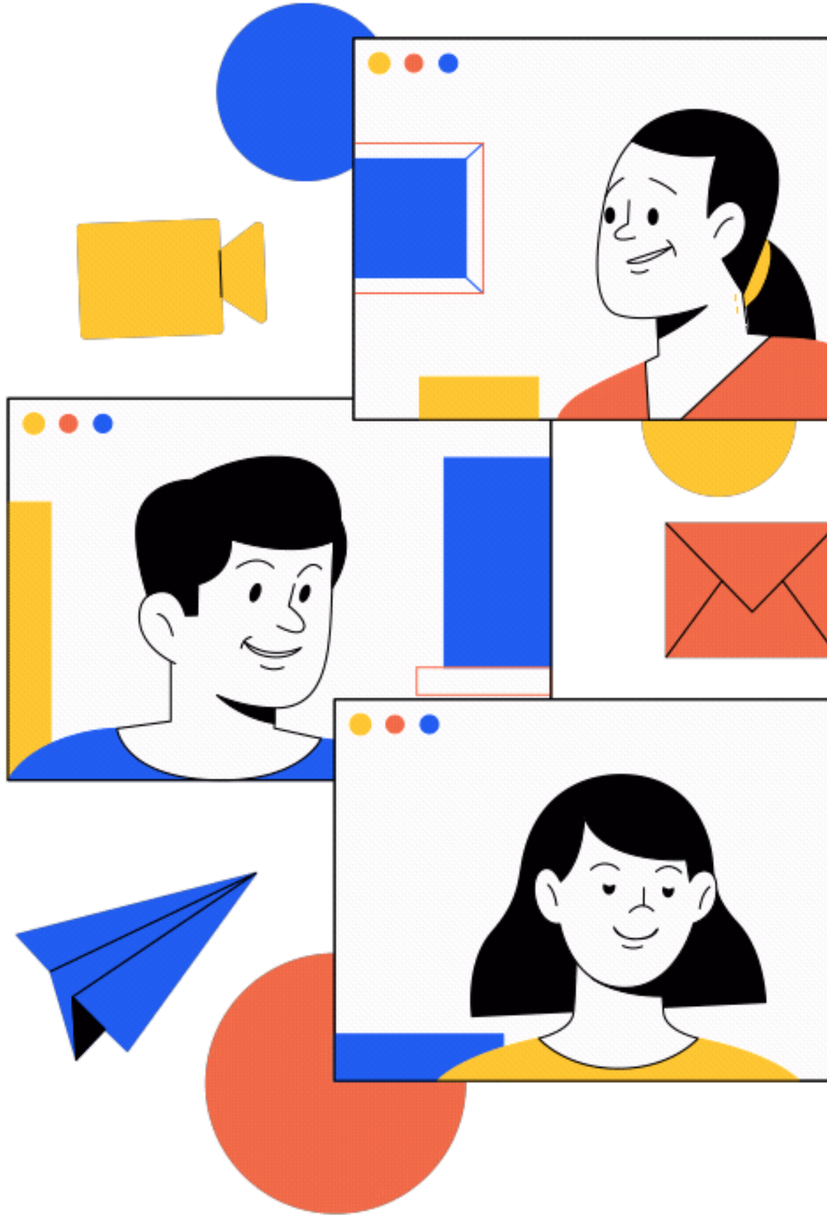
UDACITY

Week 1 - React Dev. Cross-Skilling ND

Welcome onboard



Ahmed Abdelbakey Ghonem

React Session Lead

Agenda

What we'll cover in this session

- Introduction
- Join our Community
- Classroom Overview
- What is React? and Why using React?
- Rendering UI Components with React
- Live Demo

Introduce Yourself

In just 30 second, tell us more about yourself

01 Name

02 Age

03 Major

04 Favorite series, film or cusine

Program Prerequisites

you don't have to be a master

1. HTML/CSS
2. JavaScript

Dos

What students should do to graduate

✅ Attend Weekly Sessions

✅ Make progress in your classroom each week.

