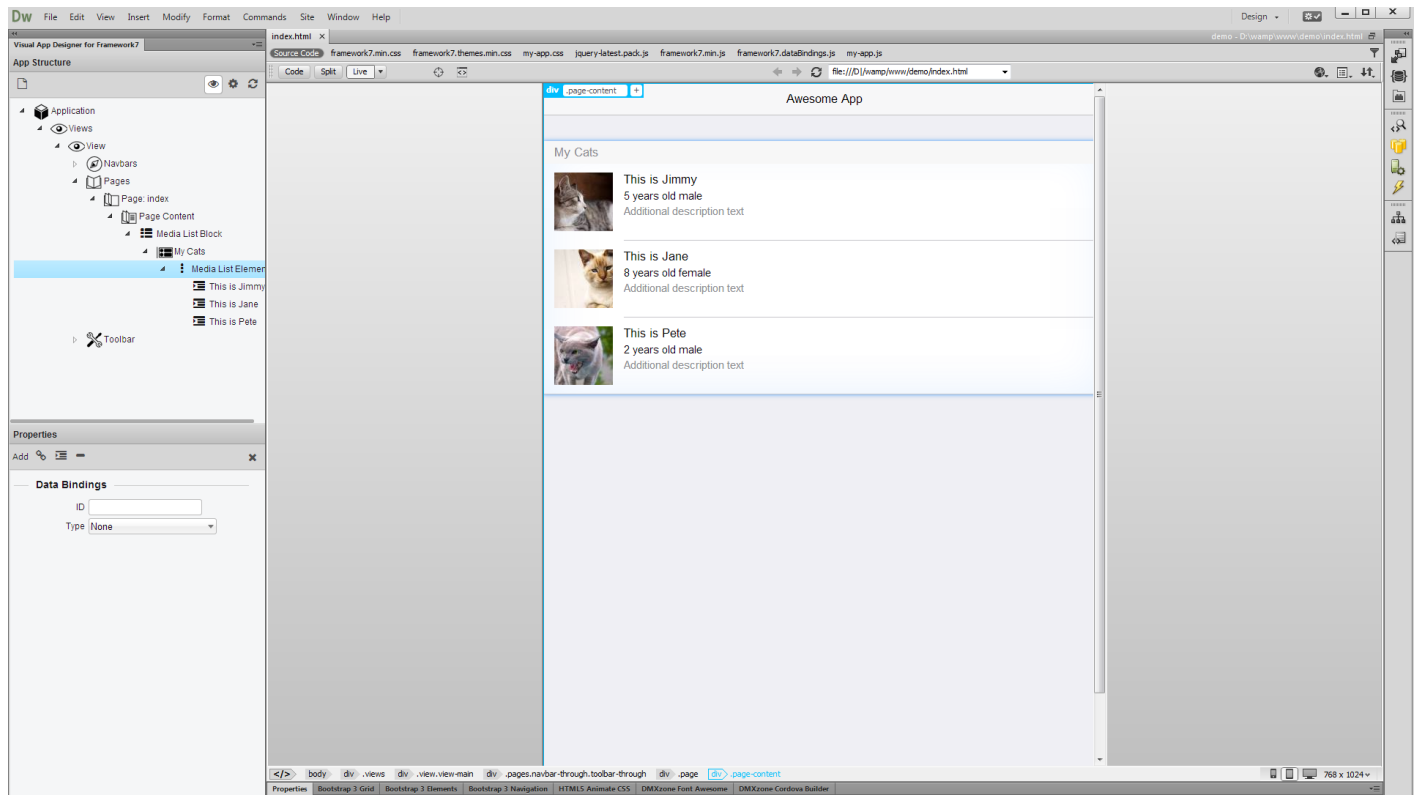
- Now our card header, footer and content are done. That's how easy it is to use cards with DMXzone Visual App Designer for Framework 7.



## Advanced: Using Media Lists in DMXzone Visual App Designer

In this tutorial we will show you how to insert media list blocks with [DMXzone Visual App Designer for Framework 7](#).

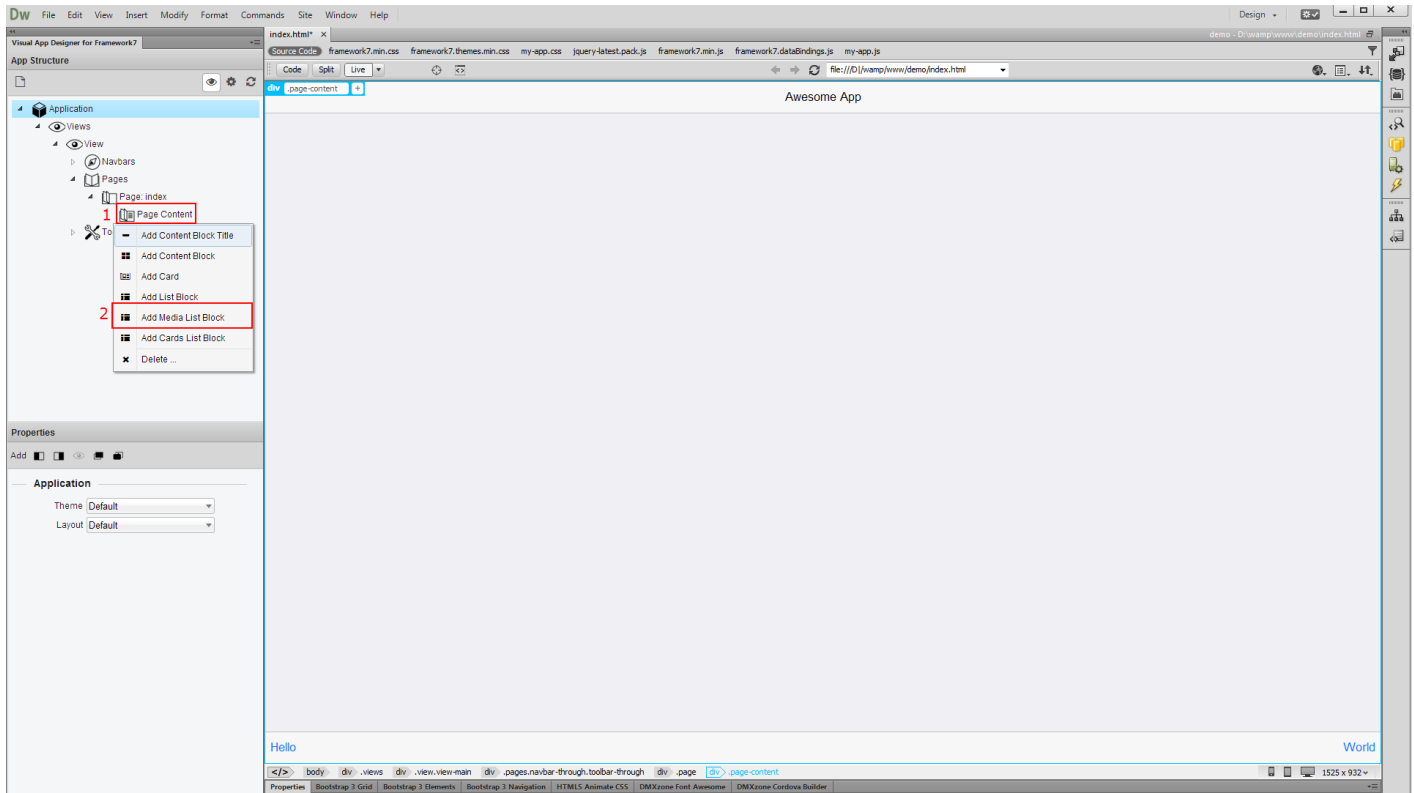
\*You can also check out the [video](#).



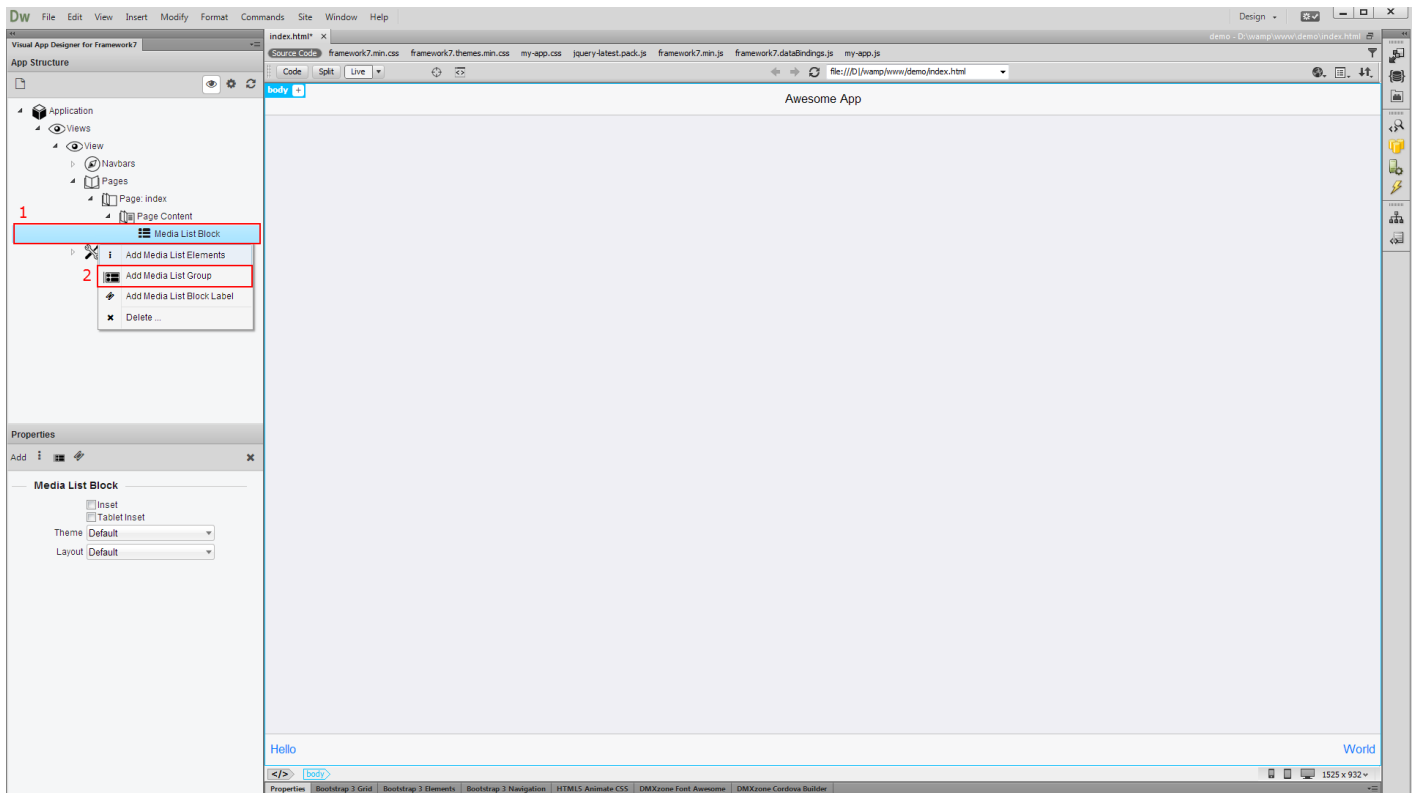


## How to do it

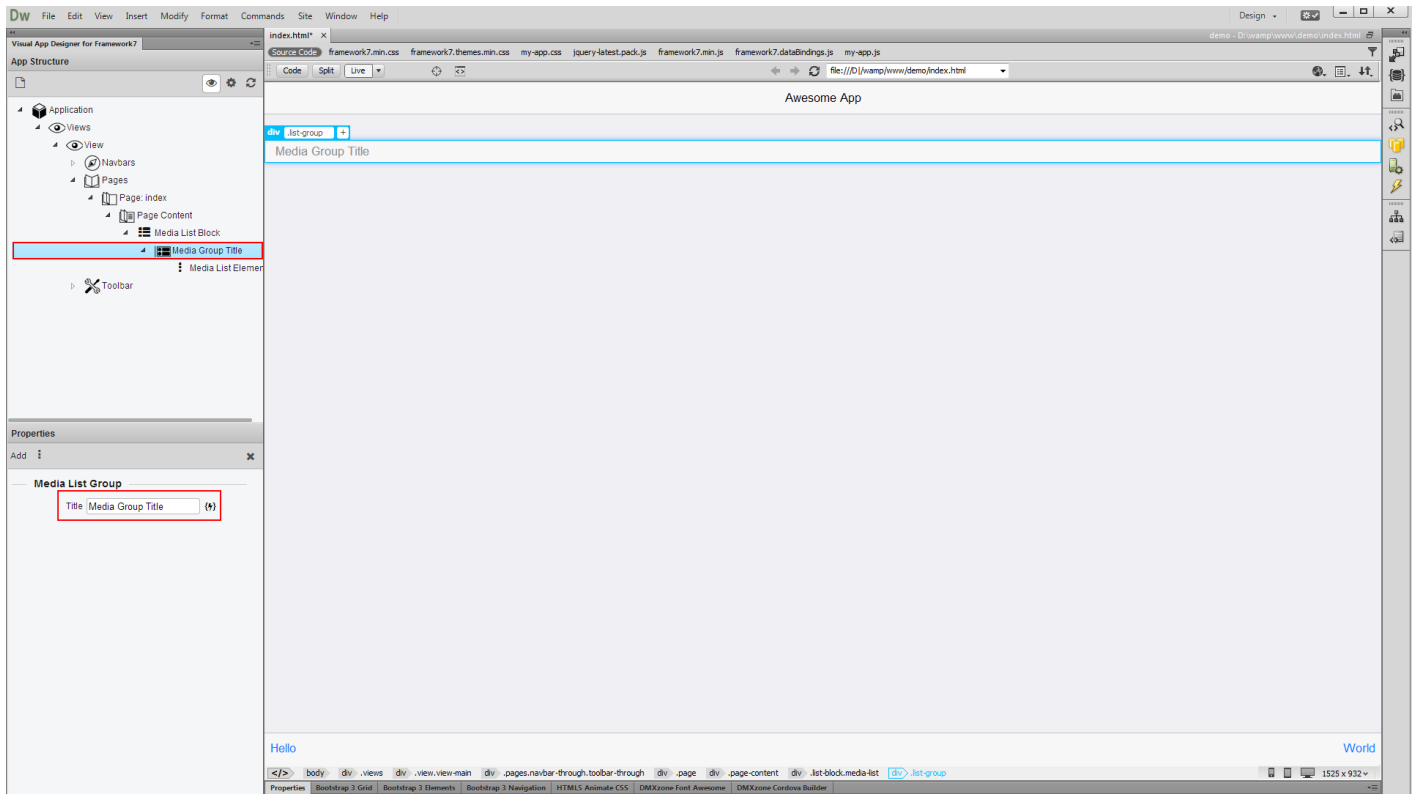
1. Right click the page content wrapper (1) in the app structure and select add media list block (2).



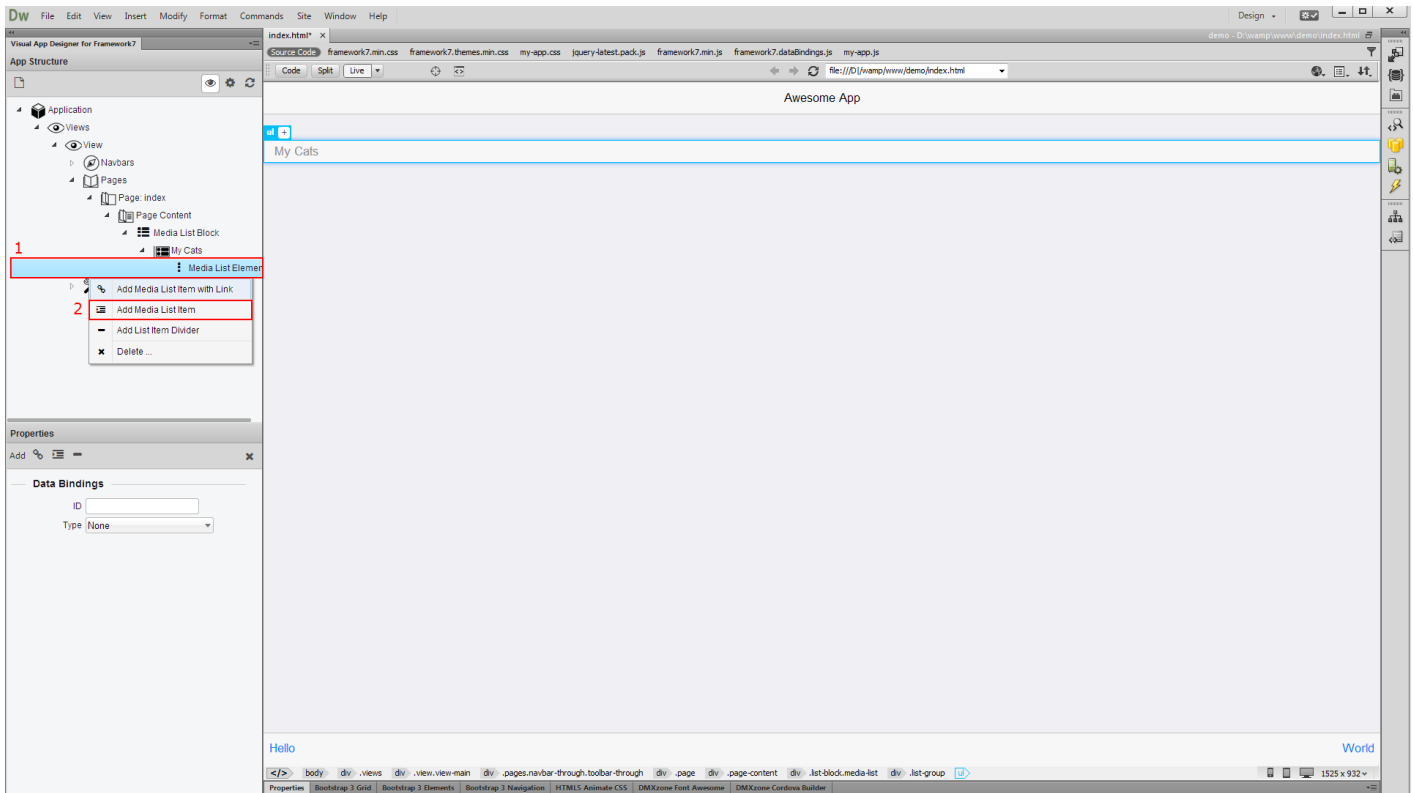
2. Right click the **media list block (1)** and select **media list group (2)**.



