

# Project WONDER-Frameworks-ERXNavigation

- [Quick Getting Started Placeholder](#)
  - [1. Simple example](#)
  - [2. Using Conditionally Displayed Tabs](#)

## Quick Getting Started Placeholder

### 1. Simple example

To use ERXNavigation you must initialize the ERXNavigationManager in a method such as finishInitialization in Application.java. This requires that a NavigationMenu.plist file already be defined and placed in your project's Resources folder. The Root node's children will be the level 1 navigation items. The Root node can be variable depending on code if you like, but it is not described below. The BugTracker application uses a variable Root (and therefore a different menu structure visible) depending on the type of user that is logged in, so you can refer to that until I get the chance to write it up here. For each of your Children you can define further children up to 3 levels deep. ERXModernNavigationMenu does not have the hierarchical depth limits, but it is not covered here yet. For each menu item you'll want to define an action. I have used Session as my storage for actions, but you may choose to use a navigation controller class for this. Each component must tell the NavigationManager its state in its awake() method. I think that covers the basics with examples below. This document is not terribly complete so don't hesitate to contact me if you have questions in the meantime.

The Wonder API documentation says "Please read "Documentation/Navigation.html" to find out how to use the navigation components." If you download the source distribution of Wonder, the full path from the top of the source archive is Wonder-Source/Frameworks/Core/ERExtensions/Documentation/Navigation.html.

#### Application.java

```
// initialize the navigation

public void finishInitialization() {
    super.finishInitialization();
    ERXNavigationManager.manager().configureNavigation();
    NSLog.debug.appendln("finishInitialization called.");
}
```

Example actions in Session:

#### Session.java

```
// Don't forget to set the navigation state in your component.java (see below)

public WOActionResults toUserPage() {
    return context().page().pageWithName(UserPage.class.getName());
}
```

Setting the navigation state in the component

#### UserPage.java

```
// Method used to set navigation state for use with the Navigation Menu components
// State is pushed from the page level component so that you don't have problems opening
// multiple tabs in modern browsers
// The code below tells the context that the selected tabs are "my profile" on level 1 and "summary" on
level 2
// Add similar code to each of your page level components
public void awake() {
    ERXNavigationManager.manager().navigationStateForSession(session()).setState(new NSArray(new
Object[] {"my profile", "summmmary"}));
}
```

Example NavigationMenu.plist

## NavigationMenu.plist

```
// there are more options than I am using below. See the docs for more examples
// added an example for using "href"
// updated an item to show use of displayName
(
    {
        name = "Root";
        children = ("my profile", "my survey", "Apple");
    },
    {
        name = "my profile";
        children = ("summary", "details");
        action = session.toUserPage;
    },
    {
        name = "summary";
        action = session.toUserPage;
    },
    {
        name = "details";
        displayName = "my personal details";
        action = session.toUserDetails;
    },
    {
        name = "my survey";
        children = ("instructions", "section I");
        action = session.toSurvey;
    },
    {
        name = "instructions";
        action = session.toSurvey;
    },
    {
        name = "section I";
        action = session.toSurvey1;
    },
    {
        name = "Apple";
        href = "http://www.apple.com";
    },
)
```

HTML Source you'll see generated by ERXNavigationMenu when the app is run

## Main.html

```
<div class="ERXNavigationMenu">
  <ul class="Level1Items">
<li nowrap="nowrap" class="Nav1" id="id1"><a href="/cgi-bin/WebObjects/MyApp.woa/wo/49YrOFk9VT0wc2WYDlZ8b0/5.0.11.0.0.1.1.2.1.0.0.1.1.0.0">my profile</a></li>

<li nowrap="nowrap" class="Nav1Selected" id="id2"><a href="/cgi-bin/WebObjects/MyApp.woa/wo/49YrOFk9VT0wc2WYDlZ8b0/5.0.11.0.0.1.1.2.1.1.0.1.1.0.0">my survey</a></li>
</ul>
  <ul class="Level2Items">
<li nowrap="nowrap" class="Nav2" id="id3"><a href="/cgi-bin/WebObjects/MyApp.woa/wo/49YrOFk9VT0wc2WYDlZ8b0/5.0.11.0.0.1.1.2.3.1.0.0.1.1.0.0">instructions</a></li>

<li nowrap="nowrap" class="Nav2Selected" id="id4"><a href="/cgi-bin/WebObjects/MyApp.woa/wo/49YrOFk9VT0wc2WYDlZ8b0/5.0.11.0.0.1.1.2.3.1.1.0.1.1.0.0">section I</a></li>
</ul>

</div>
```