✅ Submit the first project before **25 Dec**

Donts

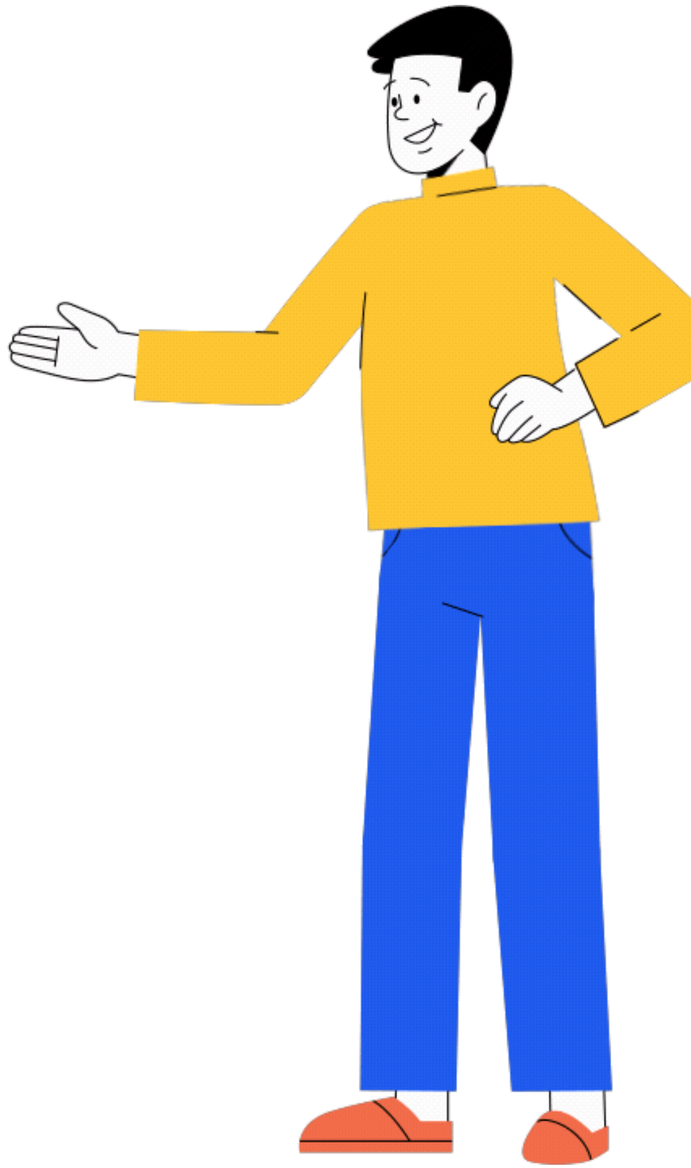
What to avoid

❌ **Plagiarism**

slack

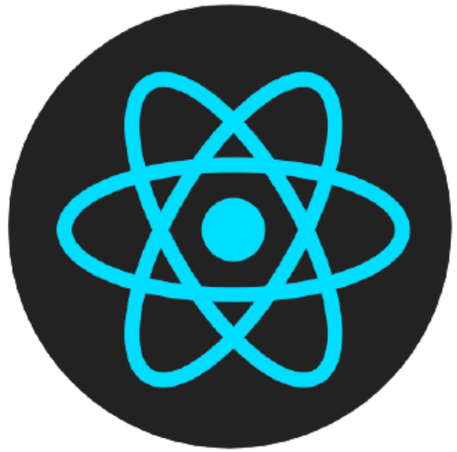
Join our Slack Community

https://join.slack.com/t/egf- wd-t4-dec-21/shared_invite/zt-z4c48u- ub-TQ~S2DdfrS7JZoti50Rb4w



What is React?

- React is a JavaScript library used for building reusable, complex UI components.
- Components are the heart of a React application.
- React is developed and maintained by Facebook and became open-sourced in 2013
- Components can be nested inside each other to build the whole page.