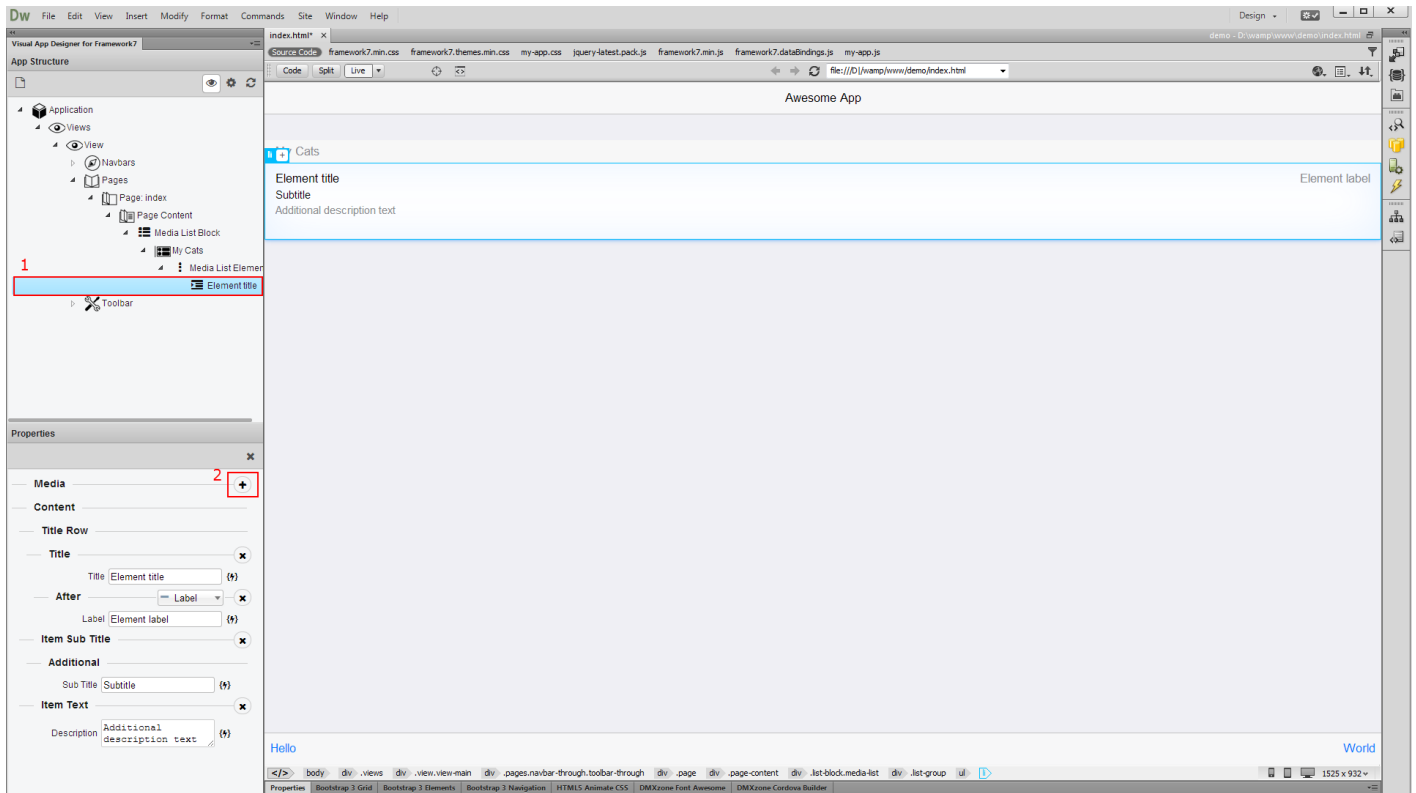
3. Click the **media list group title (1)** in order to edit it in the **properties (2)**.



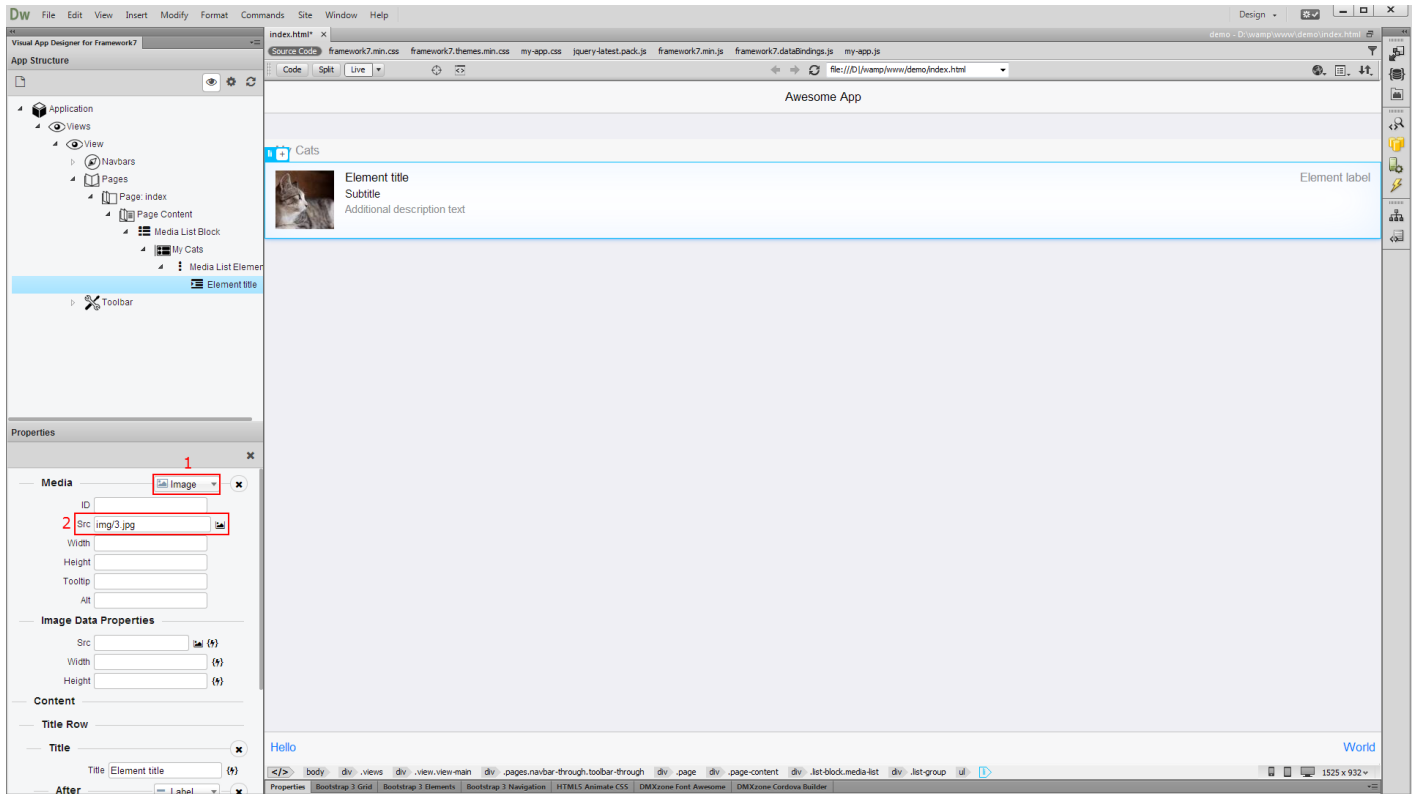
- Right click the **media list elements (1)**, when you are done changing the title and add **media list item (2)**.



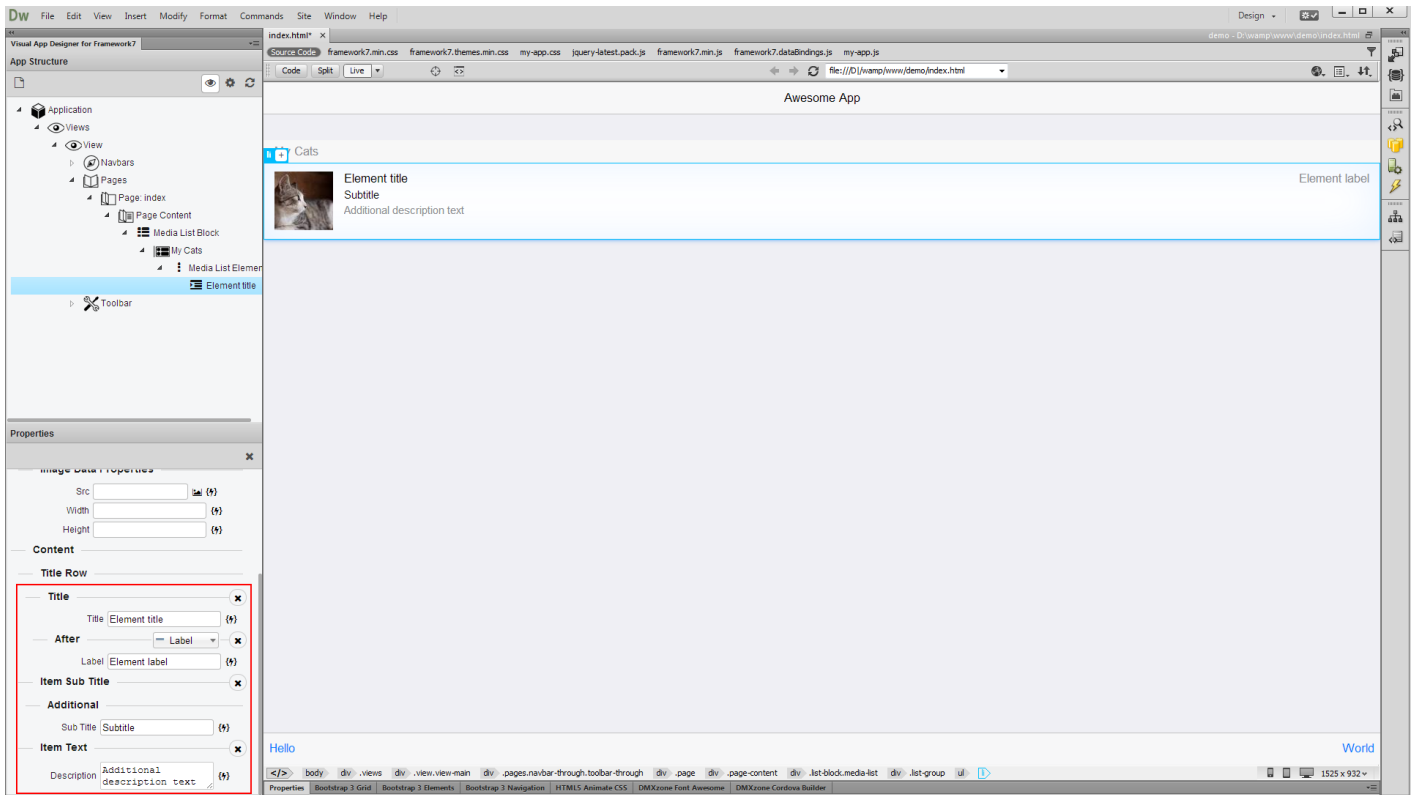
5. Select the **media list item (1)**, that we've just added and click the **add media button (2)** in the properties inspector.



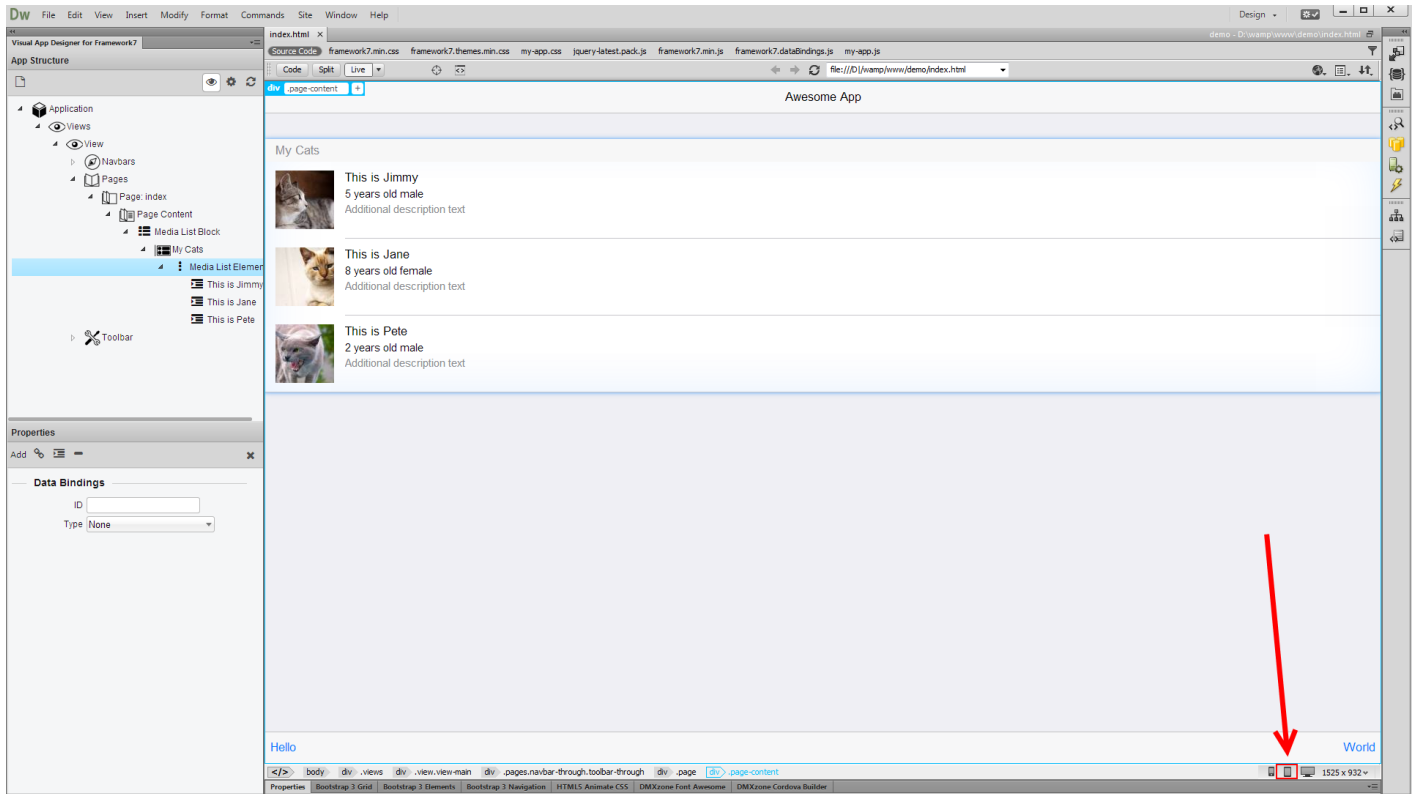
6. Select the media type, in our case it is **image (1)** and select **its source (2)**.



- You can change or remove the item title, subtitle, label and description. Also, you can add as many media list items as you need.

