

Latest Trends and Technologies in User Interfaces

ARUN KUMAR ARUMUGAM

Senior IT Architect
IQVIA, Pennsylvania, USA.

Abstract: User Interface plays a key role in the field of Information Technology. It helps the business to provide better visualization of the information and expand the business. There are various User Interface types and multiple ways to implement the User Interface. In this paper, we are going to discuss the various types of user interfaces and its implementation

Index Terms: User Interfaces, Ember JS Backbone JS, Vue JS, Angular JS, React JS, GUI, CLI

I. INTRODUCTION

The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website. There are various types of User Interfaces available which are discussed in this paper. Also, there are multiple frameworks available to implement the user interface which are also discussed below.

II. WHAT IS USER INTERFACE AND ITS TYPES

User Interface is a visual part of computer application or operating system through which a user interacts with a computer or a software. It determines how commands are given to the computer or the program and how information is displayed on the screen. There are different ways of interacting with computer systems which have evolved over the years. There are five main types of user interface:

- Command Line (CLI)
 - CLI is the oldest User Interfaces. It involves the computer responding to commands typed by the operator.
 - This type of interface has the drawback that it requires the operator to remember a range of different commands and is not ideal for novice users.
- Graphical User Interface (GUI)
 - It allows the user to interact with devices through graphical icons and visual indicators such as secondary notations.
 - Operators use a pointing device (such as a mouse, touchpad or trackball) to control a pointer on the screen which then interacts with other on-screen elements.
- Menu Driven (MDI)
 - A menu driven interface is commonly used on cash machines (also known as automated teller machines (ATM's), ticket machines and information kiosks.
 - They provide a simple and easy to use interface comprised of a series of menus and sub-menus which the user accesses by pressing buttons, often on a touch-screen device.
- Form Based (FBI)
 - A form-based interface uses text-boxes, drop-down menus, text areas, check boxes, radio boxes and buttons to create an electronic form which a user completes in order to enter data into a system
 - This is commonly used on websites to gather data from a user, or in call centres to allow operators to quickly enter information gathered over the phone.
- Natural Language (NLI)

- A natural language interface is a spoken interface where the user interacts with the computer by talking to it. Sometimes referred to as a 'conversational interface', this interface simulates having a conversation with a computer.
- An Example of this type of interface is Voice Recognition

III. MOST POPULAR USER INTERFACES

Out of many types of User Interfaces, Graphical User Interfaces (GUI) are most popular and used by leading business organizations

A GUI displays objects that convey information and represent actions that can be taken by the user. The objects change color, size, or visibility when the user interacts with them.

GUI objects include icons, cursors, and buttons. These graphical elements are sometimes enhanced with sounds, or visual effects like transparency and drop shadows.

A GUI is considered to be more user-friendly than a text-based command-line interface, such as MS-DOS, or the shell of Unix-like operating systems.

The GUI was first developed at Xerox PARC by Alan Kay, Douglas Engelbart, and a group of other researchers in 1981. Later, Apple introduced the Lisa computer with a GUI on January 19, 1983.

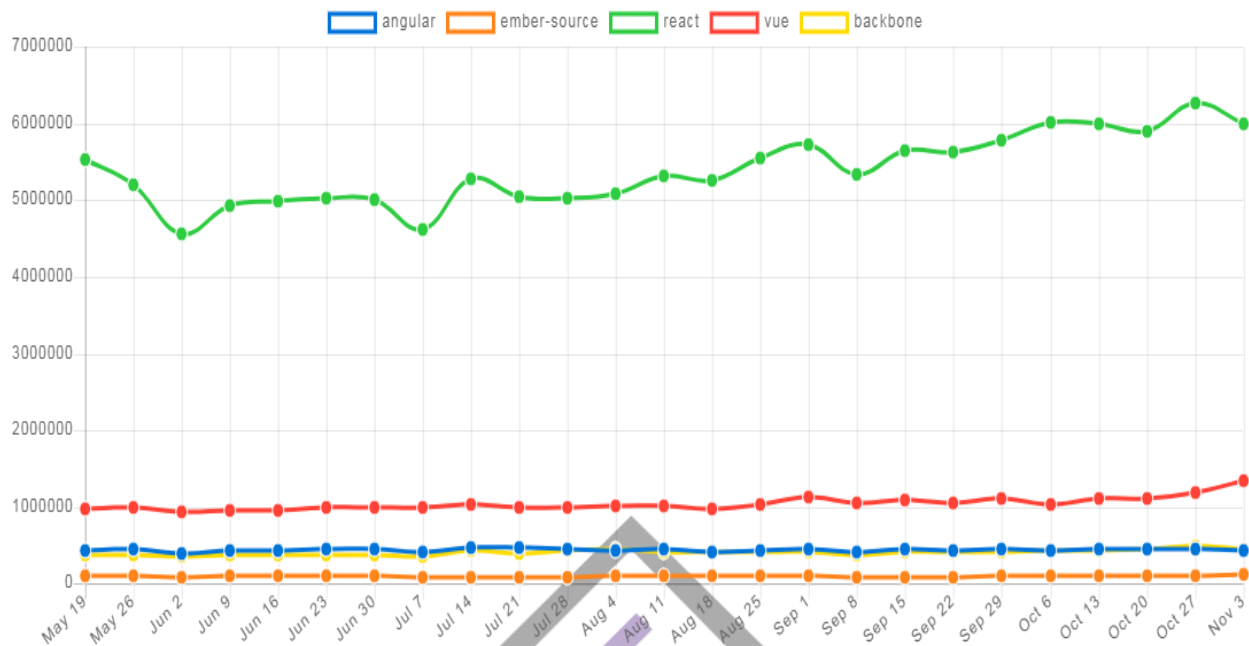
IV. HOW TO BUILD GUI?

