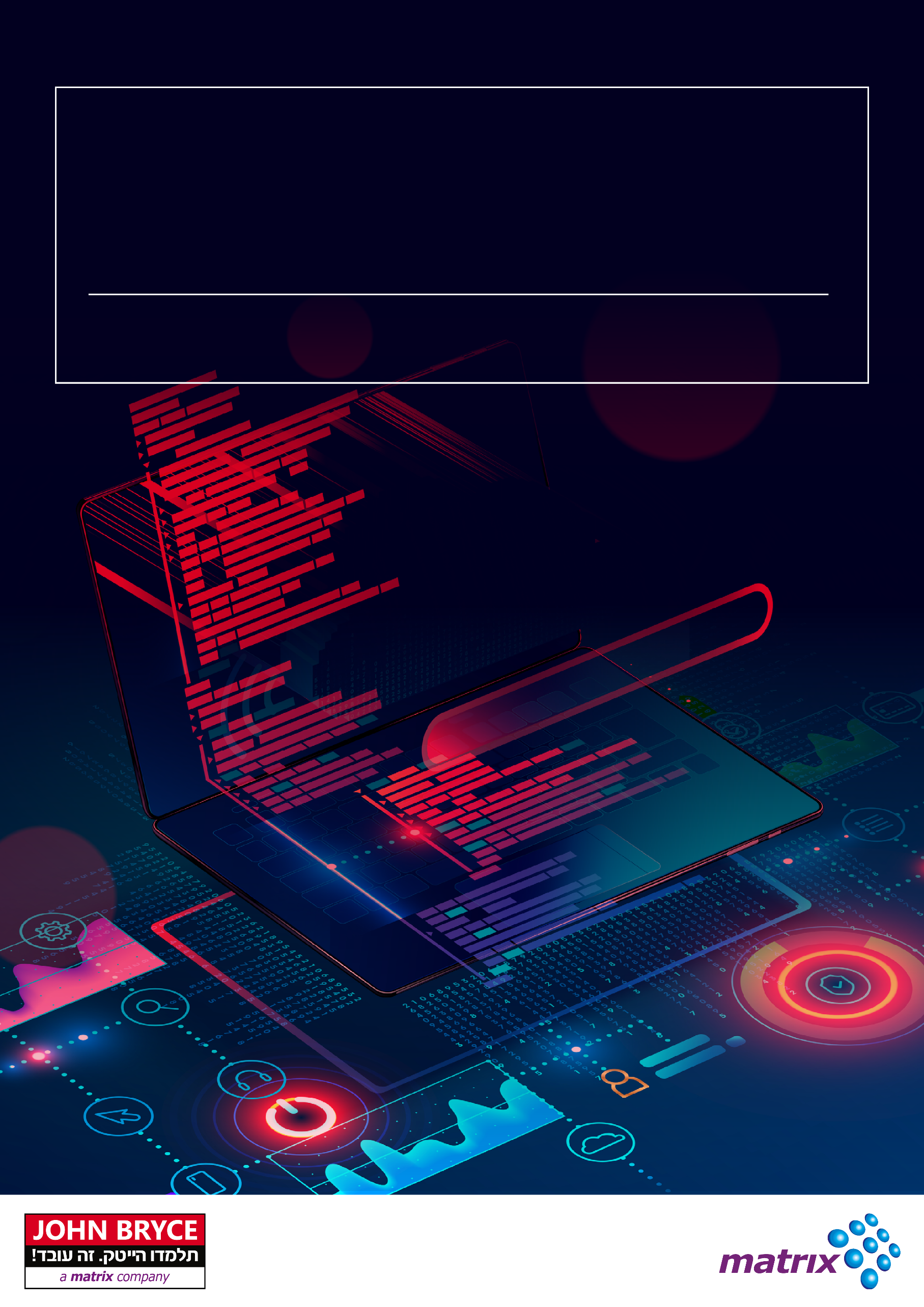
****

**Course 3378 | 40 Hours**

**React with Typescript**

Overview

This course focus on implementing Single Page Application via React technology using Typescript. The course details the SPA architecture and details about the different roles in client MVC design. The course walks through the way of designing and constructing a fully features SPA compliant web modules with React and covers both basics and advanced features.

This course focused on the new React versions and Hooks features combine functional programing with React.

These days, react technology becomes very popular for complex and advanced front end development and becomes a popular long-run solution for web based services.

We will combine an example using open AI to bring the latest innovations from the industry.

On Completion Delegates will be able to

* SPA
* Describe React technology and application structure
* Use React basic components
* Use ECMAScript
* Typescript
* Combine Redux with React
* Use routing and asynchronous messaging
* Use some React advanced features

Who Should Attend?

Web developer that choose React as their solution for constructing Single Page Applications and implement client side.

Prerequisites

* Experience in client-side technologies based on HTML & CSS
* Experience in JavaScript programming
* Some background in REST and Web-Services

Course Contents

**Introduction to SPA**

* SPA Architecture
* Getting ready for future internet clients & IoT
* Overview of supporting technologies (AJAX, REST)
* Vite.js

**JavaScript for react**

* JavaScript latest features
* Using async await
* Functional programing
* Pure functions
* Higher order function
* Shallow copy VS Deep copy
* Understanding the JavaScript Closures
* axios

**Typescript**

* Interfaces
* Type declaration
* Classes & modifiers
* Generics
* Modules
* Type narrow
* Tools

**Introduction to React**

* React Lib concept
* Folder recommendation structure
* Dynamic Imports and file loading
* Component architecture
* The virtual DOM

**React components & Lifecycle**

* Basic components
* State & props
* Stateless components
* Stateful components
* Class component & lifecycle
* Function components
* Class vs function components
* Managing component state
* Pure components
* Hooks
  + useContext
  + useState
  + useEffect
  + useMemo
  + useCallback
  + useDeferredValue
  + useRef
  + useId
* React children

**Introduction to Redux with React**

* Introduction to Redux
  + Redux Flow
  + Redux setup
  + React-redux & redux
  + Redux toolkit
* React terms:
  + Store
  + Action
  + Reducer
  + Middleware
* Combined with React
  + Use store with React
  + Using reducers and actions to manage component state
  + useSelector & useDispatch

**React routing**

* React router
* Single routes
* Multiple routes

**Advanced Redux – Middleware**

* Async Actions
* Async functions
* Redux-thunk

**Error Handling**

* Error boundaries
* componentDidCatch

**Testing react components**

* Jest
* Mocha

**State Management compare ( if time permits )**

* zustand
* jotai