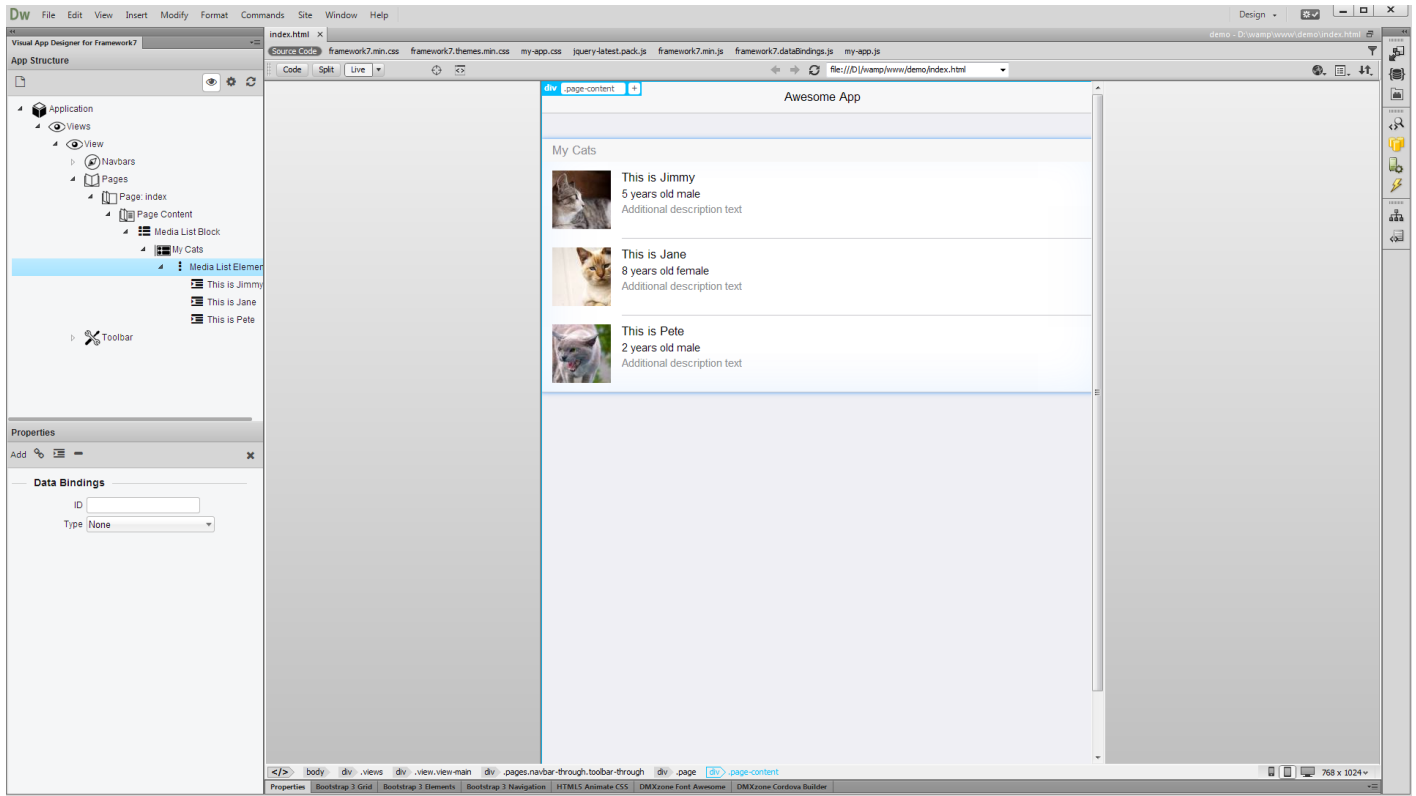


8. We are done adding our media list items, so let's switch to tablet layout, for more realistic preview.

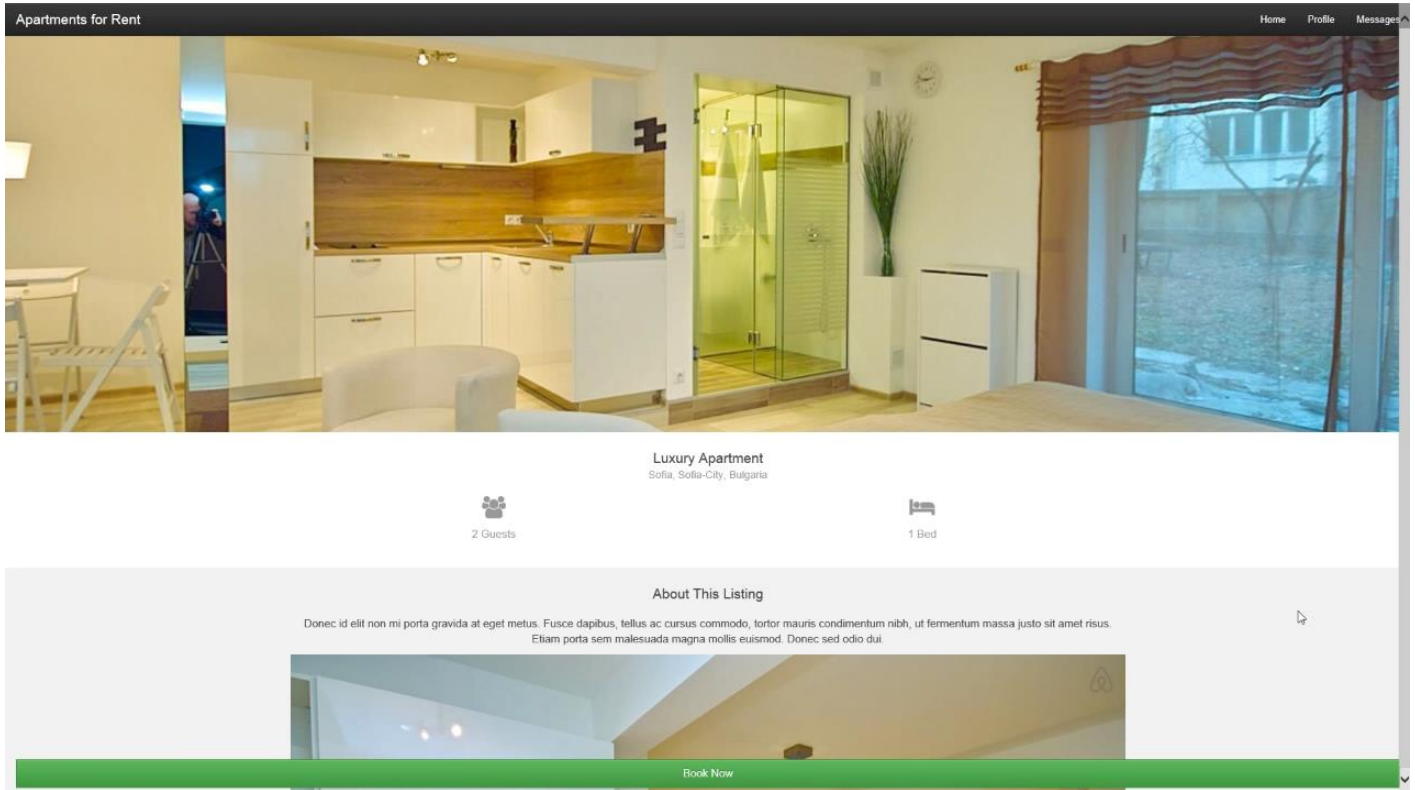




9. That's how easy it is, to insert media list blocks with DMXzone Visual App designer for Framework 7.



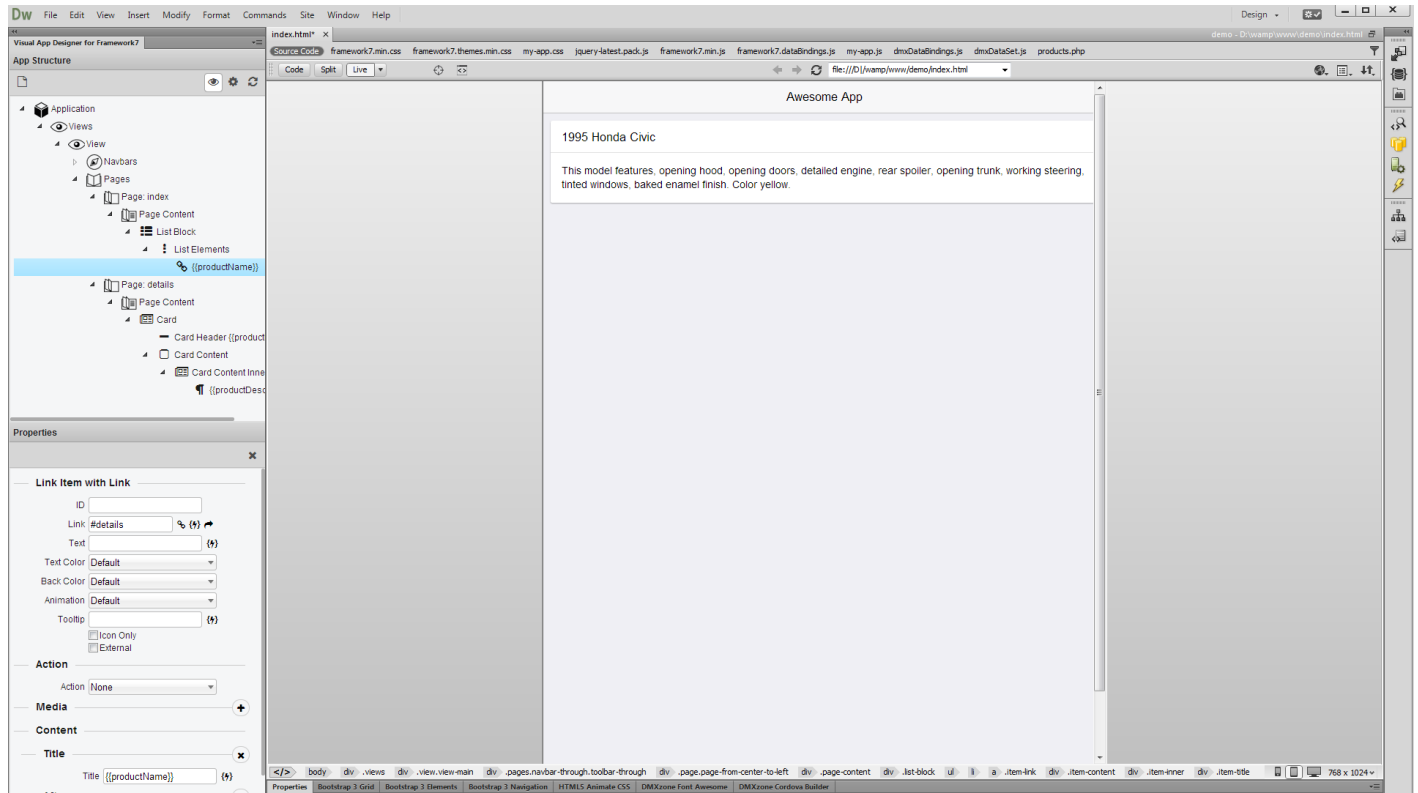
10. The emulator opens the site as a fullscreen app. That's how easy it is to build apps for Windows.



## Advanced: Building Database Driven Mobile Apps

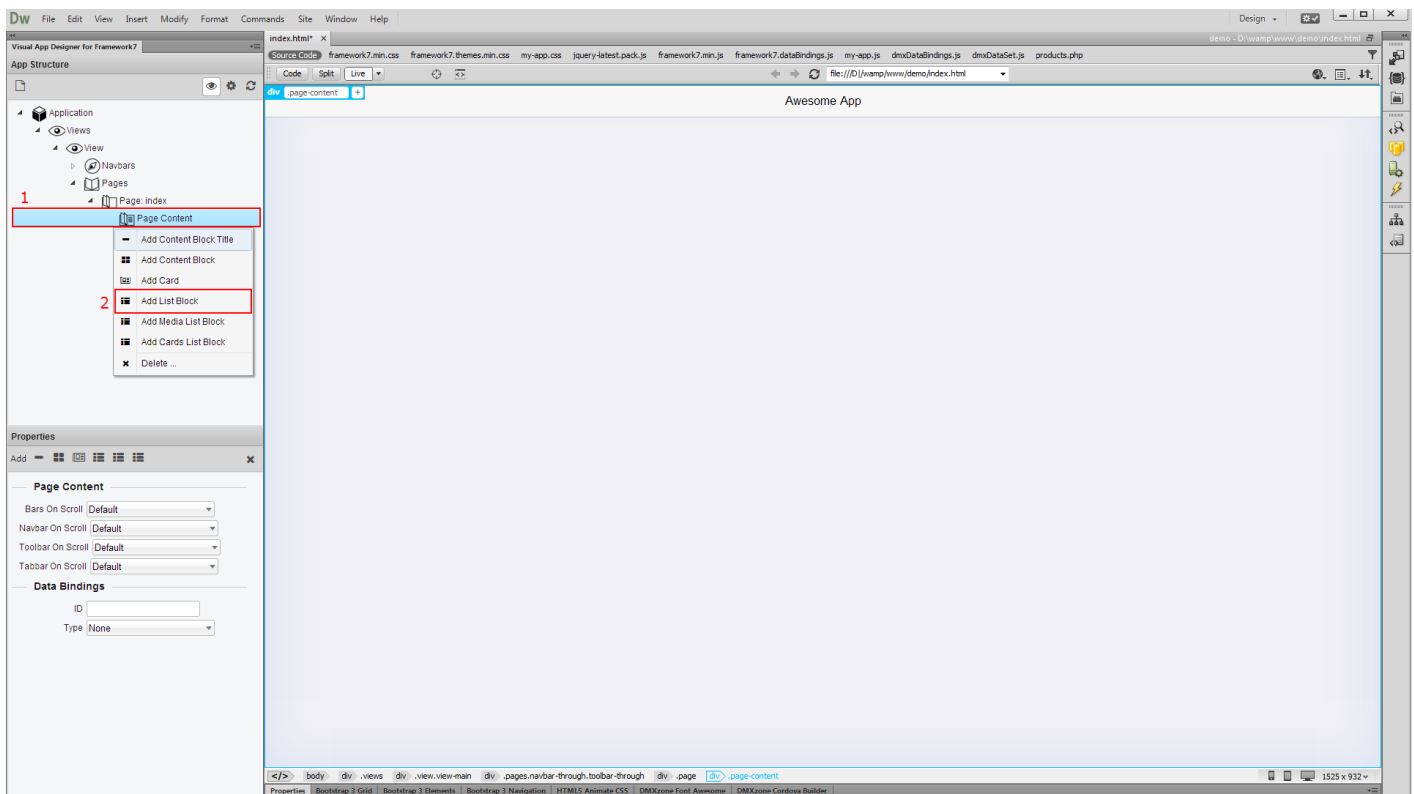
In this tutorial we will show you how to build dynamic, database driven app layout, using [DMXzone Visual App Designer for Framework 7](#).

\*You can also check out the [video](#).