GUI can be built with many ways. There are various programming languages available like Java Swings, Python, HTML, CSS, JavaScript, Dot net and so on. Out of which JavaScript framework GUI developments are more familiar and emerging in the field of information technology. Let's discuss about the latest JavaScript frameworks available in the market.

The top five JavaScript frameworks that currently dominate the market in terms of popularity and usage are:

- React
- Vue
- Angular
- Ember
- Backbone.js.

They each have large communities. If you are a front-end developer or are going to start your new project on front-end technologies, these five are your best bets. Here's a look at the npm trends over the last six months.



A. React

React is the definite leader in the JS world. This JavaScript framework uses a reactive approach and also introduces many of its own concepts for front-end web development.

To use React, we have to learn to use a plethora of additional tools to reach high flexibility in front-end development. For example, here's a less exhaustive list of libraries you can use with React: Redux, MobX, Fluxy, Fluxible, or RefluxJS. React can also be used with jQuery AJAX, fetch API, Superagent, and Axios.

React is constantly working towards improving concurrent mode. To take this forward, React Conf 2019 wrapped up last month where the React team talked about improving Concurrent Mode and the Suspense model. Both the features make React apps more responsive by rendering trees without blocking threads. This allows React to focus on high priority tasks like responding to user input.

React also introduced Suspense to improve the developer's experience when handling asynchronous data fetching in React apps. In short, the new update to Suspense lets the component wait until a condition is met.

Hooks are another important update to React 16.8. React hooks lets you use every important feature of React – server-side rendering, accessibility, concurrent mode, and suspense – all without writing a class.

React applications are divided into multiple components that contain both business logic and HTML markup functions. To improve the communication between components, developers can use either Flux or a similar JavaScript library.

React also introduced objects, like state and props. With the state and props objects, you can simply pass data from a component to the layout or from a parent component to a child component.

Introduction to the React ecosystem:

- The React library plus React router for implementing routes.
- React-DOM for DOM manipulation.
- React developer tools for Firefox and Chrome browsers.
- React JSX, a markup language that mixes HTML into JavaScript.
- React Create App command line interface to set up a React project.
- Redux and Axios libraries to organize communication with the backend team.

No doubt, React is one of the most popular JavaScript frameworks. And, I think that React can be your first choice for creating advanced-grade apps.

B. Angular 2 to Angular 9

Angular 9 will mark a turning point revealed by the Angular team at the recent AngularConnect 2019. According to the update, the team is planning to make the Angular Ivy compiler available for all apps. The main benefit of Angular Ivy is that it is able to reduce the size of applications.

Angular today has become very advanced and modular to use for front-end development. Previously you could insert a link to the AngularJS library in the main HTML file, but now you can do the same by installing separate modules.

Angular's flexibility is commendable. That's why Angular's 1.x versions are still in demand. However, many developers currently rely on Angular 2+ because of its MVC architecture which has changed substantially to a component based architecture.

Angular has a couple of additional challenges. You're almost obliged to use TypeScript to ensure type safety in Angular apps. TypeScript makes the Angular 2+ framework not so pleasant to work with.

