**UX Design**

**Duration: 4 Days**

**Day 1**

Creating Persona's & Ideation

**What are personas?**

**Marketing personas**

**Proto—personas**

**Design personas**

**Characteristics of a good persona**

**Business** **cases for personas**

**The Value of Personas**

**Using personas throughout the design process**

Creating Scenarios & Storyboard

**Why Stories?**

**How UX Stories Work**

**Stories Start with Listening and Observing**

**The Ethics of Stories**

**Stories as Part of a UX Process**

**Collecting Stories (as Part of Research)**

**Selecting Stories (as Part of Analysis)**

**Using Stories for Design Ideas**

**Evaluating with Stories**

**Sharing Stories (Managing Up and Across)**

**Crafting a Story**

**Considering the Audience**

**Combining the Ingredients of a Story**

**Developing Structure and Plot**

**Ways to Tell Stories**

Research Fundamentals

**Colorful Emotions: Visual Vibrancy in Web UI Design**

**Uncovering User Expectations Before Design**

**Day 2**

Knowing Your User

**Learn how to structure and conduct user interviews to better identify the needs and current behaviors of the user.**

**Understand how to engage the user to work through assumptions and preconceived notions that arise during the process.**

**Write down assumptions for your class project.**

Homework:**Interview 1–5 people about your project.**

User Research and Personas

Review user research questions/review homework.

Refine user interview questions.

Interview classmate for your project.

Create user personas from the interviews to represent the goals and behaviors of the users within the system.

Develop user scenarios to show when and how the user engages with the system.

**Homework:** Begin to synthesize results of user research.

Synthesis of User Research, Journey Maps, and User Flows

Continue to synthesize results of user research to find patterns.

Create a journey map for your project.

Think through the different features of the system by building user flows.

Take the written narrative to the next level through sketching and improvising using techniques such as storyboarding and bodystorming.

**Homework:** Refine your user flows.

Wireframing, Sketching, Prototyping

Practice a sketching session for an existing website or mobile application of your choice. Start with sketching on a whiteboard or your notebook.

Sketch wireframes for your class project.

Learn Sketch to create wireframes/prototypes.

Learn the power of prototyping and testing an idea. Understand different methods of prototyping and their levels of fidelity.

Discuss the motivation behind the prototyping, such as newly discovered user goals, business needs and improved functionality.

**Homework:** Finish prototyping your user flows.

Prototyping & User Testing

InVision for clickable prototypes.

Use Sketch and InVision to create a clickable prototype for your project.

User testing methods and strategies.

Do's and don'ts of user testing.

Do a user test on your project with a classmate.

Clearly list the assumptions and test your concepts. Work on multiple iterations of the prototype based on the test results.

**Homework:** Test your prototype on 1–5 users.

Iterating the Prototype and Further User Testing

Report conclusions based on user testing.

Create a user testing report.

Rework the prototype based on user testing conclusions.

Case studies.

**Homework:** Finish iterating the prototype based on user testing.

**Homework:** Create a case study of your project.

Usability and UCD Methodology

Day 3

HTML5

Introduction:

Overview of HTML5

History of HTML5

The Myth of 2022 and Why It Doesn’t Matter

Who Is Developing HTML5?

A New Vision

Compatibility and Paving the Cow Paths

Utility and the Priority of Constituencies

Interoperability Simplification

Universal Access

A Plugin–Free Paradigm

What’s In and What’s Out?

What’s New in HTML5?

New DOCTYPE and Character Set

New and Deprecated Elements

Semantic Markup

Simplifying Selection Using the Selectors API

JavaScript Logging and Debugging

window JSON

DOM Level 3

Using WAI-ARIA with HTML5 for Accessibility

CANVAS API:

Overview of HTML5 Canvas

Canvas Coordinates& registering the Canvas dimensions

Drawing on Cavas with paths,curves etc.

Working with Solid colors,Gradients & Transparancy

Importing External Images & Setting the background

Working with Color & Geometrical transformations

Creating text,graphs & charts

Animating a Vertical Bar-Chartwith fine tuning.

Working with Pixel Data

CSS and Canvas

Create High-Res, Retina-Display-Ready Media with Canvas

Clipping Cavas drawings & saving them to a file.

When Not to Use Canvas

Fallback Content

Implementing Canvas Security

Implementing techniquesfor Backward compatibility.