Real (Actual) DOM

- DOM is an acronym that stands for ***Document Object Model.***
- When a web page is loaded, the browser creates a Document Object Model of the page.
- DOM is data representation of the objects that build the structure and content of a document on the web.
- DOM can be used as a programming interface for web documents, it represents the page so that the programs can manipulate it easily.
- DOM Manipulation is very expensive and memory intensive

Virtual DOM

- **VDOM** is a virtual representation of the actual DOM.
- React BTS creates a whole copy of the initial Real DOM and call it virtual DOM.
- If something needs to change the page, the VDOM is updated.
- React matches the actual DOM with its copy (updated VDOM) and detects what has been changed. and then apply these changes to the Real DOM.
- VDOM manipulation is Cheap, Fast and memory efficient

[React and the Virtual DOM]

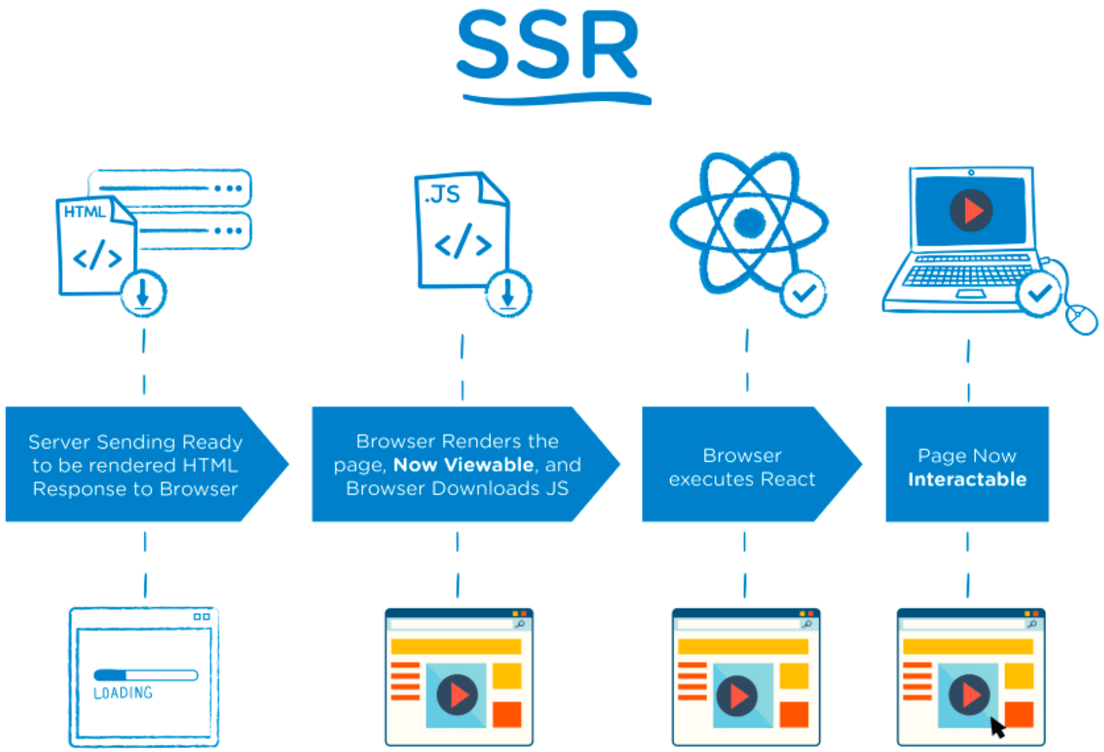
Why using React?

Why choosing React in the first place

- **It uses Virtual DOM**
- **It can be used for SSR or CSR**
- **It follows Uni-Directional Data flow**
- **Composition**
- **The most famous web framework**
- **Uses JSX**

Server-Side Rendering (SSR)

- it is the ability of a web application to render the web page on the ***server*** instead of rendering it in the browser
- SSR is ideal for static sites
- Use SSR when SEO is a priority
- SSR can be used with React using frameworks like [Next.js]