Angular's ecosystem is comprised of:

- For quick project setup, Angular's command line interface is helpful.
- Developers will get a set of modules for Angular projects: @angular/common, @angular/compiler, @angular/core, @angular/forms, @angular/http, @angular/platform-browser, @angular/platform-browser-dynamic, @angular/router, and @angular/upgrade.
- Angular uses Zone.js a JavaScript library to implement zones in Angular apps.
- TypeScript and CoffeeScript both can be used with Angular.
- For communication with server-side apps, Angular uses RxJS and the Observable pattern.
- Angular Augury for debugging Angular apps.
- Angular Universal for creating server-side apps with Angular.
- Angular2 is a complete JavaScript framework with all the tools a modern front-end developer needs. You can choose Angular if you don't like to work with additional libraries as with React.

3. Vue

The Snyk JavaScript framework report for 2019 is out. The report mainly focused on security risks in both React and Angular.

The concept of Vue has been taken from Angular and React, but Vue is better in many ways. I'll talk about its features, but first check out what the Snyk report says about Vue's front-end security. Vue has been downloaded 40 million times this year and records only four direct vulnerabilities. All of them have been fixed.

For any front-end developer unfamiliar with Vue, let's clarify several points.

With Vue you store component logic and layouts along with stylesheets in one file. This is the same way React works, without stylesheets. To let components talk to each other, Vue uses the props and state objects. This approach also existed in React before Vue adopted it.

Similar to Angular, Vue wants you to mix HTML layouts with JavaScript. You have to use Vue directives such as v-bind or v-if to interpolate values from the component logic to templates.

One of the reasons why Vue is worth considering instead of React is because of the Redux library that's often used in large-scale React applications. As explained in the React section, when a React+Redux app grows bigger, you'll spend a lot of time applying small changes to multiple files instead of actually working on features. The Vuex library – a Flux-like state management tool designed for Vue – seems less unwieldy than Redux.

If you're choosing between Vue and Angular, the reasons to opt for Vue over Angular can be reduced to the following: Angular is an over-complicated, full-fledged framework with a restrictive nature; Vue is much simpler and less restrictive than Angular.

Another advantage of Vue over Angular and React is that you don't have to learn JavaScript once more.

An introduction to the VueJS ecosystem:

- Vuex comes with a dedicated library for application management.
- Vuex is similar to the concept of Flux.
- You will get Vue-loader for components and vue.js devtools for Chrome and Firefox browsers.
- Vue-resource and Axios tools for communication between Vue and the backend source.
- Vue.js support Nuxt.js for creating server-side applications with Vue; Nuxt.js is basically a competitor to Angular Universal.
- You will get a Weex JavaScript library with Vue syntax that is used for mobile app development.
- Vue is excellent in terms of its workflow to other frameworks. I might opt for Vue because it's less complicated than React and Angular JS and a great choice for developing enterprise-level apps.

4. Ember

Ember 3.13 released this year with some new updates and features. Ember is just like Backbone and AngularJS and is also one of the oldest JavaScript frameworks. But with the new update, Ember 3.13 is compatible with new bug fixes, performance improvements, and deprecation. Tracked property updates have also been introduced that allow simpler ways of tracking state change in the ergonomic system of Ember apps.

Ember has a relatively intricate architecture, which will allow you to quickly build huge client-side applications. It realizes a typical MVC JavaScript framework, and Ember's architecture comprises the following parts: adapters, components, controllers, helpers, models, routes, services, templates, utils, and addons.

One of Ember's best features is its command line interface tool. The Ember CLI helps front-end developers be highly productive and lets them complete projects on time. You can not only create new projects with ready setups, but you can also create controllers, components, and project files using automatic generation.

The EmberJS ecosystem is comprised of:

- Ember CLI tool for quick prototyping and managing dependencies.
- Ember server built into the framework for the development of apps.
- You'll get Ember.js library and Ember Data for data management.
- Handlebars template engine for Ember applications.
- QUnit testing framework for Ember.
- Ember Inspector development tool for Chrome and Firefox browsers.
- Ember Observer for public storage and Ember addons to implement generic functionalities.

Although Ember is underrated, it's perfect for creating complex client-side apps.

5. Backbone.js

Backbone is a JavaScript framework based on the MVC architecture. In Backbone.js, the View of MVC helps implement component logic similarly to a Controller. Backbone view can use engines like Mustache and Underscore.js.

Backbone is an easy to use JavaScript framework that allows for quick development of single page applications. To use Backbone.js to the fullest extent, you'll have to choose tools: Chaplin, Marionette, Thorax, Handlebars or Mustache, and so on.

If you need to design an app that has different types of users, Backbone collections (arrays) can be used here to separate the models. Backbone. Events can be used with Backbone models, collections, routes, and views.