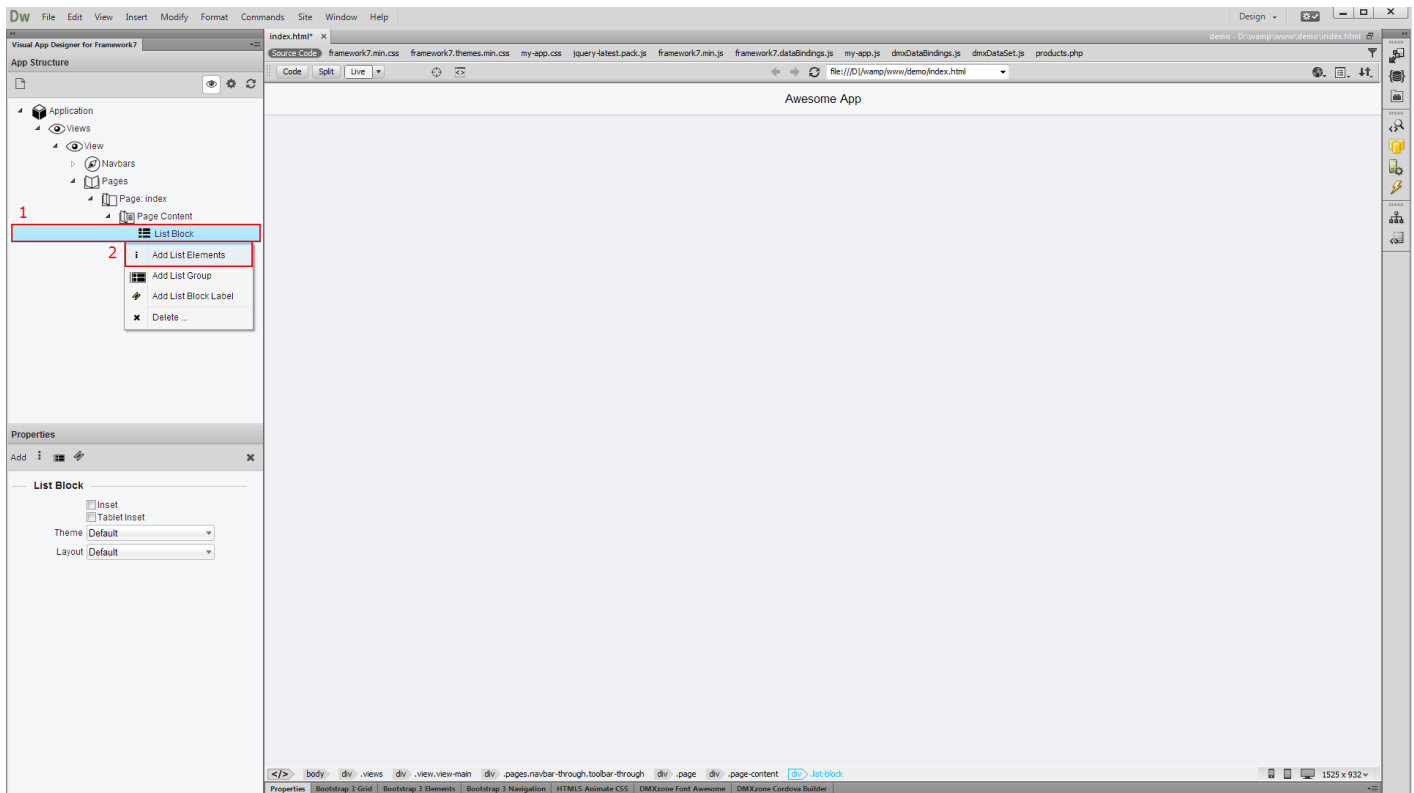


## How to do it

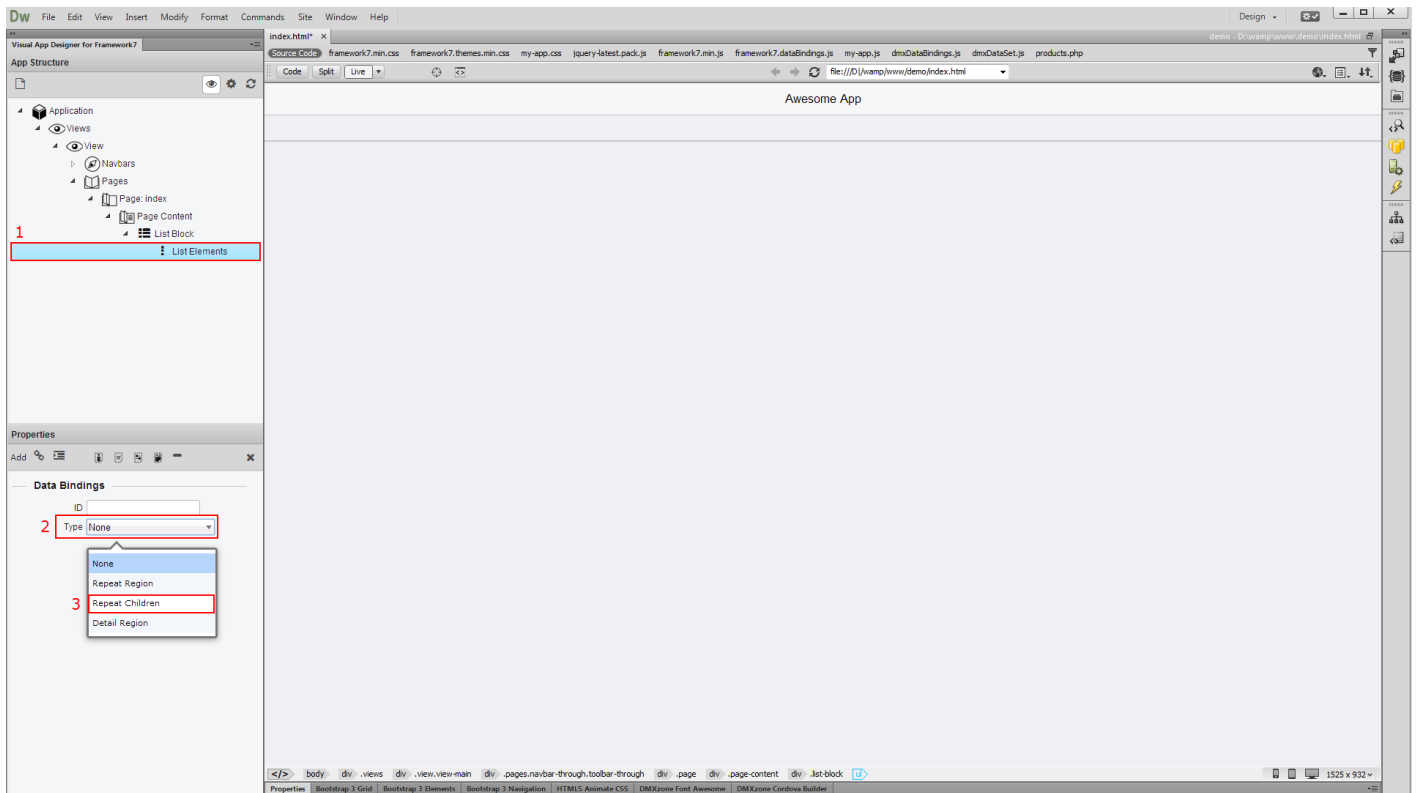
1. We are going to build an app that lists our products. When the user selects any of the products, the app will display a details page, with information about the selected product. We've created a database source, using DMXzone Database Connector and HTML5 Data Bindings extensions. Our data source is called **products**. Let's add the products list on our index page. Right click the **page content (1)** and select **add list block (2)**.



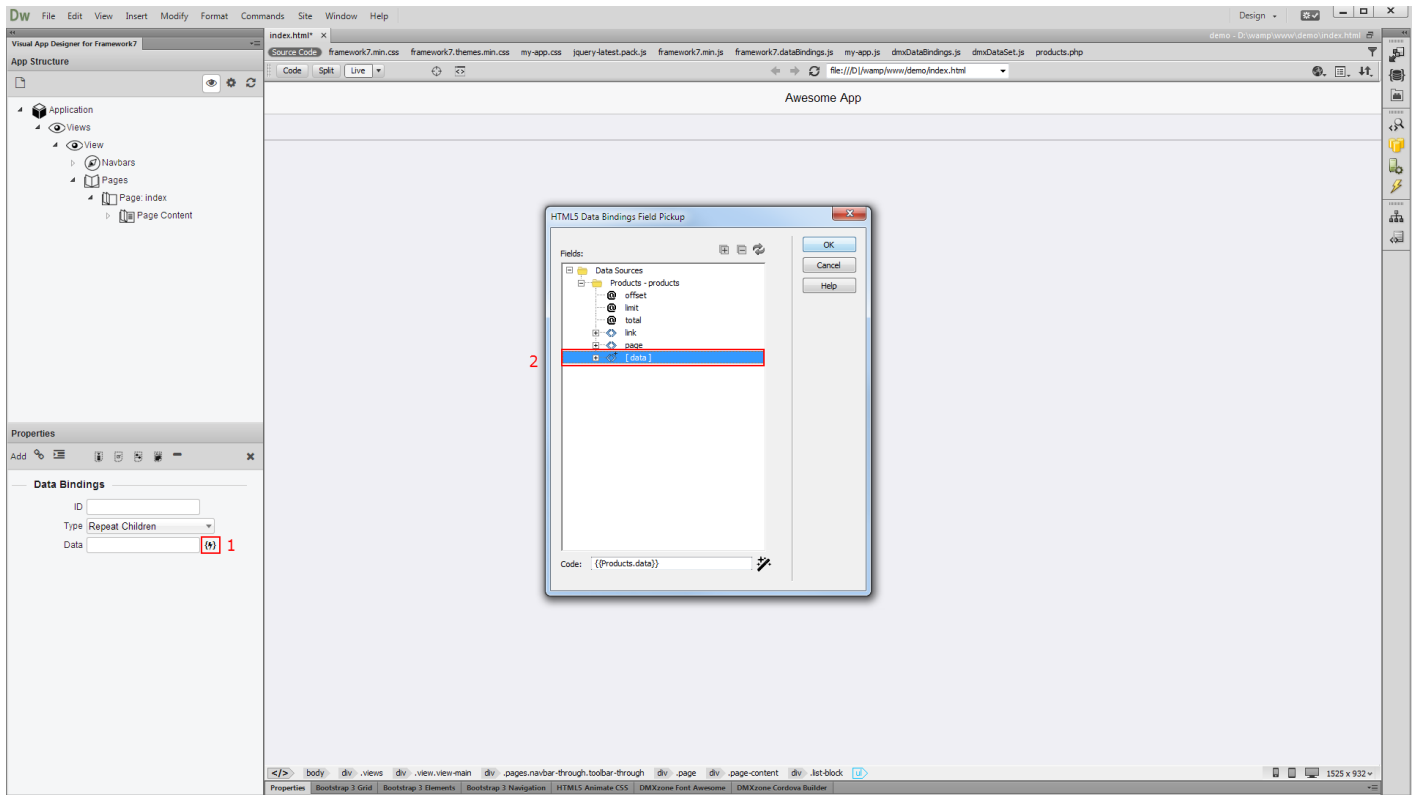
2. Right click the **list block (1)** that we've just added and select **add list elements (2)**.



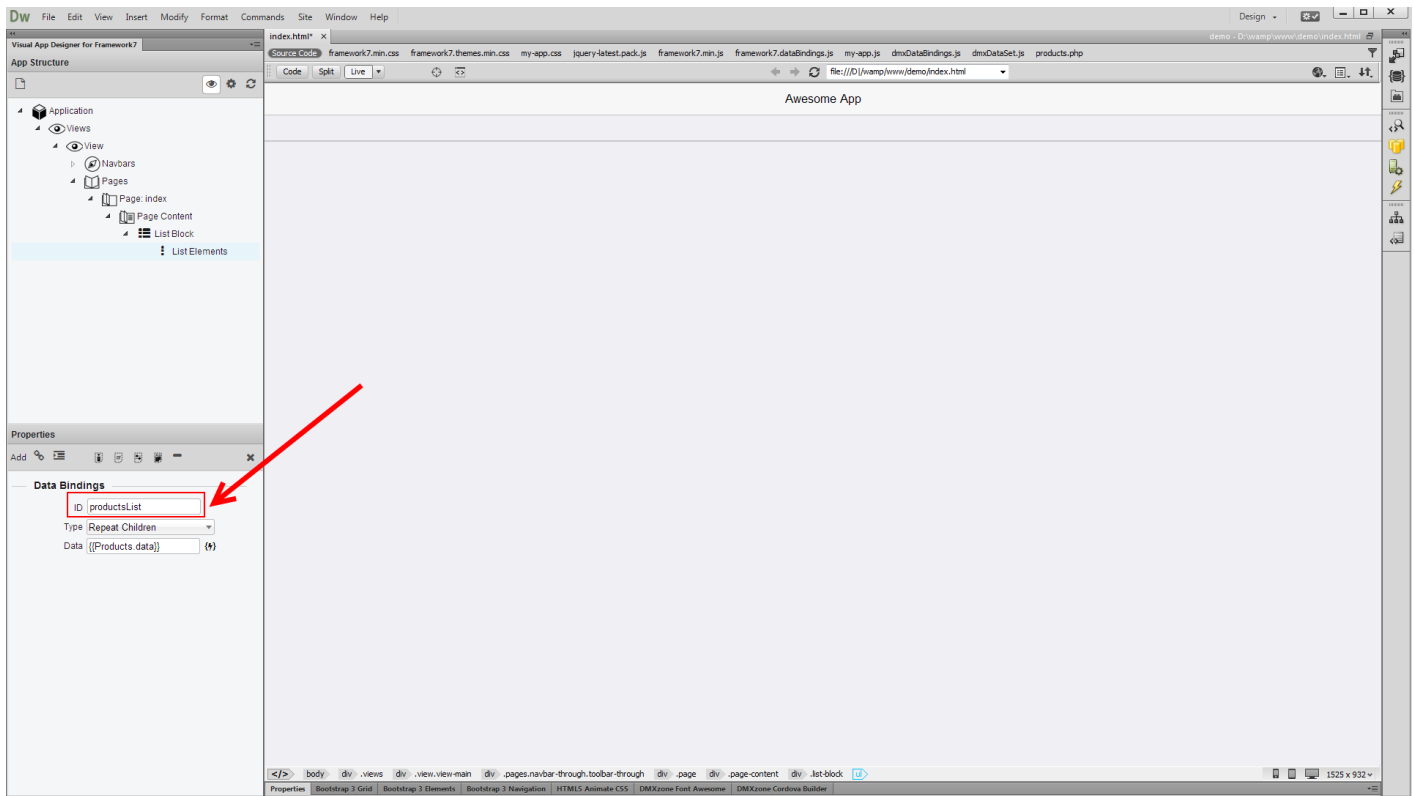
- When you **select list elements (1)** you can see the data bindings options in the properties inspector. Open the **type dropdown (2)** and select **repeat children (3)**.



- Now click the **dynamic data button (1)** and **select the repeating element (2)**. When you're done, click OK.