Building an Application with HTML5 Canvas

SVG API :

Understanding SVG

Creating 2D Graphics with SVG

Adding SVG to a Page

Simple Shapes & Text

Transforming SVG Elements

Reusing Content

Patterns and Gradients

SVG Paths

Building an Interactive Application with SVG

Adding the CSS Styles

Implementing techniques for Backward compatibility.

Activity

Magazine Layout creation using semantic tags

System File Directory and event handling using SVG

Creating custom shapes using canvas

Building an Application with SVG

GEOLOCATION:

Comparing Geolocation techniques in the past & modern day Geolocation

Understanding GPS/ IP Address/ Cell IDs/ WiFi and Bluetooth

LBS (Location based services)

Understanding Latitude,Longitude,Speed,Course & Accuracy

Getting you current location

Browser compatibility & Fallbacks.

Reverse geocoding&Mapping location

Getting Distance & Directions between two places.

Following a moving location

Combing geolocation with google maps

Triggering the Privacy Protection Mechanism

Saving Geographical information

Geolocation usage – Geo Marketing,Geo social,Geo tagging,Geo tagging & Geo applications.

Building a Real-Time Application with HTML5 Geolocation

Implementing techniques for Backward compatibility.

Alternative methods when Native Geolocation fails (Geo.js & MaxMind)

Media API (video & audio):

Flash V/s HTML5 video

Adding Video & Audio to a page

Supported Audio & Video formats& Codecs

Lossy & Lossless compression

Media specific attributes Vs Global attributes

Deployment challenges on Mobiles Converting Audio & Video to supported formats using open source & commercial software

Using a Frame grabber

Custom Controls,Seekbar,Progressbar with Javascript & CSS

Applying CSS skins & transforms

Working with multiple tracks,Subtitles & Captions with Captionator,Playr & the Leanback Player

Integrating Video with Canvas & SVG

Applying Visual filters using Canvas & SVG

Debuggin,Browser support &Licensing issues.

Implementing techniques for Backward compatibility.

Building an Application with HTML5 Media API

WEB FORMS API:

HTML forms fundamentals:

Introduction

Form Basics

Demo Form Basics

Form Settings

Demo Form Settings

Summary

HTML form inputs:

Introduction

Text inputs

Demo Text inputs

Selections

Demo Selections

Input attributes

Demo Input attributes

Input commands

Demo Input commands

Summary

Activity

Generating Map for the given address

Creating Entertainment Library like you tube

Organizing HTML forms:

Introduction

Labels

Fieldsets

Tab index

Access keys

Summary

HTML form scenarios:

Introduction

Scripting forms

Handling multiple forms

Uploading files

Summary

Constraint Validation: Native Client Side Validation for Web Forms

What is Constraint Validation?

DOM API

willValidate

validity

validationMessage

checkValidity()

setCustomValidity()

HTML Attributes

Novalidate

Formnovalidate

CSS Hooks

:invalid and :valid

Resetting Default Styling

Inline Bubbles

Removing the Default Bubble

Current Implementation Issues and Limitations

setCustomValidity

Declarative Error Messages

Title Attribute

Activity

Creating Order Form

WEB WORKERS API:

What are web workers ?

Possibilities & Limitations of web workers

Inline,Dedicated & Shared Workers

Creating a worker,Assign roles & Deploying the same.

Leveragin a Shared Worker

Worker support in modern browsers

Managing multiple workers

Parsing data with workers

Perform Heavy array computations

Using timers in conjunction with worker

Work with pixel manipulations

Make twitter JSONP requests

Connect to share workers at same time with multiple browser windows

Transferable objects

Debuging Your Workers

Implementing techniques for Backward compatibility.

Building an Application with HTML5 Web Workers API

HTML5 Custom Data Attributes (data-\*)

Attribute Name

Attribute Value

How can I use data attributes?

What shouldn’t I use data attributes for?

Using data- attributes with JavaScript

A word of warning

Activity

Creating Worker for background I/O

Reading the length of songs

CROSS DOCUMENT MESSAGING API:

Understanding Origin Security

Browser Support for Cross Document Messaging

Building an Application Using the postMessage API

XMLHttpRequest Level

Cross-Origin XMLHttpRequest

Progress Events

Browser Support for HTML XMLHttpRequest Level

Building an Application Using XMLHttpRequest Structured Data&Framebusting

Implementing techniques for Backward compatibility.