CSS items that I defined in my application:

```
.ERXNavigationMenu {
  width: 800;
  margin: auto;
  border-top: #bdf solid 3px;
  border-left: #bdf solid 3px;
  border-right: #bdf solid 3px;
  border-bottom: #bdf solid 1px;
  padding-top: .5em;
  padding-left: 1em;
  padding-right: 1em;
  font-family: Helvetica;
}

ul.Level1Items {
  height: 2em;
  list-style: none;
  margin: 0;
  padding: 0;
  font-size: 1em;
  font-weight: bold;
}

ul.Level1Items li {
  background-color: #48f;
  float: left;
  margin: 0 2px 0 0;
}

ul.Level1Items a {
  background: #48f;
  color: #fff;
  display: block;
  float: left;
  height: 2em;
  line-height: 2em;
  text-decoration: none;
  padding-left: 10px;
  padding-right: 10px;
}

ul.Level1Items li.Nav1 a:hover {
  background-color: #3af;
  color: #000;
}

ul.Level1Items li.Nav1Selected {
  background-color: #bdf;
}

ul.Level1Items li.Nav1Selected a {
  background-color: #bdf;
  color: #008;
  font-weight: bold;
}

ul.Level1Items li.Nav1Selected a:hover {
  color: #008;
  font-weight: bold;
}

ul.Level2Items {
  height: 1.5em;
  list-style: none;
  margin: 0;
  padding: 2px 2px 0 2px;
  font-size: 1em;
  font-weight: bold;
}
```

```

    background-color: #bdf;
}

ul.Level2Items li {
    background-color: #fff;
    float: left;
    margin: 2px 2px 0 2px;
}

ul.Level2Items a {
    background: #bdf;
    display: block;
    float: left;
    height: 1.5em;
    line-height: 1.5em;
    font-size: .9em;
    font-weight: bold;
    text-decoration: none;
    padding-left: 10px;
    padding-right: 10px;
}

ul.Level2Items li.Nav2 a:hover {
    background-color: #3af;
    color: #000;
    font-weight: bold;
}

ul.Level2Items li.Nav2Selected {
    background-color: #bdf;
    font-weight: bold;
    text-decoration: underline;
}

ul.Level2Items li.Nav2Selected a {
    color: #008;
    font-weight: bold;
}

ul.Level2Items li.Nav2Selected a:hover {
    color: #000;
    font-weight: bold;
}

```

## 2. Using Conditionally Displayed Tabs

You can do something like:

### NavigationMenu.plist

```

{
    action = "session.toItemList";
    name = "item list";
    displayName = search;
    children = "session.isItemSelectedForNavigation";
    childrenChoices = {
        itemSelected = ("list", "details", "history");
        noItemSelected = ( "list");
    };
},

```

## Session.java

```
public String isItemSelectedForNavigation() {
    Item _selectedItem = (Item) selectedItem();
    if(_selectedItem !=null) {
        return "itemSelected";
    }
    return "noItemSelected";
}

public WOActionResults toItemList() {
    ItemList nextPage = (ItemList) pageWithName(ItemList.class.getName());
    setSelectedItem(null);
    return nextPage;
}

public WOActionResults toItemHistory() {
    ItemHistory nextPage = (ItemHistory) pageWithName(ItemHistory.class.getName());
    nextPage.setAnItem(_selectedItem);
    return nextPage;
}
```