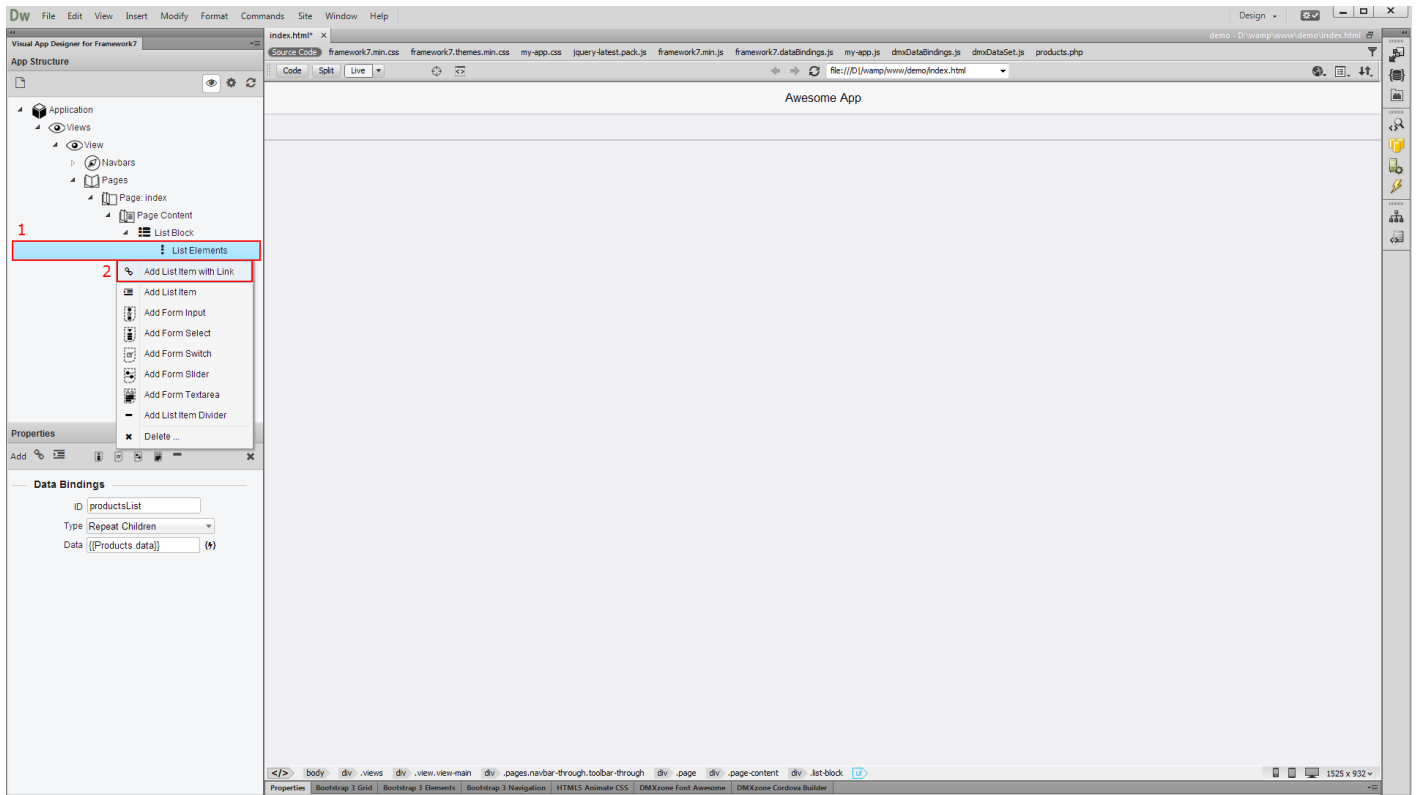


5. Make sure to **enter an ID** for the repeat children region.

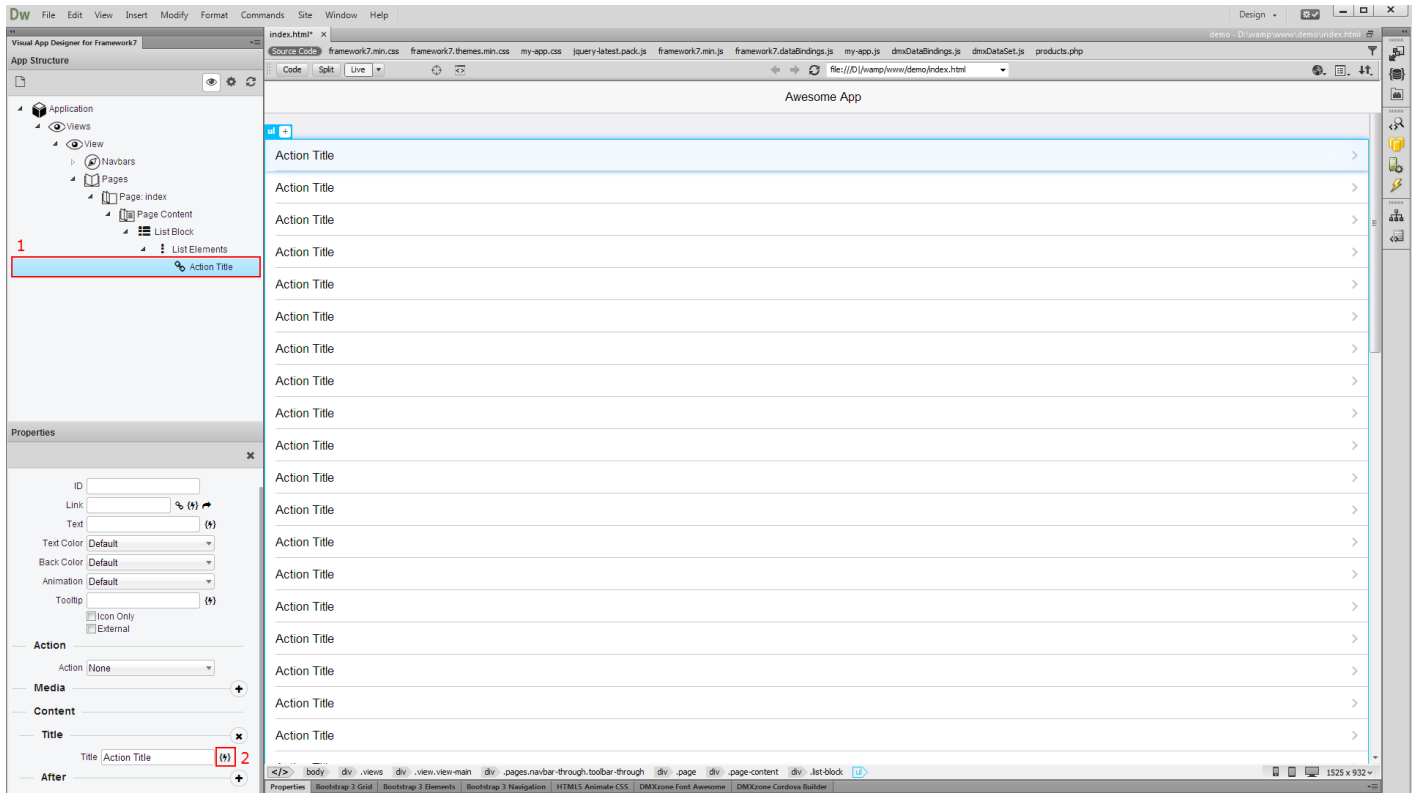




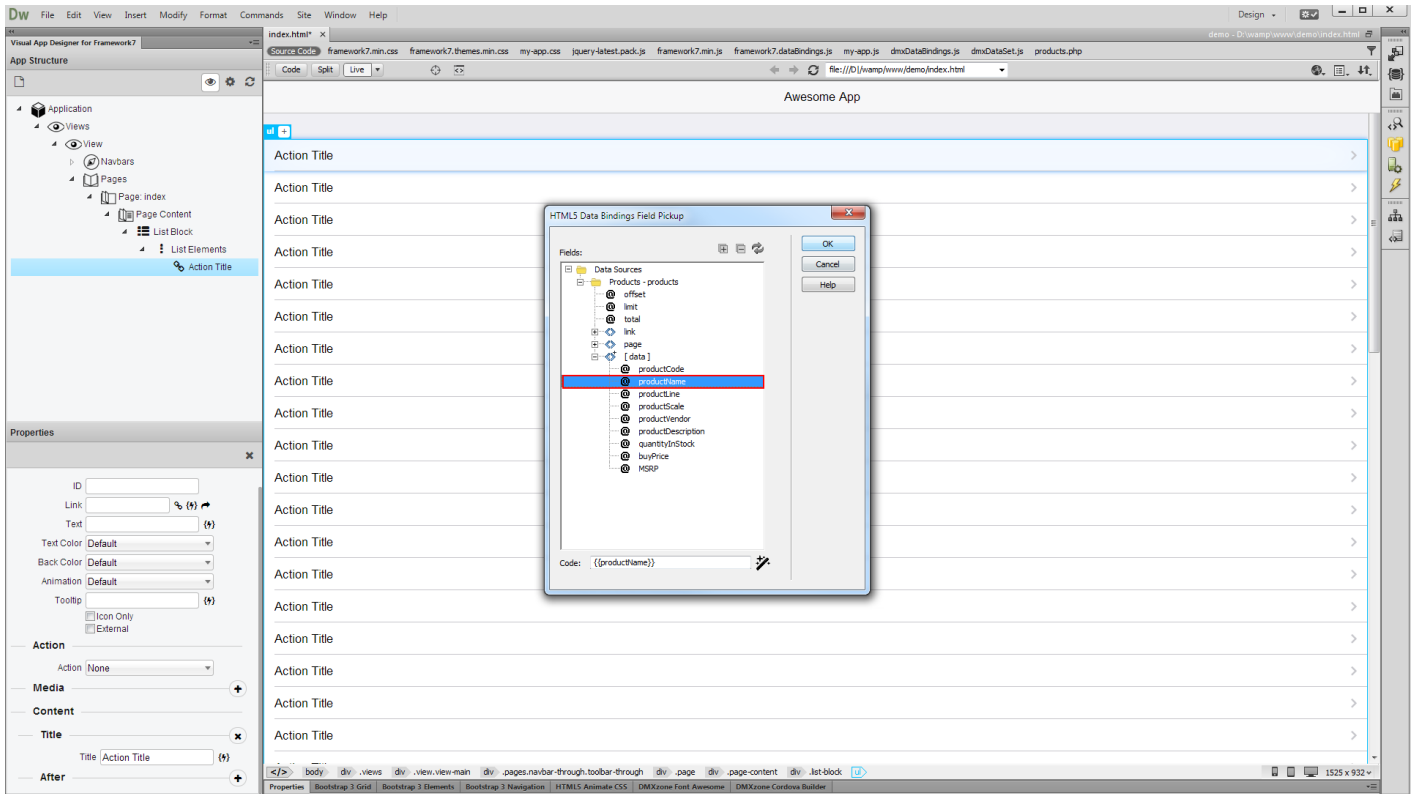
- Right click **the list elements (1)**, when you are done applying the dynamic data to it and **add list item with link (2)**.



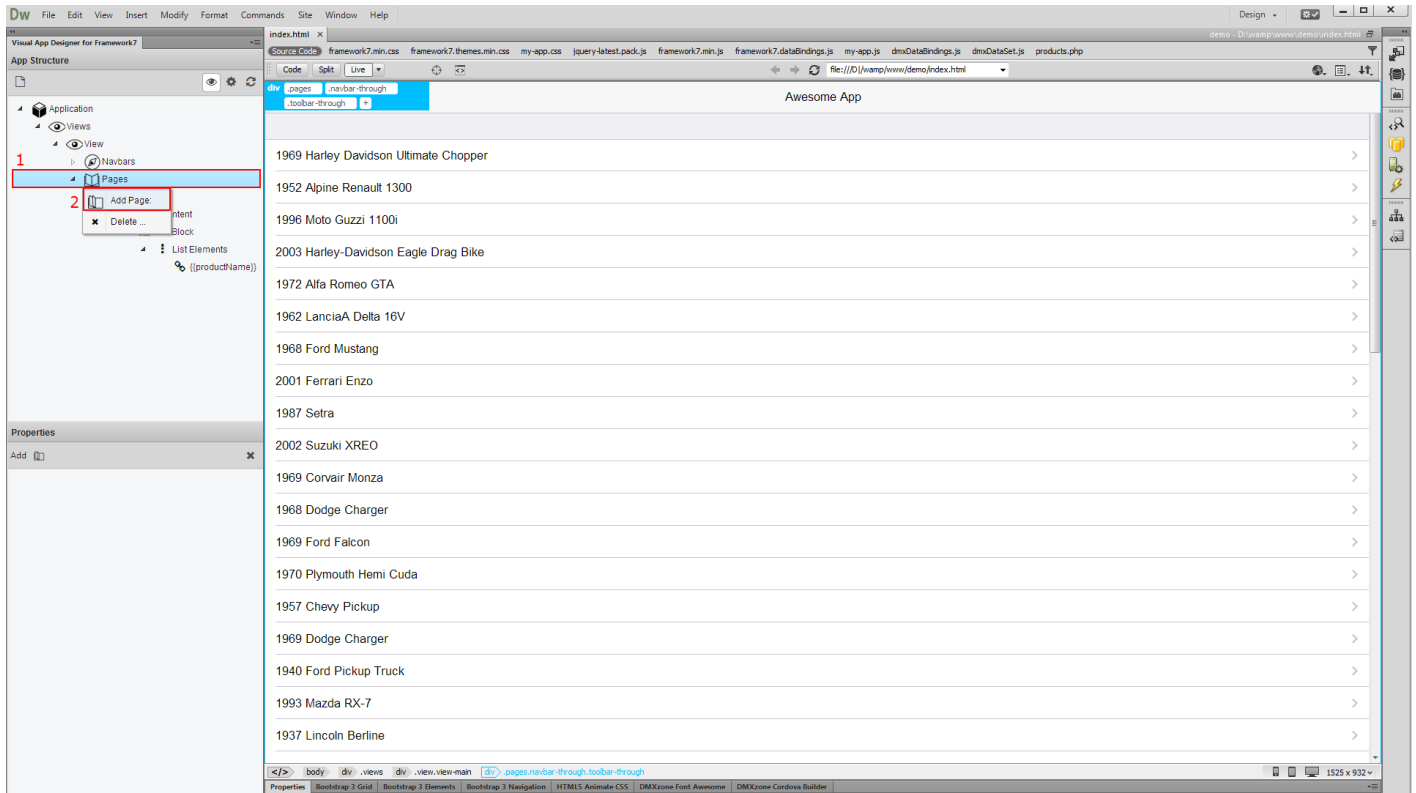
- You can see that the repeat children, repeats our list item but with static data inside. Let's assign dynamic data to it. Click **action title (1)** and select the **dynamic data button (2)** for the list item title inside the properties inspector.



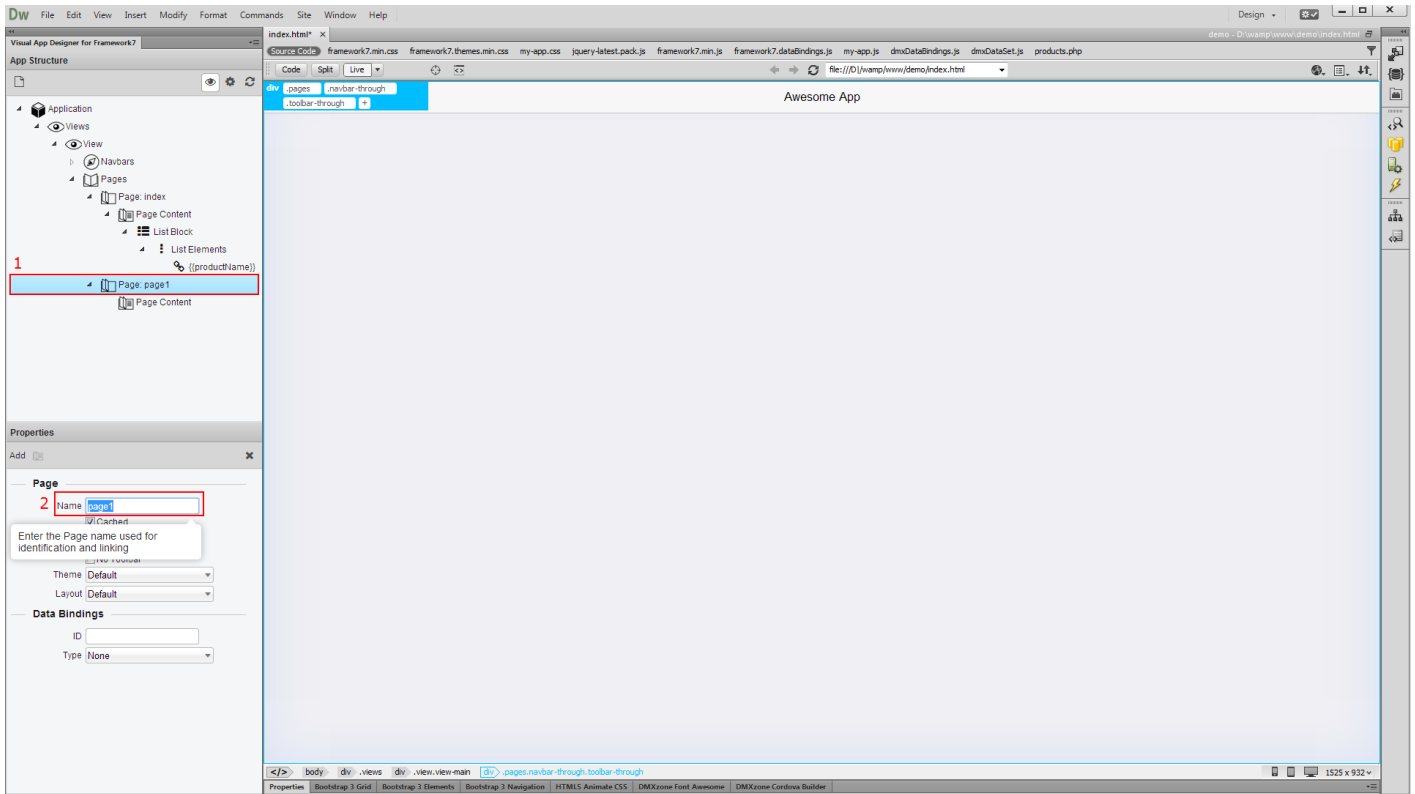
8. Select the data binding that you want to use for the list item title and click OK.



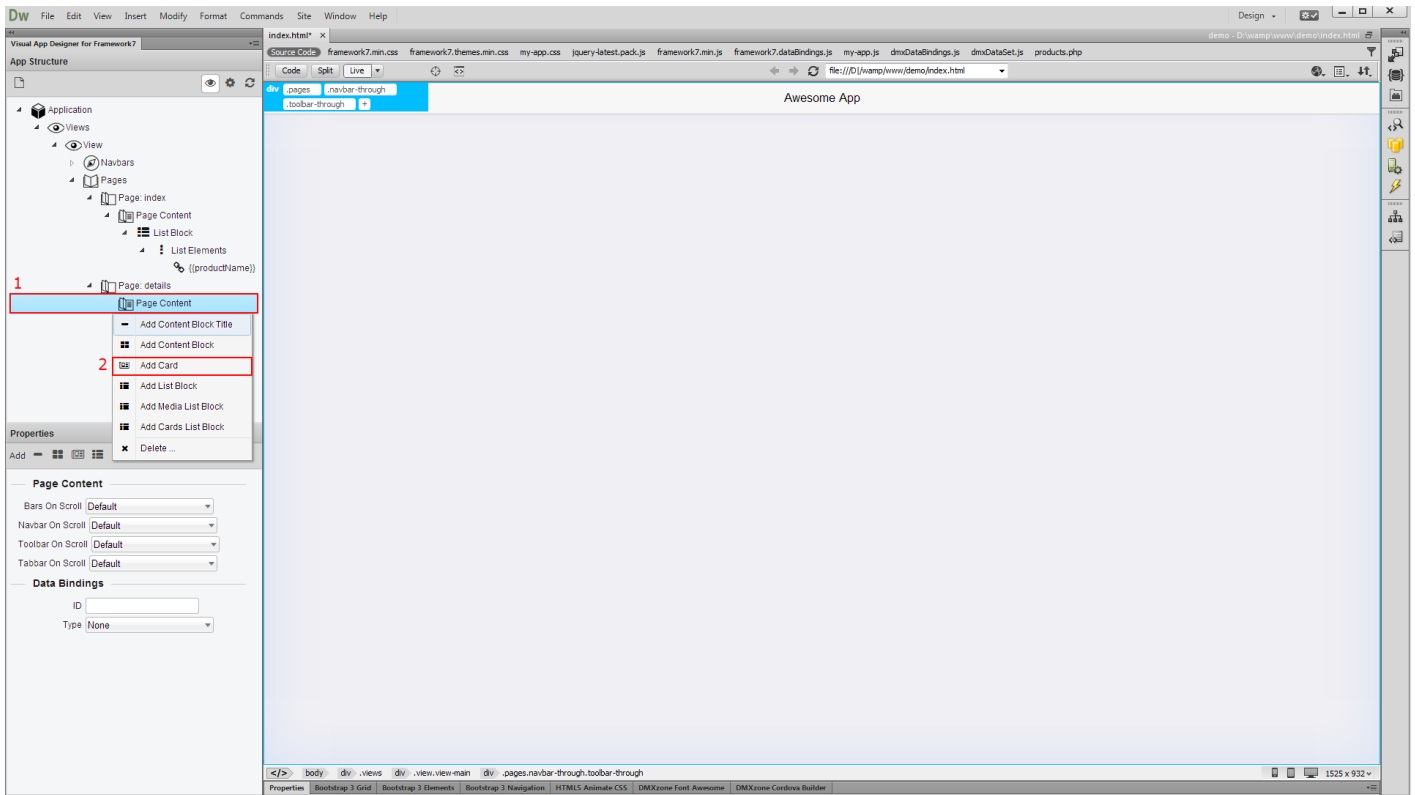
- You can see that our product names are properly displayed in the list now. Let's create the details page, where additional information about the selected product will be displayed. Create a new page by **right clicking pages (1)** and select **add page (2)**.