Building an Application with HTML5 CDM

Server Sent Events (SSE)

Possible Applications

Overview of the API

new EventSource(url)

Properties of Server-Sent Events

Message Format

Typical Server

Polyfills and Tweaks to the Server

Why Not Use WebSockets

Activity

Creating message using server sent event

WEB SOCKET API:

Understanding WebSocket

WebSocket API

WebSocket Protocol

Writing a Simple Echo WebSocket Server

Using the WebSocket API

Checking for Browser Support

Building a WebSocket Application

Adding the Geolocation Code

Combining Geolocation & Web sockets together.

Building Instant Messaging and Chat over WebSocket with XMPP

Using Messaging over WebSocket with STOMP

VNC with the Remote Framebuffer Protocol

WebSocket Security

Deployment Considerations

Inspecting WebSocket Traffic

Implementing techniques for Backward compatibility.

Web RTC API:

Introduction to Web Real-time Communication (WebRTC)

Introduction

History of Real-time Communication on the web

What you can do with WebRTC

Where WebRTC is supported

Architecture of a WebRTC Application

Introduction

Security with WebRTC

The Full WebRTC Environment

Introducing the WebRTC API

Overview of the WebRTC APIs

Accessing Webcam and Microphone with MediaStream (getUserMedia)

Establishing a Peer Connection with RTCPeerConnection

Understanding Data Communication with RTCDataChannel

Recap of the Module

Setting Up Your Development Environment

Introduction to Setting Up Your Environment

Software and tools for WebRTC development

Recap of the module

Create a Multi-person Chat Application Using SimpleWebRTC

Introduction to SimpleWebRTC Framework

Set up the HTML for Multi-person Video Chat

Write JavaScript Calls to SimpleWebRTC

Test the Multi-person Chat Application

Recap of the Module

Activity

Deonstrating live video using webrtc

WEB STORAGE & File System API:

**Web Storage:**

Introduction

What's in a Name?

What is Web Storage?

Capacity

Testing Capacity

Features

Browser Support

Fallbacks and Polyfills

Caveats

Getting and Setting Values

Remove Item

Keys and Length

Clear

Session Storage

Exceed Quota

Storage Event

Persistent Form Demonstration

HTML Markup

localDataService JavaScript Module

viewModel JavaScript Module

Stepping Through the Code

Summary

Building an Application with HTML5 local/session storage

IndexedDB Introduction and Concepts:

Introduction

What is IndexedDB?

Event Lifecycle

Features

Capacity

Browser Support

Fallbacks and Polyfills

Caveats

Summary

IndexedDB Initialization & CRUD :

Introduction

Demo Opening a Database

Demo Deleting a Database

Demo The db Model Object

Demo Create Object (Insert)

Demo Read Object

Demo Update Object

Demo Delete Object

Summary

IndexedDB Cursors, Indexes and Ranges :

Introduction

Demo db Model for Cursor, Index and Range Demos

Cursor Concepts

Demo Cursors - Selecting Sets of Data

Demo Indexes - Selecting Individual Objects

Range Concepts

Demo Numeric Range

Demo String Range

Demo Controlling Cursor Direction

Summary

IndexedDB Keys, Capacity, Performance and Versions:

Introduction

Unique Identifier (Keys) Concepts

Demo Creating Object Store Keys

Demo Loading k Objects into a Database

Detecting When a Cursor is 'Done'

Demo Working with Large Sets of Data

Demo Managing Database Versions

Demo Capacity Capabilities

Summary

IndexedDB Abstractions & Implementing an Edit Screen:

Introduction

Demo Introduction to the Homes List Screen

Demo Homes List Markup

Demo Homes List db Model

Demo Abstracting IndexedDB - Error Handling

Demo Abstracting IndexedDB - Delete and Open Database

Demo AbstractingIndexedDB - Get All

Demo Abstracting IndexedDB - Insert, Update and Delete

Demo Homes List View Model

Demo Stepping Through the Code

Summary

Building an Application with HTML5 Indexed DB API

File System Introduction, Concepts & Initialization:

Introduction

What is the HTML File System?

