Interface & Visual Design

# Interface Design Features

The following interface design features were used as part of the visual design of our site for the Code Fair:

## Persistent Header / Top navigation

We attempted to keep the page count minimal, so the top navigation contains only a few links.

We decided against a dropdown menu within the top navigation, the nature of the site encourages visitors to scroll through each page, without the need for the clutter of extra intra-page navigation in links. The “Learn More” buttons provide a more targeted pathway for visitors who are interested.

## Splash page

We liked the idea of having a splash page with the code fair logo and the sign-up button as an introduction to the site. A persistent image / theme that is overlaid as the user scrolls down the page gives the appearance and feel of separation of content and visual themes.

## Buttons

We chose to use buttons for the “Learn More” links within the main page to attract the user to engage further in the content.

## Forms

Keeping the form layout simple, we hoped to encourage visitors to submit an entry into the code fair.

## Persistent Footer

A persistent footer for easy navigation to “other” content, such as the site’s Privacy Policy, Site Map and Contact Us.

## Learn More links / buttons

We decided to keep the home page as simple as possible, with only brief descriptions of the Code Fair’s events. Adding “Learn More” buttons which navigate to another page with more detailed descriptions of events, allowed us to convey the information to visitors with a genuine interest, while keeping the home page brief.

Usability testing revealed the need to add easily discoverable links for more information.

## Cross-browser Compatibility and Extensions (JavaScript)

Our initial approach was to use a feature rich front-end development tool like Bootstrap.

We realized this was going to limit our site’s functionality on older browsers, our “vertical one-page scroll” would be JavaScript heavy, and be a different site altogether with JavaScript disabled.

PureCSS (Yahoo!) was just one styling tool we looked at that performs well using CSS alone, and allowed for easy JavaScript add-ons. It also is designed for use on older versions of Internet Explorer.

# Visual Design Inspirations

## Tapestry Website

<http://tapestry.life/welcome#HOME>

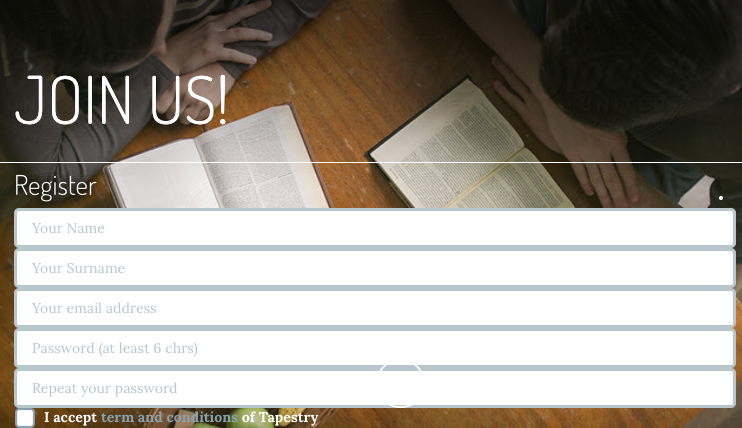
This website impressed us in terms of user experience. It was the first layout we considered when conceiving our wireframes. It relies heavily on the use of JavaScript however, and for our purposes would compromise user experience for users on non-JavaScript-enabled browsers.



Users can scroll down or click the down arrow to see the next “page”.



Three pages in total, the site displays just enough information to engage the user along with the animated “feel” of dropping to the next page.



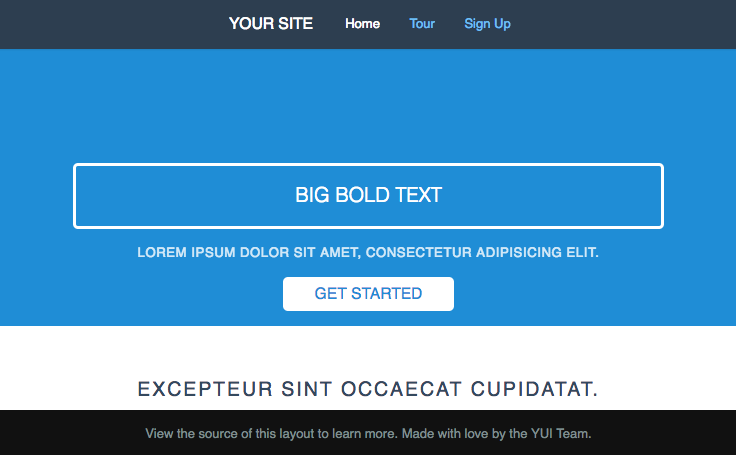
Having a full page dedicated to the Join Us form inspired the idea for the Sign-Up form on our site. While the “one-viewport-per-page scroll” feature was not included in our site, the idea remained with the placement of our form with full screen size on all pages.

## PureCSS layouts

<https://purecss.io/layouts/marketing/>

This was the foundation for our Code Fair site. It aligned with our idea to have a one-viewport per page scrolling feature, but did not require JavaScript.

PureCSS is also a minimal package, so page loading speed would not be a factor when loading stylesheets or scripts.



## Responsive grid system – purecss

PureCSS (Yahoo) contains a grid system which proved useful for the purposes of displaying our layout.

PureCSS also includes styling for responsive images, which simplified the task of cross device coding.

All that is required is the addition of a “pure-responsive-img” class to our photos and images, and PureCSS ensures the sizing remains responsive across devices.

# Client / User Feedback

## Navigation

As content has been progressively added, the main focus of user testing was on the site’s navigation. The overall navigation prototype was mostly functional, we anticipated the need to reduce content and to not have an excessive number of pages to browse through. Suggestions were made to add links to further information about coding events.

## Content

Suggestions and feedback we received for content focused on event information, and the sign-up form. We needed to make these both easily discoverable. The sign-up form had to be accessible from anywhere on the website. Testing showed users would access the form via scrolling, clicking a button, or using the top navigation link. The form had to be simple enough to not deter the user from filling it out. Content needed to be browsed easily. This meant concise headings and information. Our testers tended to click through links, “just to look”.

## Experience

User experience feedback was positive during the prototype testing. Generally, users said the site “looked good”, and commented that it worked well on their mobile phones. Site navigation was achieved with relative ease; early testing demonstrated the need to compress our layout within 2-3 pages to maintain user’s attention.