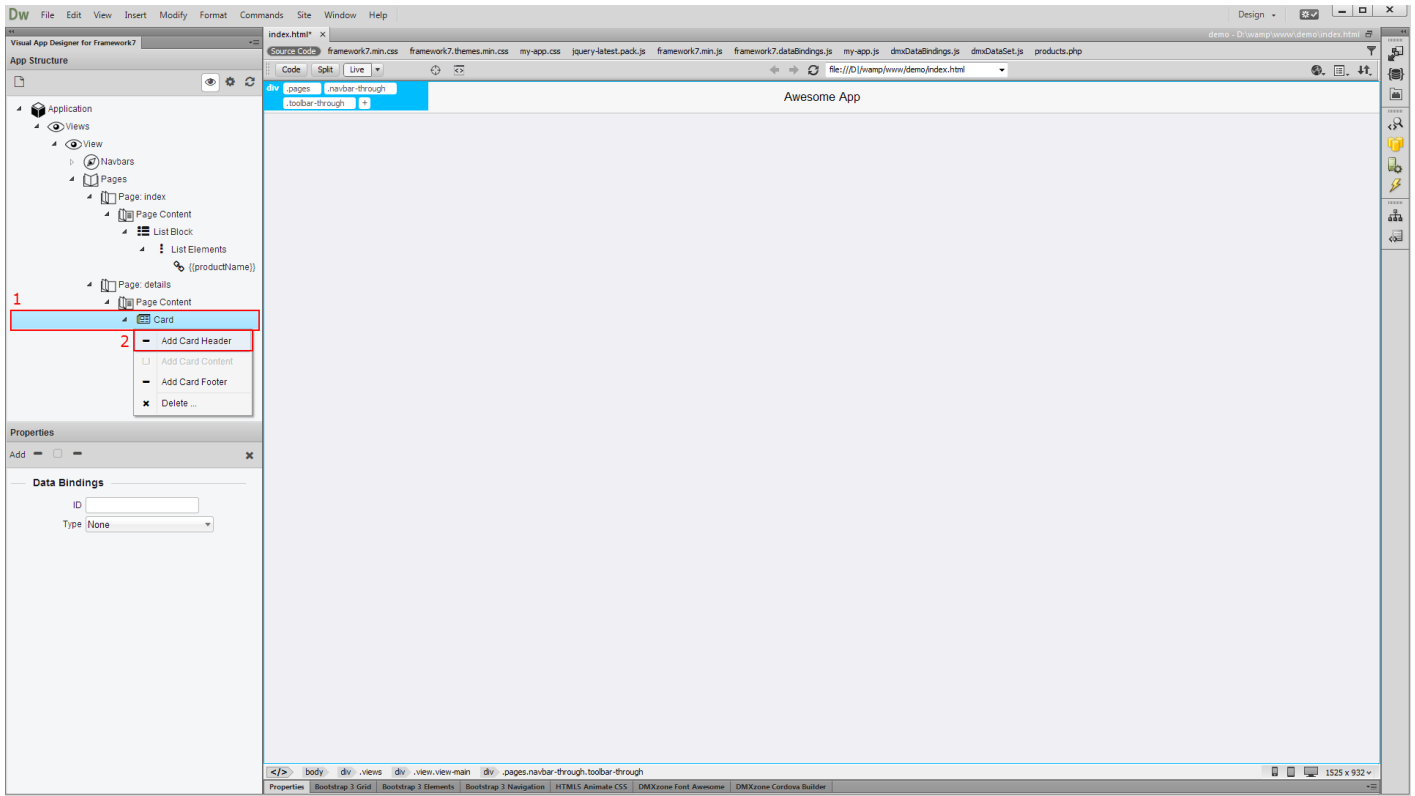
10. Let's change its name to details. **Right click page (1)** and **enter the name in the properties (2)**.



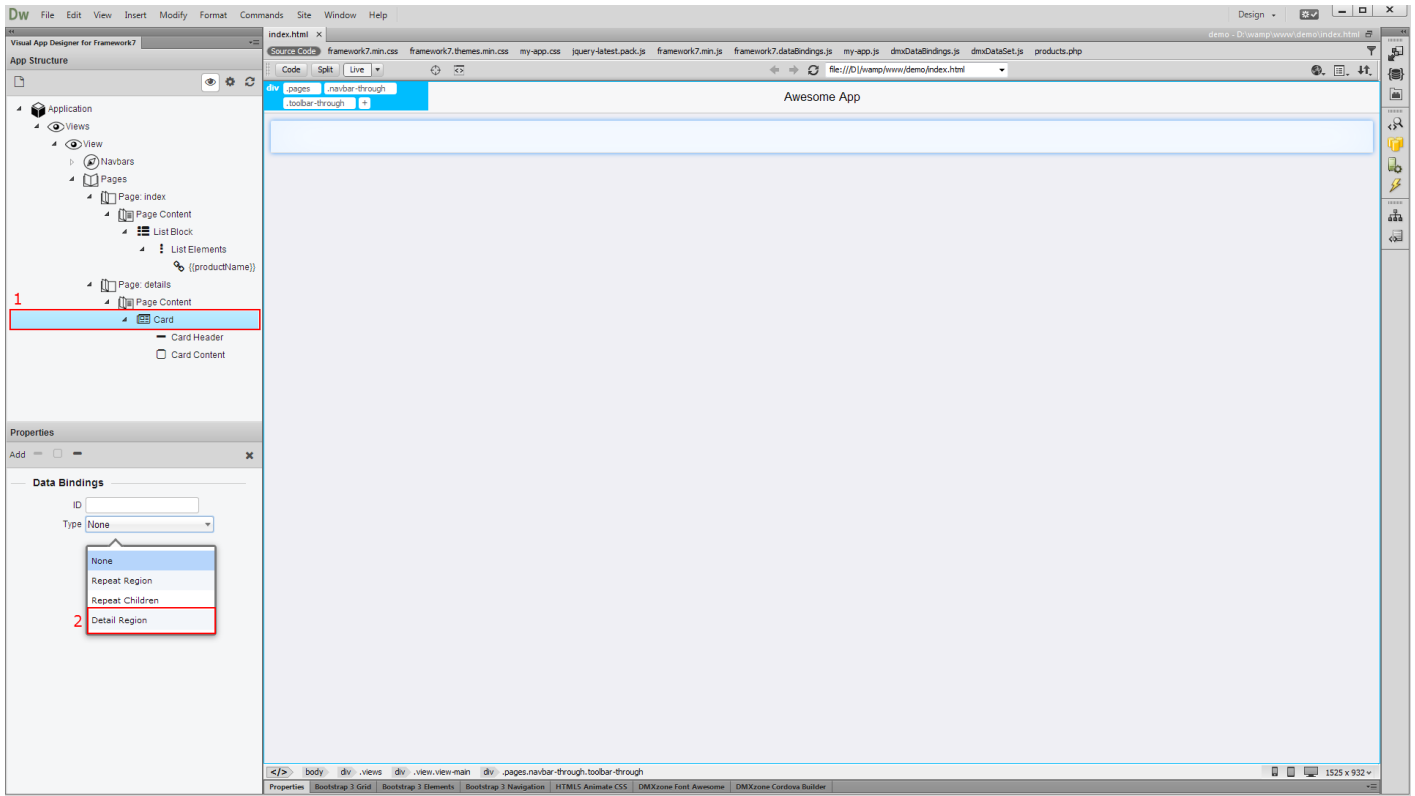
11. We'd like to show the product details into a card with header. Right click the details **page content (1)** and **add a card (2)**.



12. Right click the **card (1)** and **add card header (2)**.

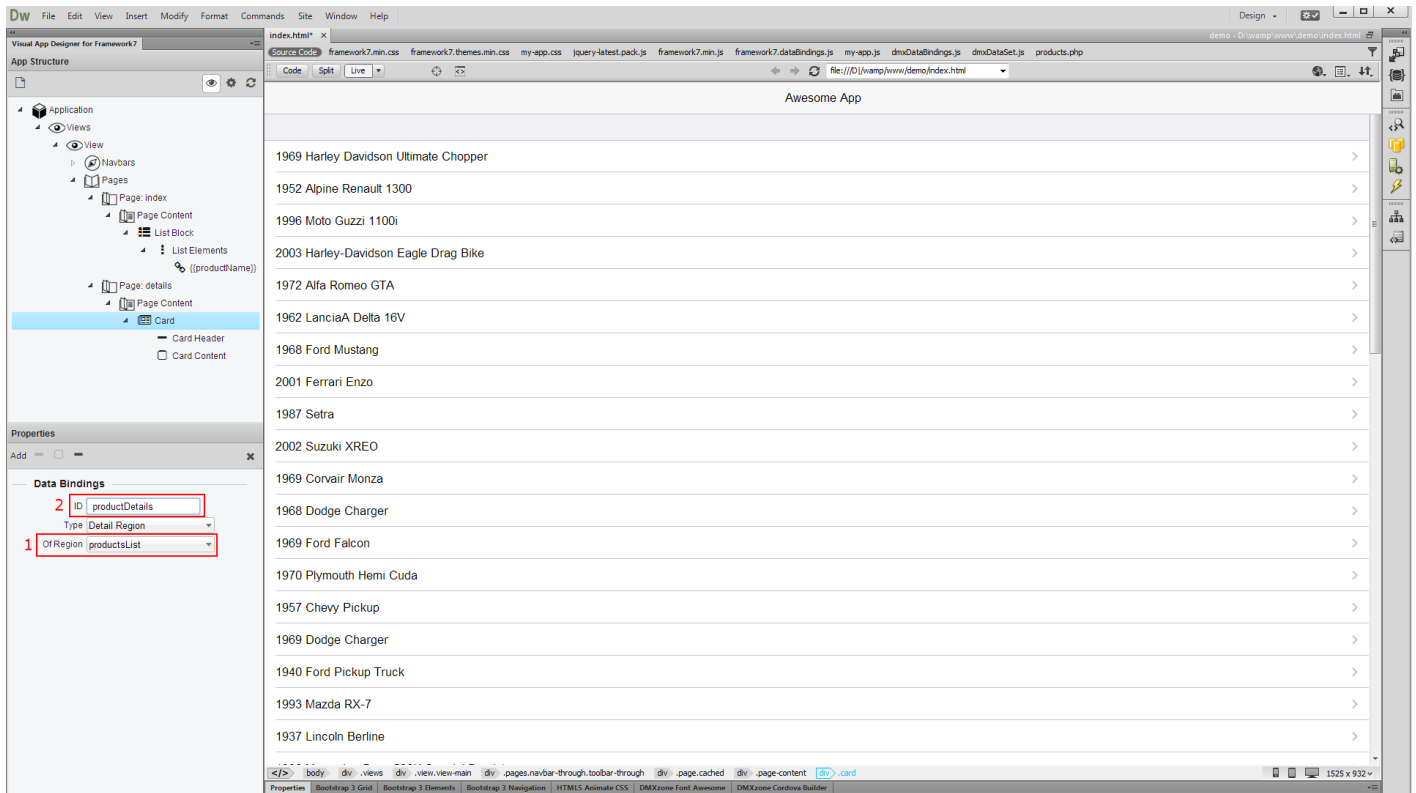


13. Click the **card (1)** and select **detail region** in the type dropdown **(2)**.

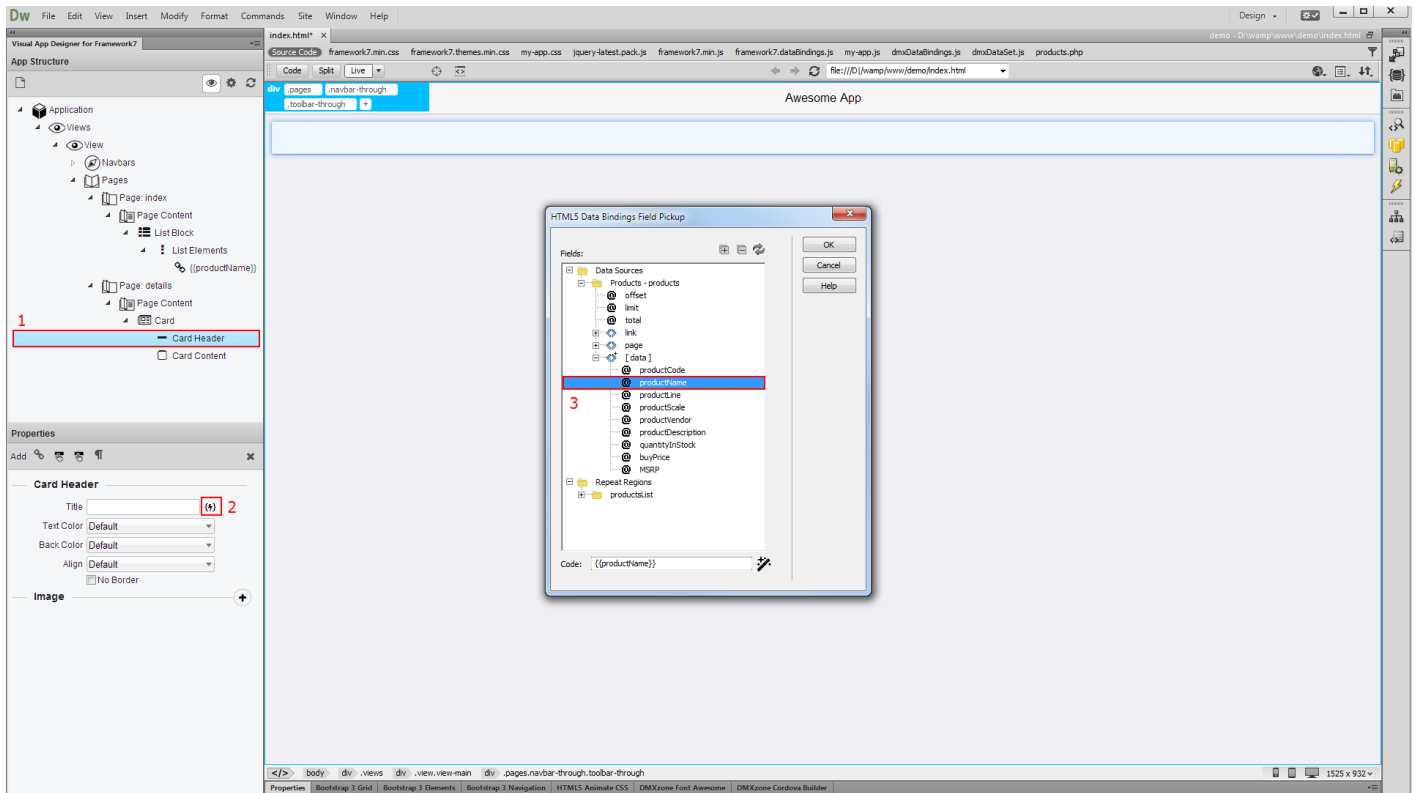




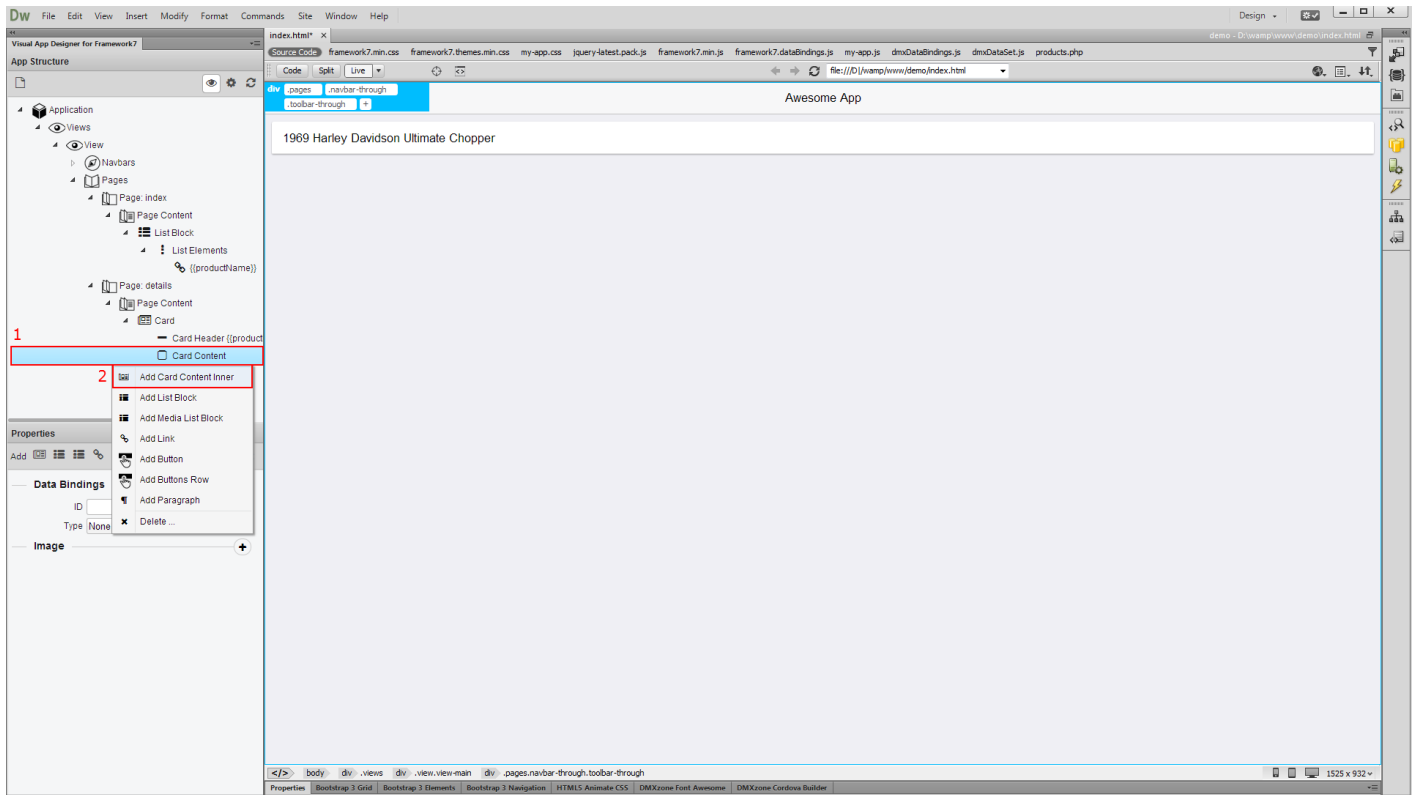
14. Open “of region” and select the **products list repeat region (1)**. Make sure to **enter an ID** for the detail region **(2)**.



