Bumblebee 2.0.3-alpha Change Log

The following are known changes for Bumblebee v2.

(NOTE: The sentences marked in bold are the breaking changes.)

- GetElement/GetElements
  - **These methods are obsolete now to remove any confusion, since the FindElement/FindElements do the same thing and are available as well.**

- Pages
  - Introduced as part of v2
  - **All top-level blocks should inherit from Page or WebPage depending on whether or not you want to have explicit Wait functionality.**
  - A type derived from Page should only require the Session and assumes that the Tag will be scope to the <body> tag.
  - The Tag property is lazy loaded in order to make sure that no elements grow stale from DOM changes.
  - **The Session.NavigateTo<T> is now limited to only those types that are derived from Page**
  - The Session has a new method, CurrentPage<T>() that returns a page representation based on the type of T which is constrained to only types that are derived from Page.

- Blocks
  - All child scopes within a page should be represented by a type that derives from Block or WebBlock depending on whether or not you want to have explicit Wait functionality.
  - **A type derived from Block should only require a parent IBlock and a By specification.**
  - The Tag property is now lazy loaded in order to make sure that no elements grow stale from DOM changes.

- SpecificBlock
  - **In the new version of the framework, since blocks are lazily loaded, the SpecificBlock class has been made obsolete.** In the previous version of the framework, the SpecificBlock allowed for the user to pass the parent IBlock and an IWebElement tag. This was primarily used for situations in which a FindElements() returned multiple elements and each had to be assigned a tag in the list.
  - A new class called Blocks<T> was added which will accept the parent IBlock and a By specification and return an IEnumerable<T> which encapsulates a lazy loading sequence.
  - A new helper method, FindBlocks<TBlock> has been added to Block that returns an IEnumerable<TBlock>.

- Elements
  - A new class called Elements was added which will accept the parent IBlock and a By specification and return an IEnumerable<IElement>.
  - A new helper method, FindElements<TElement> has been added to Block that returns an IEnumerable<TElement>.

- By Specifications
  - We have added some specifications to the framework. One is the By.JQuery() which accepts a jQuery selector to find a single element or multiple. Another is the By.Attribute() which accepts an attribute name and value and returns an element/elements that match.
  - We have added the ability for developers to create their own specifications. All that needs to be done is to create a an extension method on ISpecification.
  - In the future, we will use this new feature to introduce other common ways for discovering elements such as for the Protractor framework.

- Session
  - We have added a HasJQuery() method to determine if JQuery is installed on a page.
  - The CurrentBlock<T>() has been enhanced to handle the last block that was instantiated during the session, however, now it also handles finding any related Blocks, if requested.
  - Even though CurrentBlock<T>() supports pages since pages are derived from Block, for the sake of clarity, we have also added a CurrentPage<T>() method that only works with types that derive from the Page class.

- InlineFrame
  - Added InlineFrame which is derived from Block and allows for navigating through the ancestral tree that contains frames.