Contents

[HTML5 3](#_Toc479589167)

[Overview 3](#_Toc479589168)

[New Elements in HTML5 3](#_Toc479589169)

[New APIs in HTML5 3](#_Toc479589170)

[UX Principles 3](#_Toc479589171)

[Digestibility 3](#_Toc479589172)

[Clarity 3](#_Toc479589173)

[Trust 3](#_Toc479589174)

[Familiarity 3](#_Toc479589175)

[Delight 3](#_Toc479589176)

[JavaScript Libraries 4](#_Toc479589177)

[JSON 4](#_Toc479589178)

[Protobuf (Protocol buffer) 4](#_Toc479589179)

[JQuery 4](#_Toc479589180)

[JQueryUI 4](#_Toc479589181)

[Sencha 4](#_Toc479589182)

[UnderscoreJS 4](#_Toc479589183)

[BackboneJS 4](#_Toc479589184)

[KnockoutJS 4](#_Toc479589185)

[KandoUI 4](#_Toc479589186)

[RequireJS 4](#_Toc479589187)

[Modernizer 4](#_Toc479589188)

[ReactJS 5](#_Toc479589189)

[AngularJS 5](#_Toc479589190)

[NodeJS 5](#_Toc479589191)

[Rhino 5](#_Toc479589192)

[NPM 5](#_Toc479589193)

[Bower 5](#_Toc479589194)

[Jasmine 5](#_Toc479589195)

[Grunt 6](#_Toc479589196)

[JavaScript Libraries 6](#_Toc479589197)

[Redux 6](#_Toc479589198)

# HTML5

## Overview

* It is the latest version of HTML which includes new elements, attributes & behaviors.
* Used to build diverse web applications

## New Elements in HTML5

* Semantic elements <header> <footer> <article> and <section>
* New attributes for form elements number, date, time, character, calendar, range
* New graphic elements <svg>, <canvas>
* New multimedia elements <audio>, <video>

## New APIs in HTML5

* HTML Geolocation
* HTML Drag & Drop
* HTML Local Storage (A powerful replacement for cookies)
* HTML Application cache
* HTML Web Workers
* HTML SSE

# UX Principles

## Digestibility

* By looking at the page or design user should be able to use the page
* Design must be self explanatory. We should not provide more text to explain the field to user

## Clarity

* User must know exactly what they are buying before they click on buy now

## Trust

* Good design is easy to trust
* Before you ask someone to complete an action, make every effort to make them realize why this is needed
* Example Uber / Lift – users comfortable sharing payment with the system than a person / driver

## Familiarity

* The design must work fine in all devices (unlike flash)

## Delight

* Make the design a useful thing in a person’s life

# JavaScript Libraries

## JSON

## Protobuf (Protocol buffer)

* A binary serialization format from Google meant to serialize structured data
* It is light weight & typed when compared with JASON

## JQuery

* A simple Lightweight, write less , do more JavaScript library
* Easy to learn

## JQueryUI

* Written on top of JQuery , it is a set of user interfaces, effects, widgets & themes

## Sencha

## UnderscoreJS

* Provides utility programs for common programming tasks

## BackboneJS

* A JavaScript framework with RESTul JASON interface
* MVP (Model-View-Presenter) design Pattern

## KnockoutJS

* Standalone JavaScript implementation of Model View – View Model design pattern with templates

## KandoUI

* HTML5 based framework to build user interface

## RequireJS

* It is a module loader, it will help improve the speed and quality of code

## Modernizer

* It detects which HTML5 & CSS3 features a visitors browser supports
* It enable the developer to provide fall back option when browser do not support certain features

## ReactJS

* A JavaScript library for building UI
* Developed by Facebook
* It is component based
* It updates the UI dynamically when the data is changed

## AngularJS

* Used to add dynamic views to a Web Application
* Implements MVW (Model-View-Whatever) architecture
* Supports Dependency injection & data binding
* Used to build data driven applications
* Decoupling of DOM and application logic
* With features such as directives, filters and automatic databinding’s developer need not write more number of lines of code
* Cross browser capability & Mobile support
* It comes with a test runner called Karma which facilitates unit testing
* Protractor can be used for E2E Testing. Protractor runs tests against the application running in real browser
* Supports object oriented, event driven & functional programming paradigms

## NodeJS

* A JavaScript runtime built on Chromes V8 engine
* Helps to build scalable network applications
* Implements MVC Architecture
* It uses event driven non blocking IO model that makes it light weight & efficient
* Ideal for making data-intensive real time applications running on distributed systems
* Not suitable for CPU oriented applications
* Supports Object-oriented, Event-driven, Functional, Concurrency Oriented, Pub/Sub programming paradigms
* Offers support for Ruby, CoffeeScript and TypeScript Scripting languages
* It Communicates with databases, web-sockets, middle-ware etc
* A node JS application consists of the following
  + Import the required modules / var http = require("http")
  + Create Server / http.createServer(function (request, response) {
  + Read request & return response

## ExtJS

* Most comprehensive framework for building rich cross platform web applications
* Has ability to add responsive design styles, layouts, UI Components & theming
* Cross browser compatibility & Mobile support
* Offers powerful chart that can be used without any plugins
* Supports both MVC & MVVM architecture
* Supports object oriented & event driven paradigms

## Rhino

* Open source implementation of JavScript written entirely in Java
* It is typically embedded into Java applications to provide scripting to end users
* It converts JavaScript to classes

## NPM

* Dependency management tool for installing node.js modules

## Bower

* Dependency management tool for managing front end components like HTML, CSS, JS etc

## Jasmine

* Open source testing framework for JavaScript
* Influenced by other testing frameworks ScrewUnit, JSSpec,JSpec,Rspec

## Grunt

* A task runner for JavaScript (Similar to Ant for Java)
* Helpful for Automation (for executing repetitive tasks such as build, unit test etc)

# JavaScript Libraries

# Redux

* It’s a predictable state container for JavaScript
* It helps in building applications to work consistently across different environments (Client, Server or Native)
* It also helps to debug code