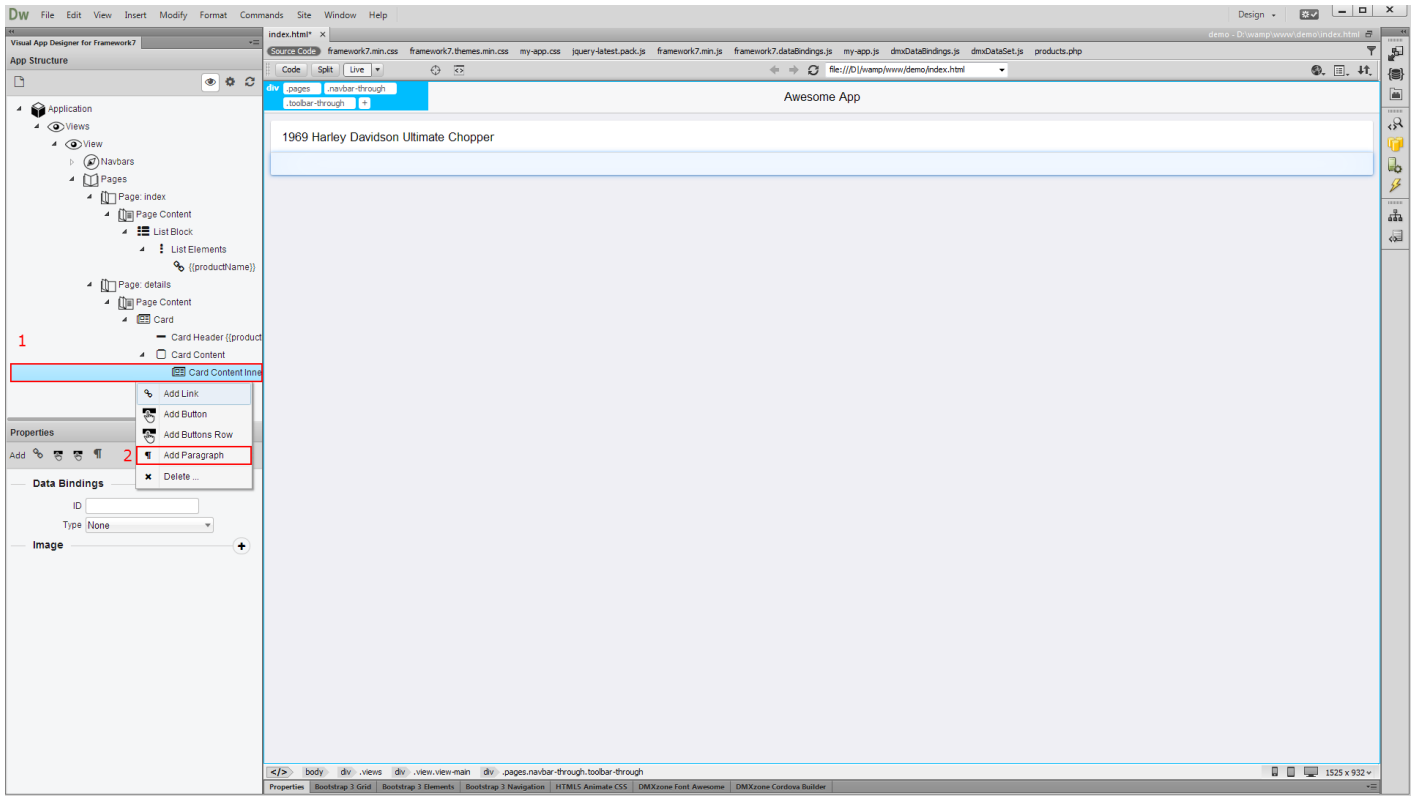
15. Select the **card header (1)** and click the **dynamic data button (2)** for the card header title. We'd like to show the **product name (3)**, inside the card header. When you select it, click ok.



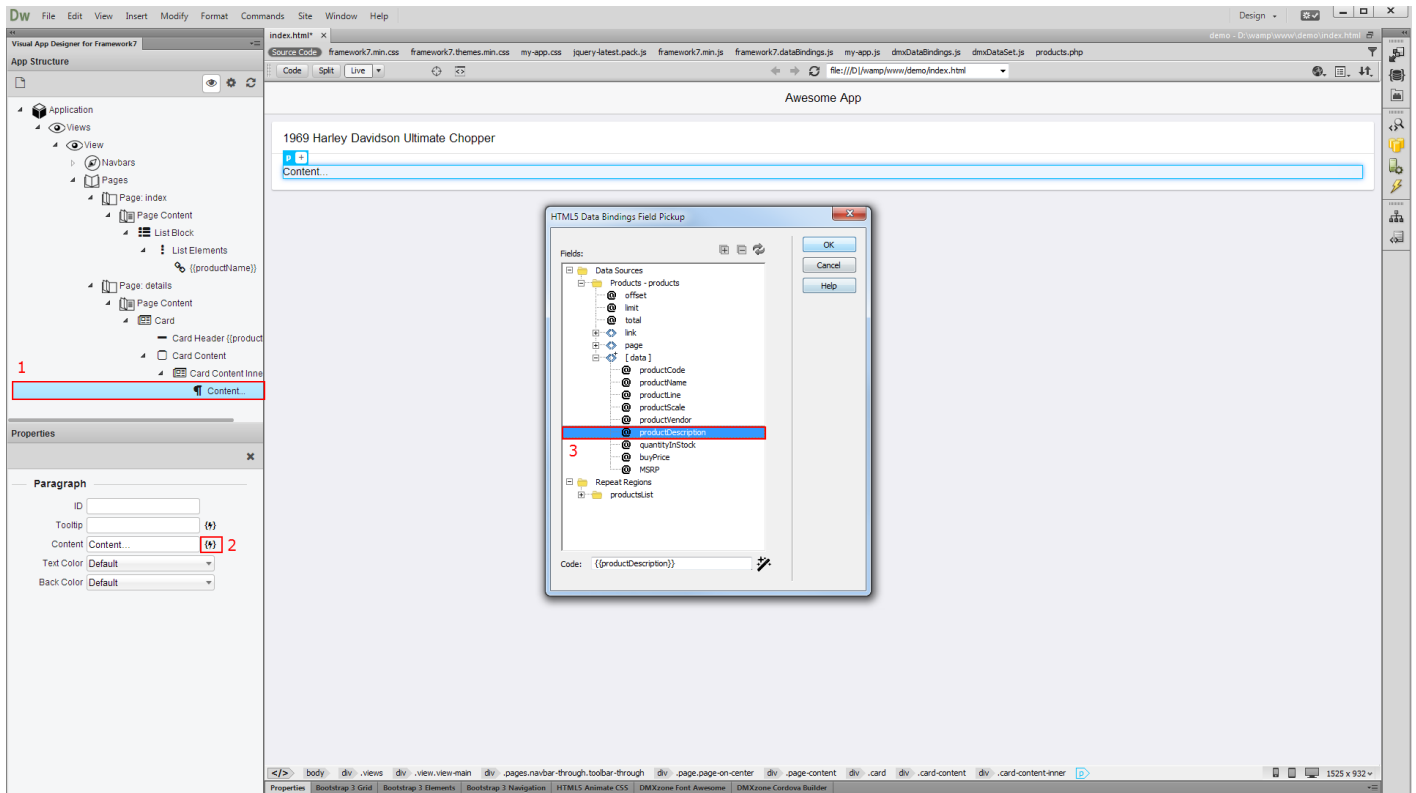
16. Now, let's add some detailed information about the products inside the card content. Right click **card (1)** and **add card content inner (2)**.



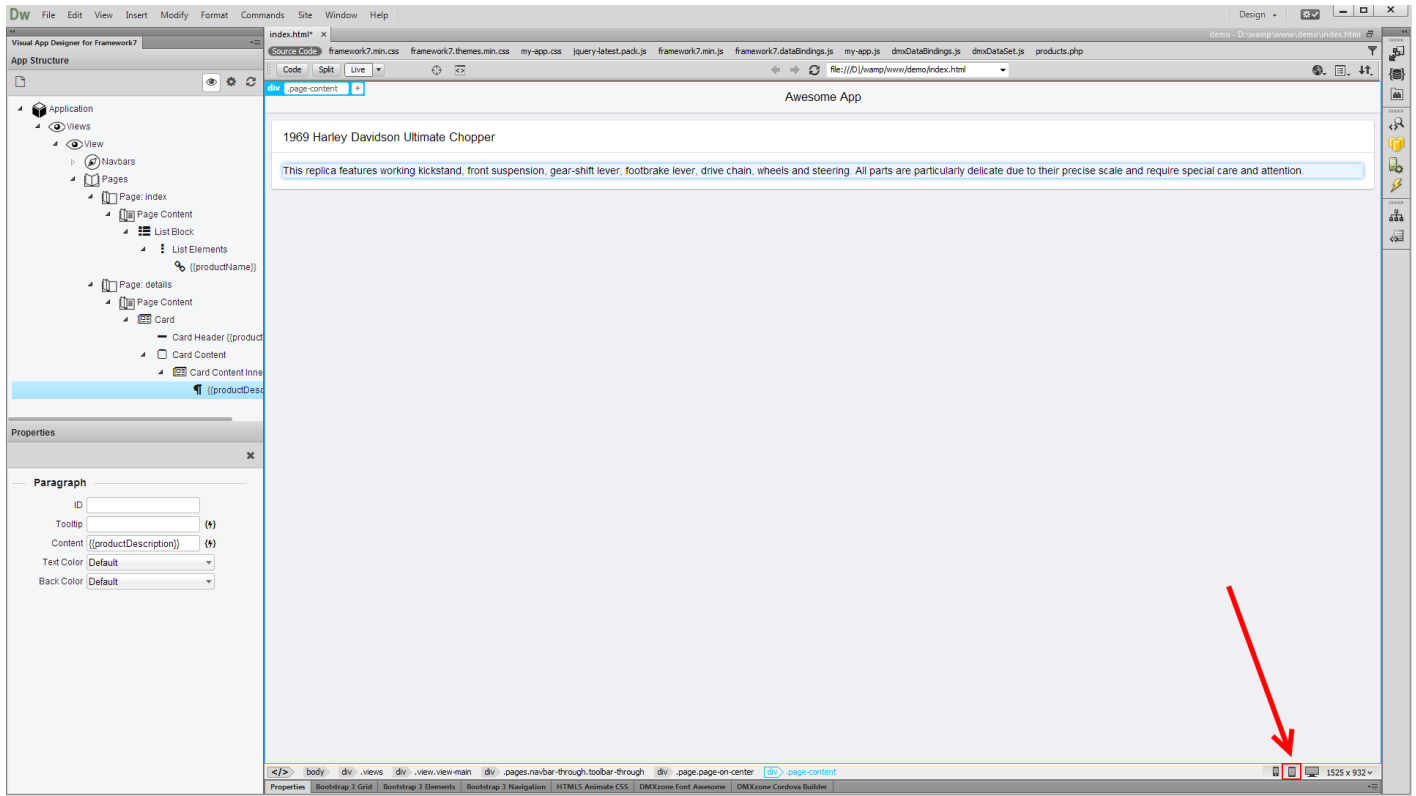
17. Right click the **card content inner (1)** and **add a paragraph (2)**.



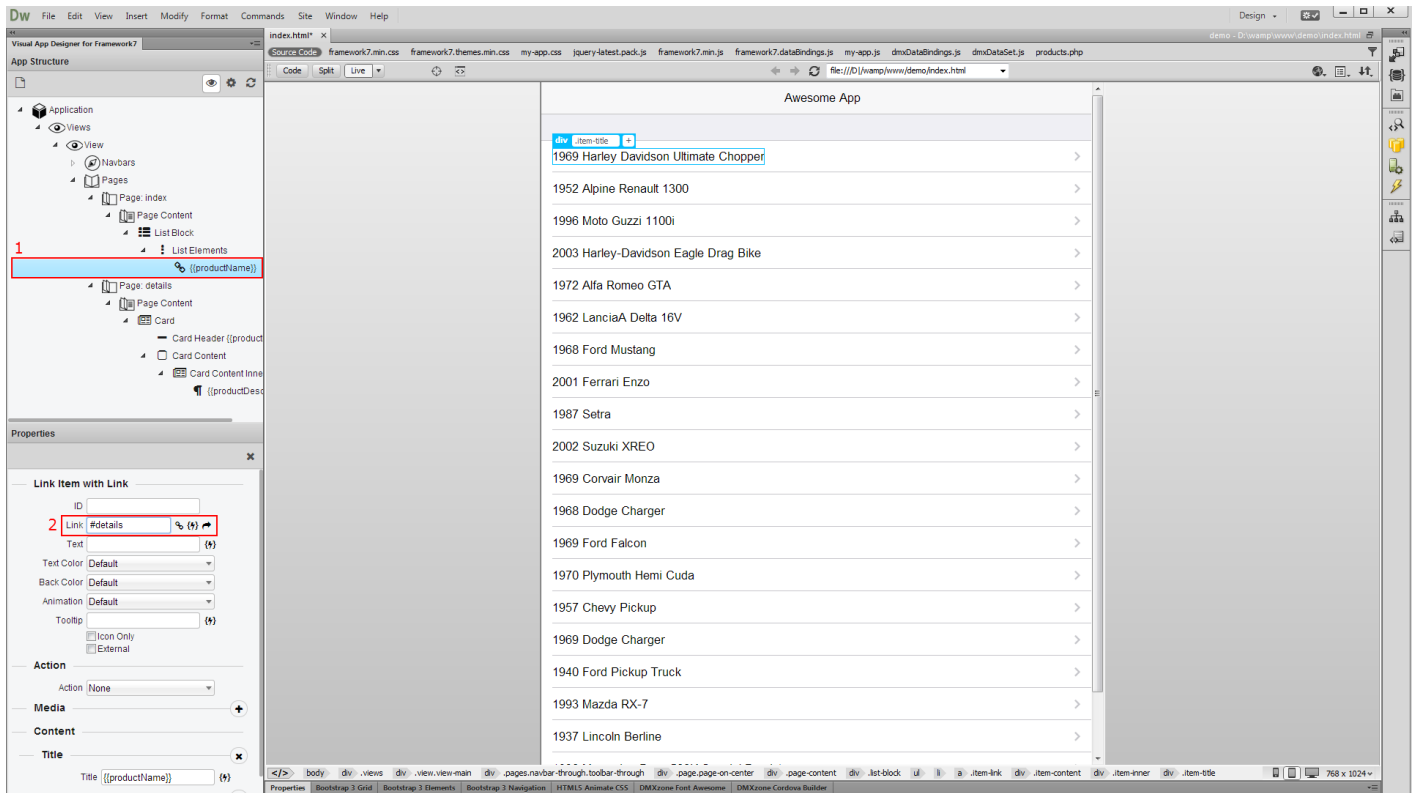
18. Select the **paragraph (1)** that you just added and click the **dynamic data button (2)** for the content. We want to show the **product description (3)** here so we select the item and click OK.



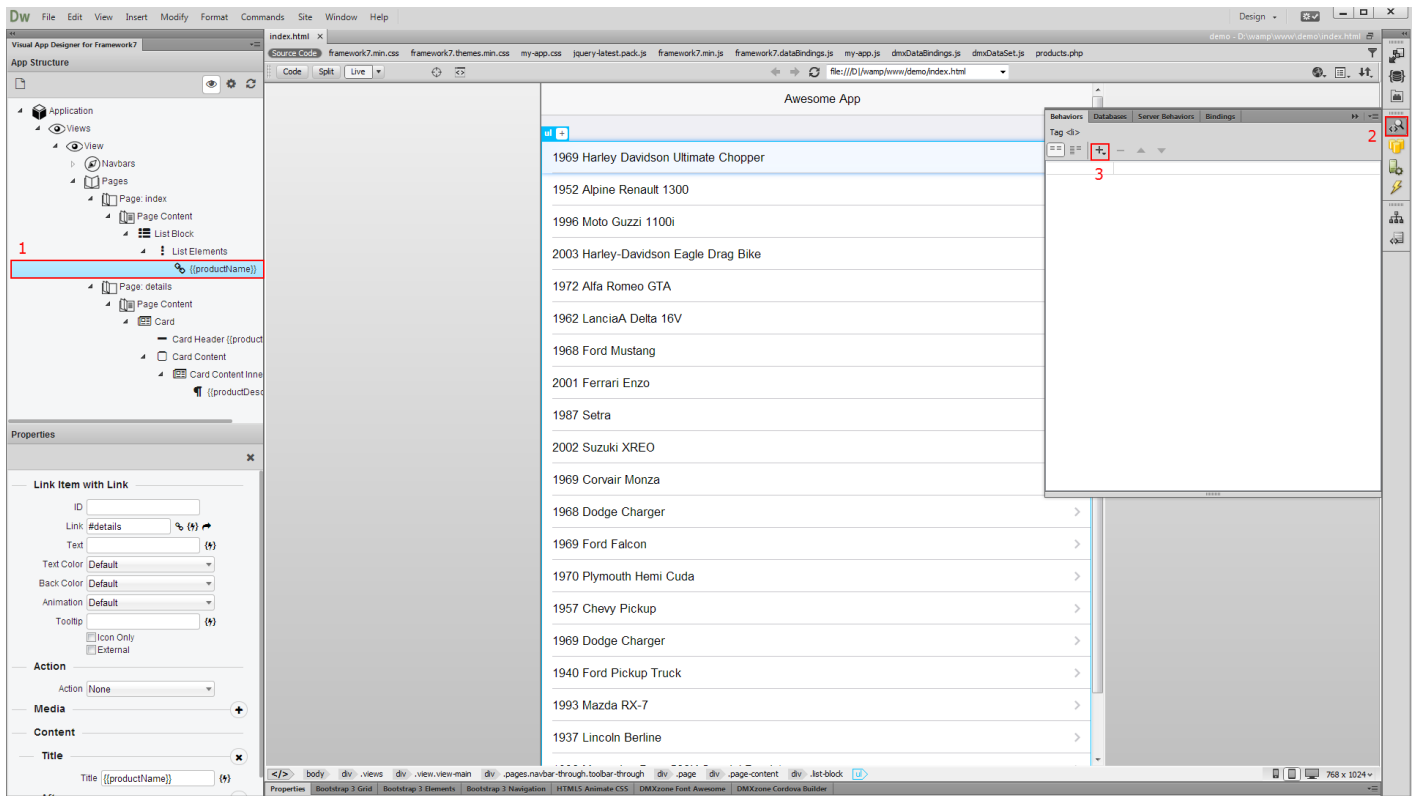
19. Let's switch to tablet view for more realistic preview.