Introducing the BackboneJS ecosystem:

- The Backbone library consists of events, models, collections, views, and router.
- Underscore.js, a JavaScript library with helper functions that you can use to write cross-browser JavaScript.
- You can use template engines such as Mustache and jQuery-tmpl.
- BackPlug online repository with a lot of ready solutions for Backbone-based apps.
- Backbone generator CLI for building Backbone apps.
- Marionette, Thorax, and Chaplin JavaScript libraries to develop an advanced architecture for Backbone apps.

Backbone.js is a perfect choice for front-end and back-end development as it supports REST APIs that are used to synchronize the front-end and back-end.

EMERGING TRENDS IN USER INTERFACE

For the last few years advances in web design have meant creating fancy flash and experimental sites that assault the user's senses, with usability often an after-thought.

However, the really great user interface (UI) design lets users complete their goals rather than distracting them with over-complicated elements. Users generally prefer websites that are intuitive to use and require a minimal number of clicks to reach the outcome they're after.

Improving your user interface:

The navigation should make it easy for the user to move around, interact with and understand the website they're visiting without prior help or explanation.

Visitors to your site don't want to do a lot of work tracking down the product or area they're looking for and they don't want to go through a lengthy signing up process. This is frustrating and will encourage them to look elsewhere.

Luckily, web developers and graphic designers are returning to basics, and considering usability factors again. Ideally a site should include great design, technology and usability, for an experience that is useful as well as pleasurable.

Our pick of important emerging trends in user interface

Responsive Design

Responsive web design has been around for a couple of years now and is ever improving. Sites are visited from an increasing range of devices, including mobile browsers, netbooks, tablets and desktop computers and it's important that a website function well on all of these.

To keep ahead of the curve companies are increasingly adapting to the new browsing habits of their audience, anticipating and responding to user's needs. Responsive design considers usability across a variety of modern mediums, making sure the layout is flexible and adaptable for every device users could be viewing a website or email on, with a variety of possible browsers.

This keeps continuity in the design and means that designers and developers only have to work on one version of a website, instead of multiple ones for desktop computers and mobile devices.

A good example of responsive design comes from digital duo Dassel and Wagner.

Flat Design

Flat design is a recent trend. The fussiness of drop shadows, textures and gradients are abandoned in favor of solid colors, sharp and striking typography and simple layouts. This makes for a fresh and appealing site with simplified navigation.

By stripping away all the unnecessary elements, usability becomes central to visitor's experience again. The big bright style of Web 2.0 is stripped back to focus on a clean, minimal aesthetic, stylish typography and information hierarchy.

When done right this approach is popular with users. The interface is easy to digest, and its elements don't get in the way of the tasks they're trying to accomplish.

Flat design is a reaction against the over-the-top designs of recent years, and therefore a trend that is unlikely to last a long time, especially as some clients may find this too simplistic for their tastes.

Take a look at this beautiful example from Layervault.

Laser Focus

Laser focused user interface means drawing the visitor's focus to a single, obvious task to do as they open a page, instead of providing a variety of options. This makes it easy for the visitor to see instantly the purpose of the site and what it provides, instead of confusing them about where exactly they should be clicking.

This approach is ideal for a site that has a single, important function it centers on – for example a search engine or hotel booking site. The Google homepage is of course the most obvious example of this.

Skeuomorphism

Skeuomorphism was originally popularized by Apple.

The concept is that real-life, familiar objects are imitated - for example, displaying a range of ebooks in a wooden bookshelf, with book cover representing the digital content, or a notes app with the appearance of note paper. It is most commonly used for mobile applications, as these devices are touch screen and tactile.

While imitating a familiar object may be fun and easy to use for users, it can also be argued that skeuomorphism is restrictive, as it means accepting the limitations of the object it's based on.

A lot of the best UI elements imitate physical or mechanical elements from everyday life however: pushing buttons, turning a dial or the pages of folders, so many see skeuomorphism as just the next step on from this.

Take a look at bagigia.com, where you pull at the zip of one of the brand's bags to scroll through images of the bag itself as well as the manufacturing process. This really brings the bag and its story to life.

Single Page Sites

Creative and visually appealing one-page sites are now flourishing, with big background images, animation and stunning illustration. Ignoring the limitations of traditional website architecture, they allow for simplicity without being dull.

Although once frowned on for the scrolling they require, for certain sites single page may be the best option. As we have all become so used to scrolling, some see it as easier to keep scrolling than to split the content over a number of pages, which visitors will have to track down.

This style won't work for everyone but looks great when done well. It is best used for telling a story or presenting product features and benefits. Take a look at this amazing example: <http://jessandruss.us/>

CONCLUSION:

User Interfaces plays a core role in the information technology business. There are various techniques and methodologies available to implement the user interface. However, the user can choose the apt framework and methodology based depends on the project architecture and requirement.