Features

Capacity

Browser Support

Fallbacks and Polyfills

Caveats

Storage Types

Demo Initialization (Temporary Storage)

Demo Initialization (Permanent Storage)

Demo Wrapping Up Initialization Code

HTML File System Explorer (Chrome Extension)

Summary

Activity

Storing favourite foods in localstorage

Storing Customer Data in indexed Db

OFFLINE APPLICATIONS API:

Understanding Offline or Occasionally connected applications

Cache manifest - Cache,Network & fallback in detail

Create &Use the manifest to detect connectivity

Updating cache with the manifest

Application Cache API

Understading Events under the AppCache API :

i.e, load,checking,noupdate,downloading,progress,

cached,updated ready,obsolete & error events

Disk Space & Expiration

Deleting the local cache

Implementing techniques for Backward compatibility.

Building an Application with HTML5 Offline Applications API

NOTIFICATIONS API:

Notification Permissions

Browser Compatibility

Displaying a Simple Notification

Creating a Tweet Notification Page

Implementing techniques for Backward compatibility.

DEVICE API’s - CONTACTS,MESSAGING & NETWORK INTERFACE API:

Retrieving All Contacts and Mobile Numbers

Battery Status Events

HTML Media Capture with File Input

Device Orientation and Motion Events

Creating a Bubble level

Implementing techniques for Backward compatibility.

CSS 3

CSS3 Features

Colour and transparency

New selectors and pseudo-classes

Layout: FlexBox, multiple columns, box-sizing,

grids

Fonts: web and icon fonts

Animation and transitions

Responsive web design using media queries

Responsive techniques: sprites, font sizes,

fluid and fixed grids

Box shadows, text shadows, backgrounds,

and border-radius

Implementing CSS3

Working with vendor prefixes

Browser support

Polyfills, Modernizr.js

Working with older versions of IE

**CSS – SASS**

* + Introducing SASS
  + SASS on the Server
  + Variables
  + Rules
  + Import
  + Extend
  + Mixin
  + Function
  + control directives

Bootstrap

Introduction

* What is Twitter Bootstrap
* Bootstrap features and components
* Bootstrap and Responsive Web Design
* Tools for development
* Responsive workflow
* Getting started with Twitter Bootstrap

Creating Structure

* The Grid
* Media Queries
* Responsive Layout
* Fluid grid system
* Nesting
* Media Queries
* Hero Unit
* Thumbnail Gallery

Bootstrap CSS Overview

* Basic Typography
* Lists
* Buttons
* Tables
* Forms
* Images and Icons
* Overriding Bootstrap Core CSS

Navigations Systems

* Adding the basic navigation bar
* Breadcrumbs
* Pagination
* Page for next and previous links
* Using tabs and pills navigation

Working with Components

* Overview of components
* Using components
* Glyphicons
* Dropdowns and buttons
* Navigation components
* Pagination
* Label and badges
* Typography
* Thumbnails
* Alerts
* Progress bars

JavaScript in Bootstrap

* Using JavaScript plugins
* Using transition effects
* Using modal prompts
* Using dropdowns
* Updating nav targets in scrolling
* Tabbing
* Using tooltips, popovers, and alerts
* Using button states or groups
* Collapsing components
* Using carousels
* **Customizing Bootstrap**
* Custom Build
* Overriding with CSS
* LESS

Day 4

Angular 1.x vs Angular 2

Angular Material

#### [Module 1: Material Design](https://www.zeolearn.com/angular-material-training#topic-description-text-curriculum-1)

Introduction

What Is Material?

Elevation and Shadows

Animation

#### [Module 2: Setting up the Development Environment](https://www.zeolearn.com/angular-material-training#topic-description-text-curriculum-2)

Overview

CodePen Community

Installing Angular Material

Setting up TypeScript

Serving the Application Files

Bootstrapping the Application

#### [Module 3: Components and Services](https://www.zeolearn.com/angular-material-training#topic-description-text-curriculum-3)

Introduction

Adaptive Layouts

Side Nav and Toolbar

Icons and Theming

Lists and Autocomplete

Tabs

Cards and the Toast Service

Action Buttons and Speed Dial

Media and the Dialog Service

Bottom Sheets

#### [Module 4: Forms](https://www.zeolearn.com/angular-material-training#topic-description-text-curriculum-4)

Overview

Building Forms

Adding Validation

#### [Module 5: Wrapping Up](https://www.zeolearn.com/angular-material-training#topic-description-text-curriculum-5)

Customization

ARIA Features

Angular 2