20. Now, let's go back to the index page. Select the **dynamic list item with link (1)**, that we've added previously. We need to link the list item with the details page. Enter: **#details (2)** in the link field.

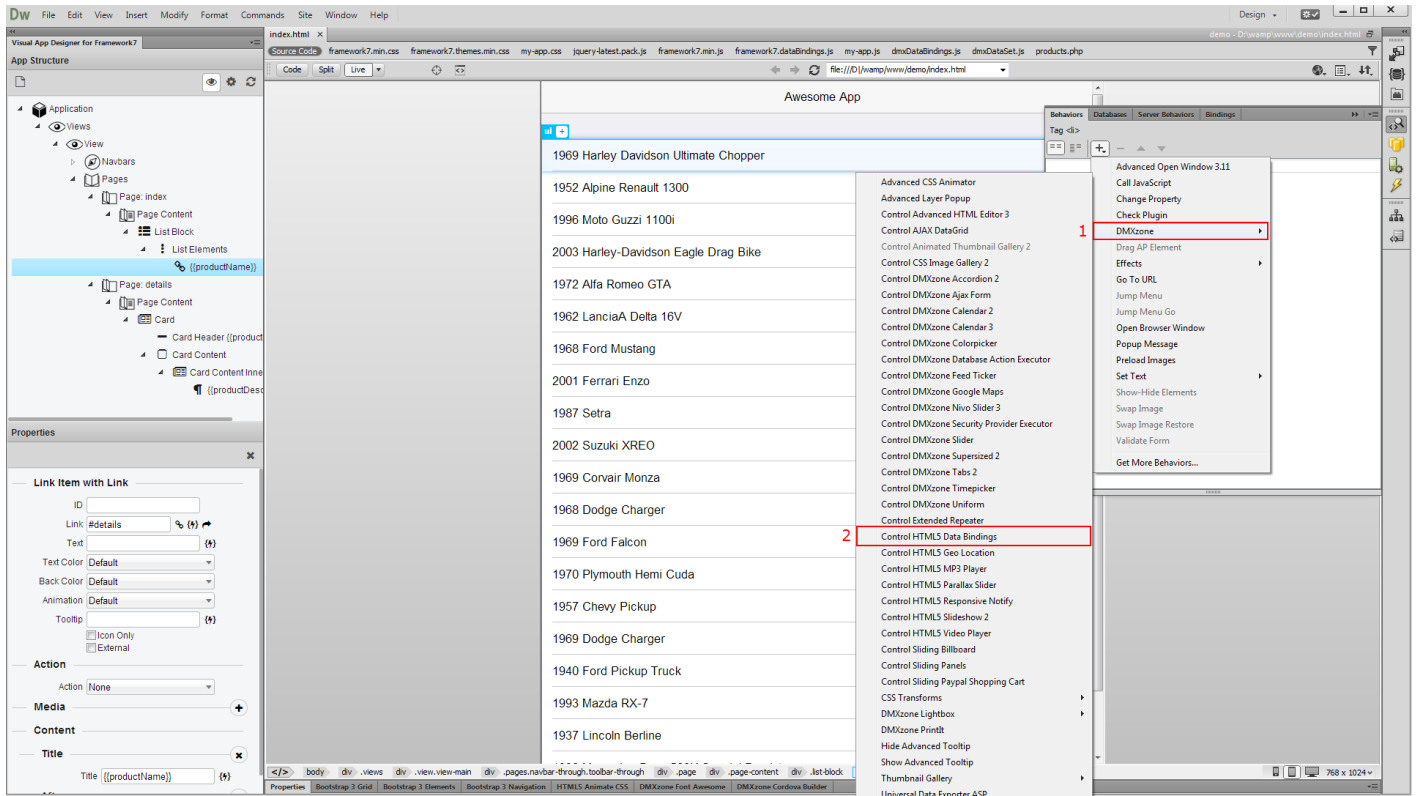


21. Now, select the **list item (1)**, open the **behaviors panel (2)** and **add new behavior (3)**.

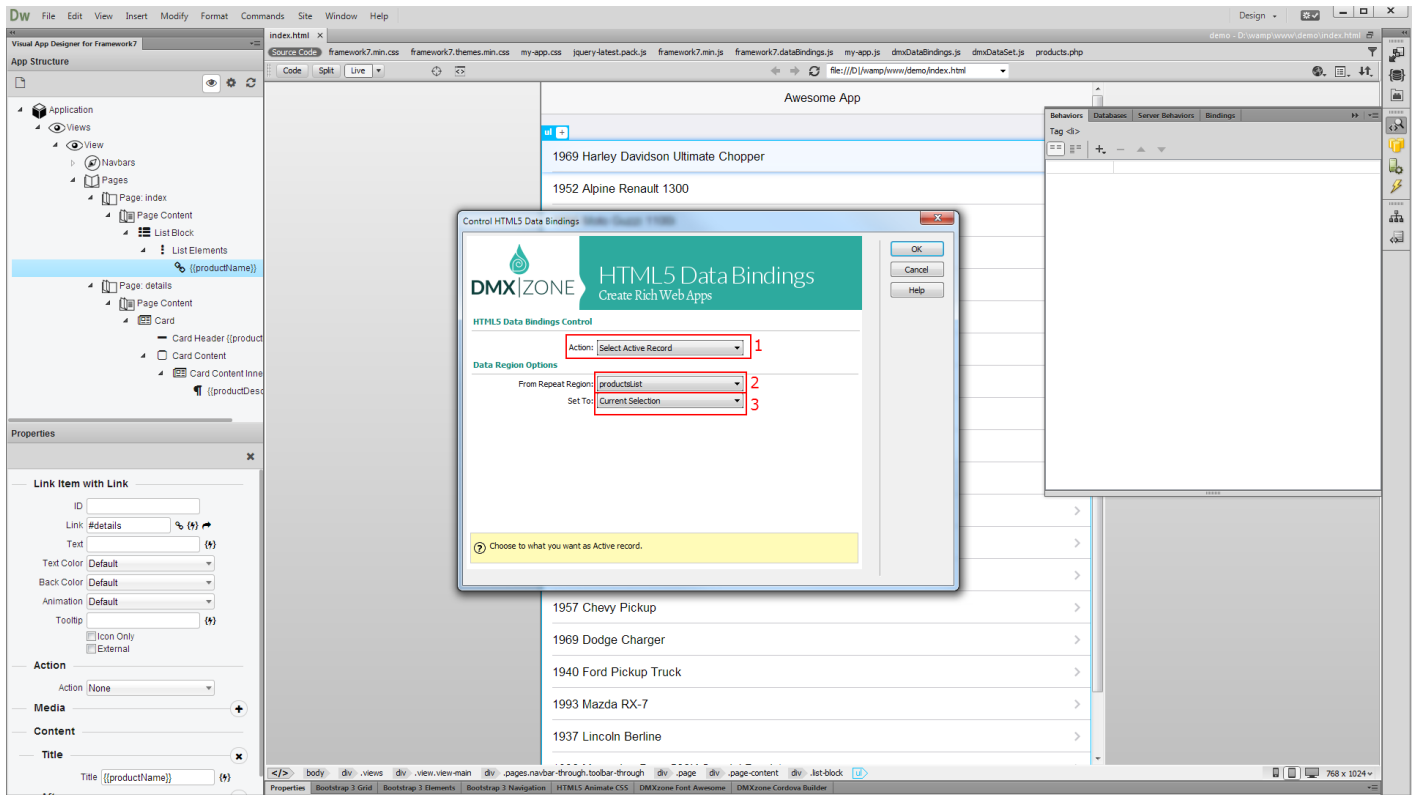




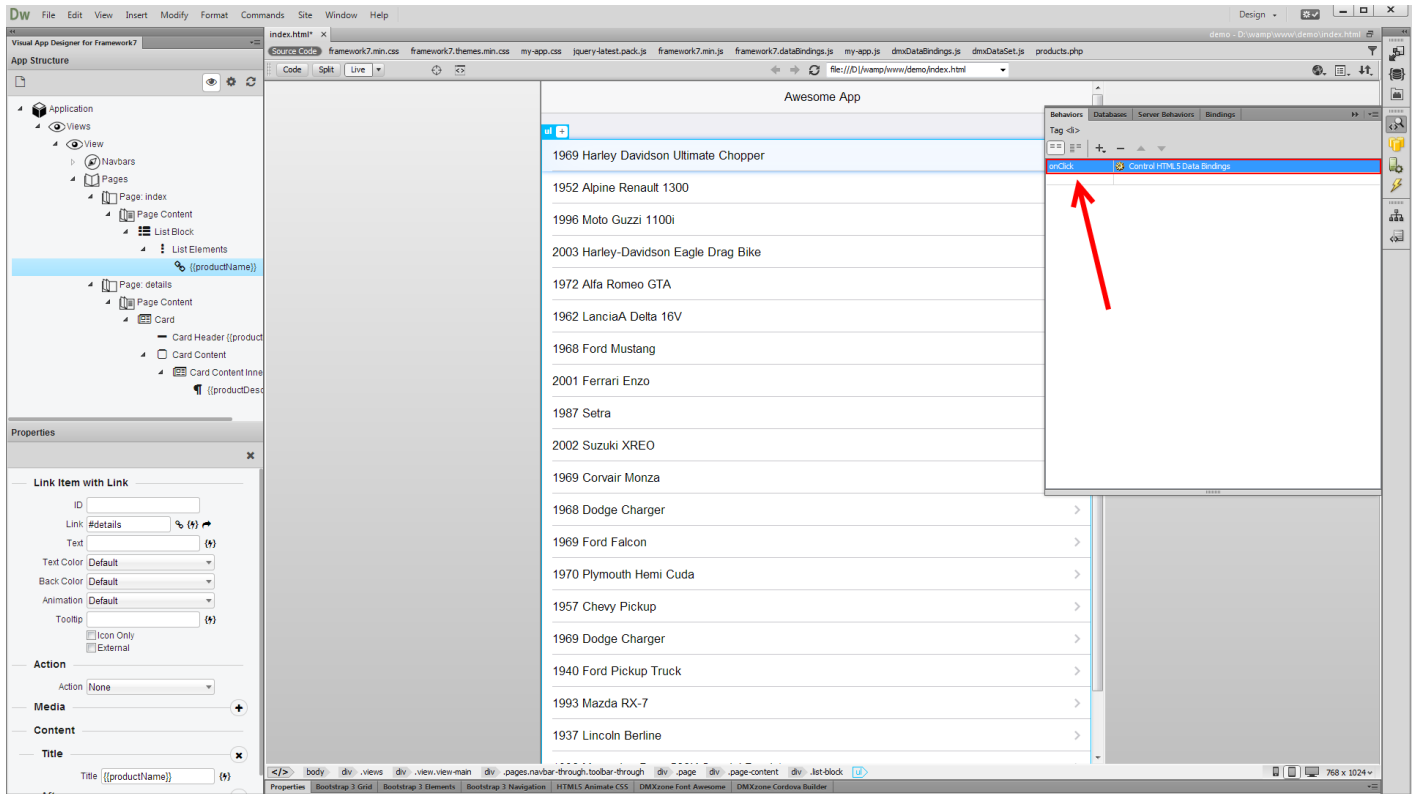
22. From the **DMXzone** menu (1), select **HTML5 Data Bindings** (2).



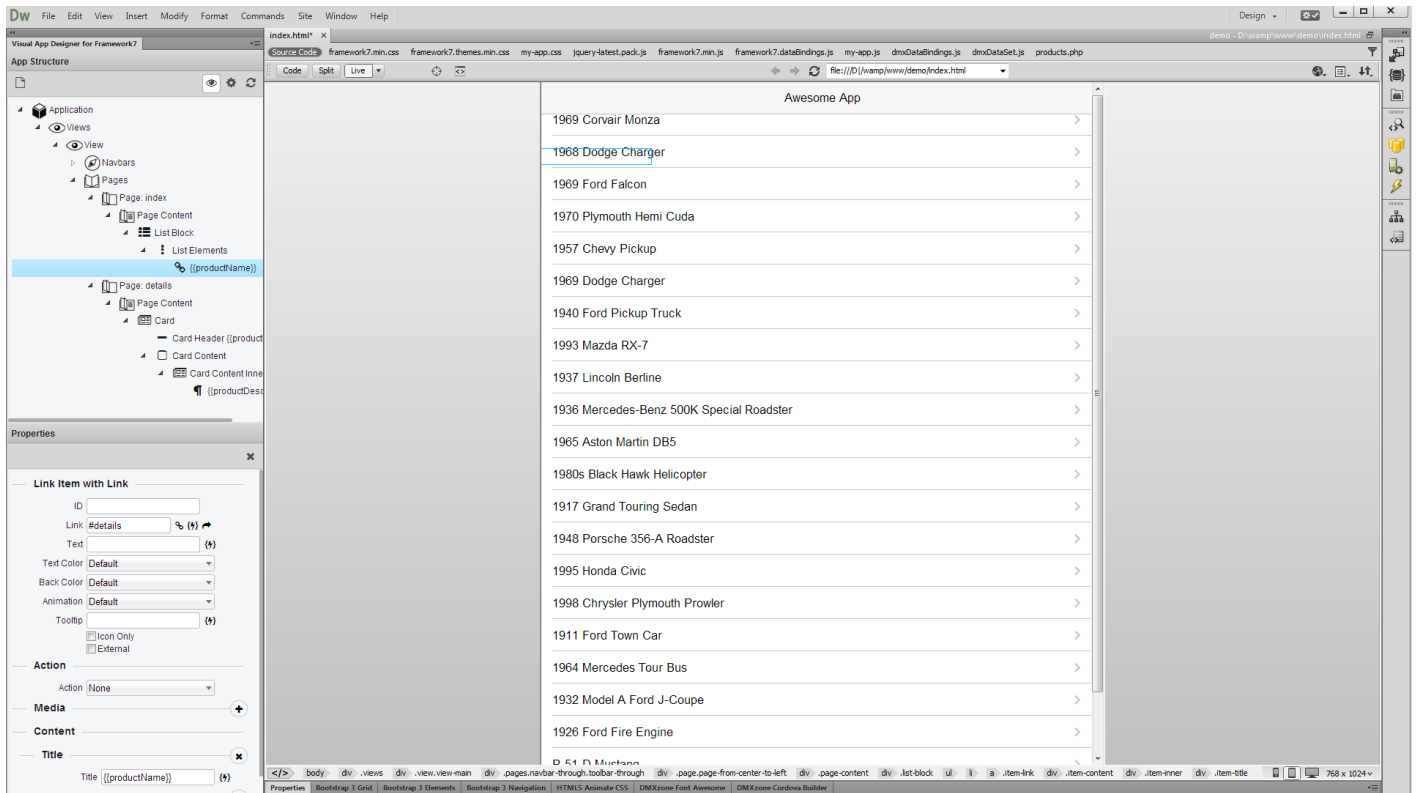
23. From the action dropdown, **select active record option (1)**, choose the **products list repeat region (2)** and select **current selection (3)** in the set to menu. Click OK when you're done.



24. Make sure to set this behavior on click.



25. Let's preview the results, directly in Dreamweaver. That's how easy it is to build an app, containing database driven lists, and detail pages.



## Video: Setting Up Visual App Designer



With this [movie](#) we will show you how to setup [DMXzone Visual App Designer for Framework 7](#) and how to insert the default app layout on your page.

## Video: Manage Pages and Content of your App



With this [movie](#) we will show you how to how to manage the pages of your app, using [DMXzone Visual App Designer for Framework 7](#).

## Video: Using Navbars and Toolbar in Visual App Designer



With this [movie](#) we will show you how to use Framework 7 nav bars and toolbar in your app.

## Video: Insert List Blocks and List Items in Visual App Designer



With this [movie](#) we will show you how to insert list blocks and list items into your app layout, using [DMXzone Visual App Designer for Framework 7](#).

## Video: Insert Cards with Header, Footer and Content



With this [movie](#) we will show you how to use cards with [DMXzone Visual App Designer for Framework 7](#).

## Video: Using Media Lists in DMXzone Visual App Designer



With this [movie](#) we will show you how to insert media list blocks with [DMXzone Visual App Designer for Framework 7](#).

## Video: Building Database Driven Mobile Apps



With this [movie](#) we will show you how to build dynamic, database driven app layout, using [DMXzone Visual App Designer for Framework 7](#).