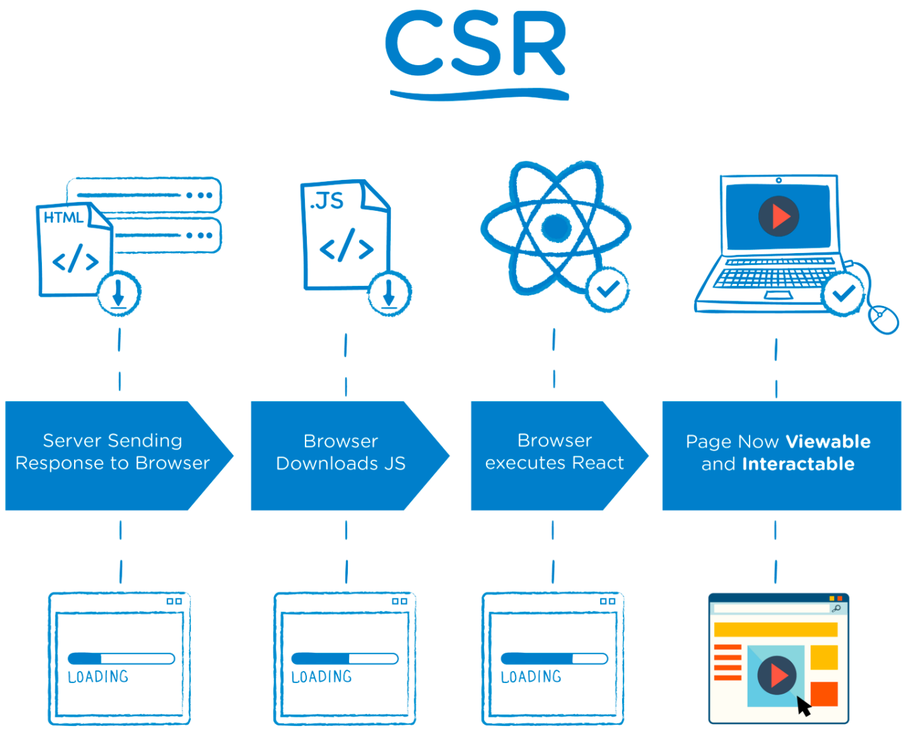
Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

Client-Side Rendering (CSR)

- Empty HTML is served to the browser at first.
- Browser download JS files from the server.
- HTML is then build on the browser and then the page is viewed and interactable.
- Use CSR when SEO is **not** a priority



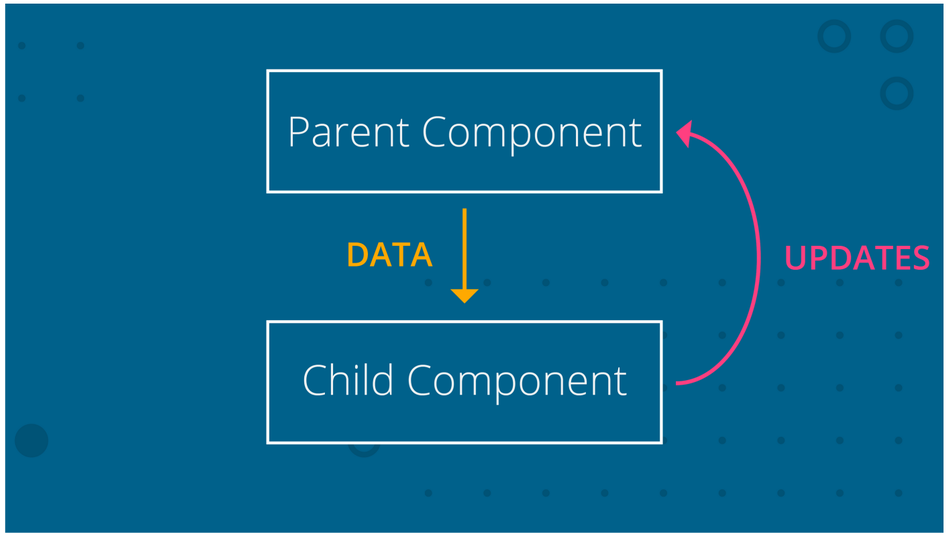
Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

Uni-Directional Data flow

- In React, data flows in only one direction, from parent to child.
- Parent is **the one and only** responsible for updating the data.
- The parent must pass a **callback function** to the child in order to be able to update the data
- If the child wants to render some data maintained by its parent, then the parent must pass it as **props**



Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

Composition

- Composition is a **pattern** that is used to combine simple functions or components to build more complicated ones
- In React, Composition means combining simple components to build complex UI.

```
<Layout>
  <Navbar />
  <MainContent />
  <Footer />
</Layout>
```

- The example above shows how we're combining ***Navbar, MainContent, and Footer*** components to build the Layout.

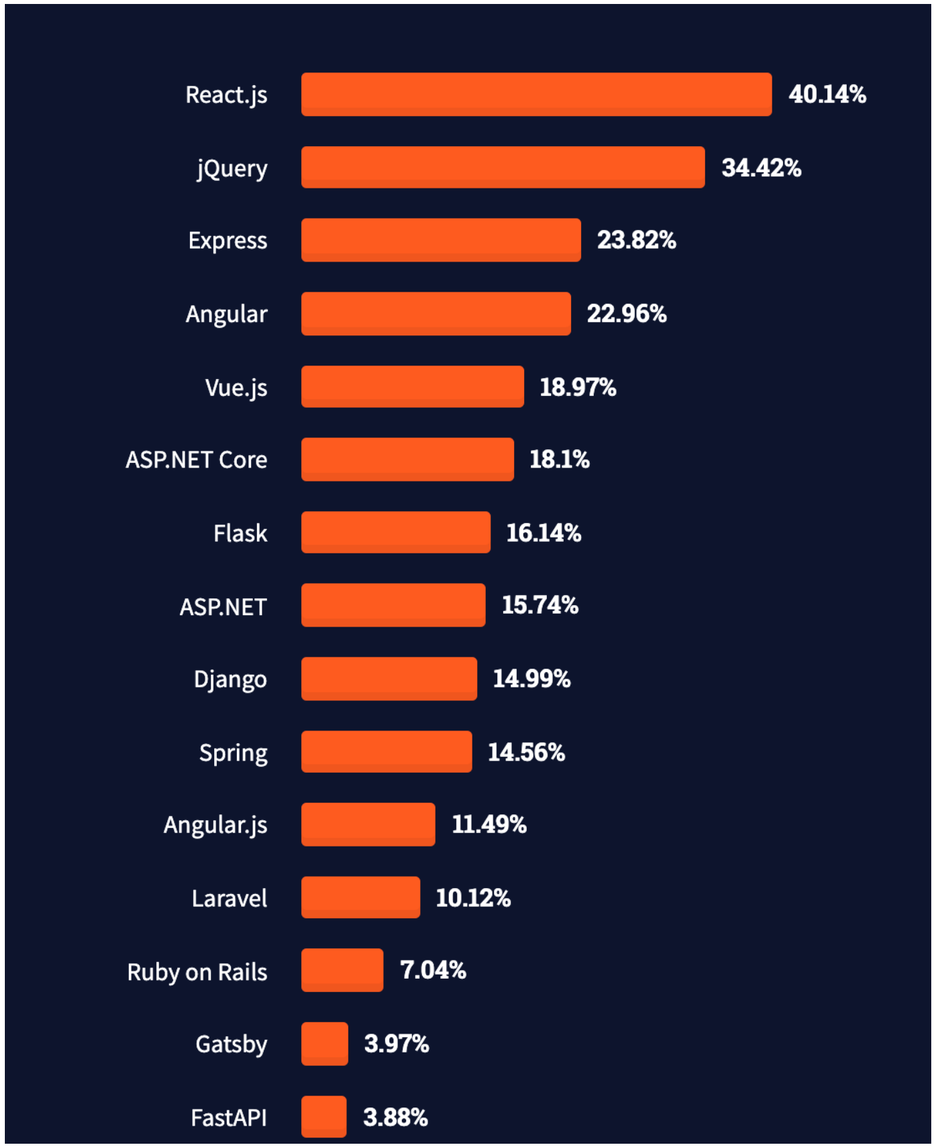
Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

Most Famous Web Framework

according to [Stackoverflow Survey 2021]



Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

JSX (JavaScript HTML)

- JSX is a combination of JavaScript and HTML that allows us to write HTML Tags and elements into javascript files.
- This lets us organize everything into components **<Button />** without using **React.createElement** to build HTML elements.

```
//written with JSX
class Hello extends React.Component {
  render() {
    return <h1>Hello, {this.props.toWhat}!</h1>;
  }
}

ReactDOM.render(
  <Hello toWhat="world" />,
  document.getElementById("root")
);

//written without JSX
class Hello extends React.Component {
  render() {
    return React.createElement("h1", null, `Hello, ${this.props.toWhat}!`);
  }
}

ReactDOM.render(
  React.createElement(Hello, { toWhat: "world" }, null),
  document.getElementById("root")
);
```

Why using React?

Why choosing React in the first place

- It uses Virtual DOM
- It can be used for SSR or CSR
- It follows Uni-Directional Data flow
- Composition
- The most famous web framework
- Uses JSX

Its Demo Time

Takeaways

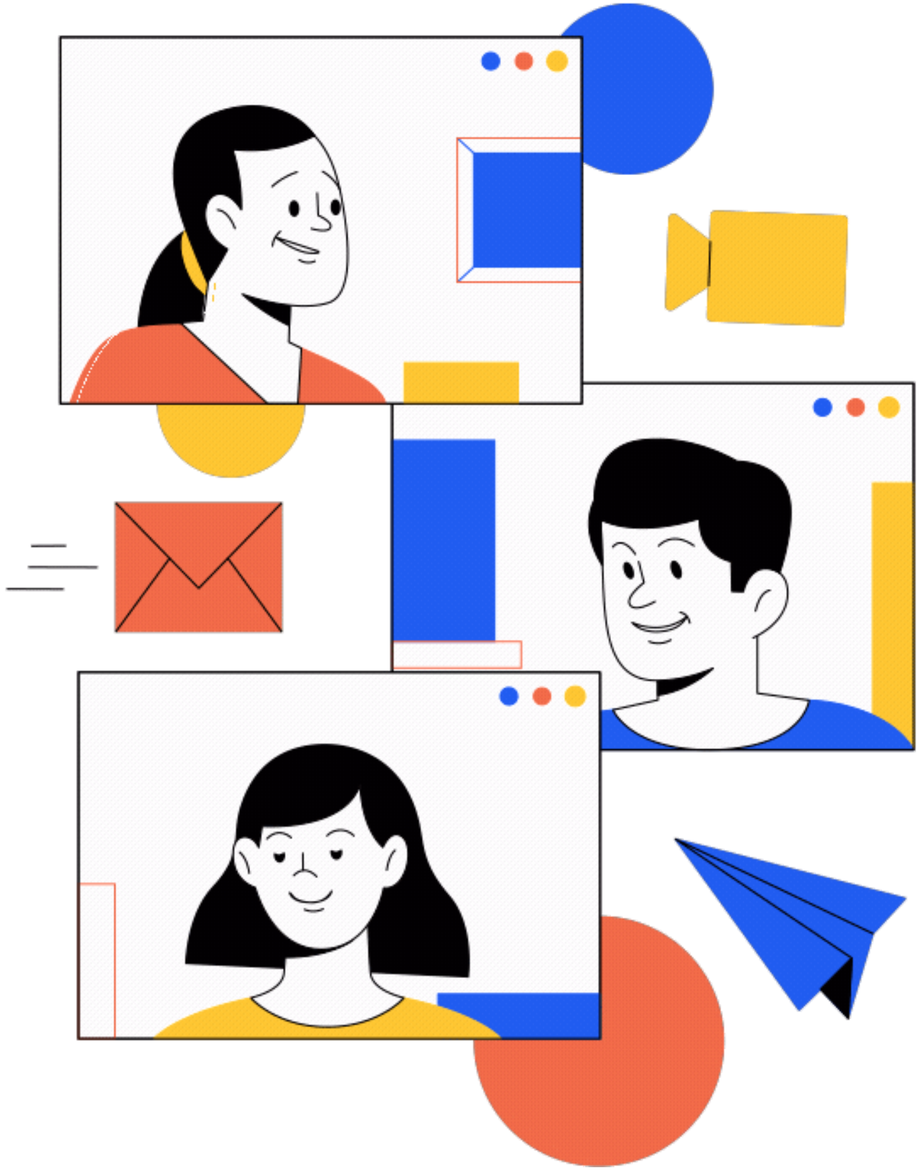
What is React?

Why using React?

What is composition?

React's Uni-Directional Data Flow

Build React project and components



Thank you for attending!

Feel free to email at **aghonem2011@gmail.com** or reach me at slack anytime for any questions or clarifications!

Follow me on Github **@3ba2ii**

code and slides are found at this [github repo]