React -

JavaScript library for building user interfaces. It’s declarative and open-based allowing for painless creation of interactive UIs. Components can be made to manage their own state, then composed to make for a more complex UI. React can also render on the server using Node and power mobile apps using React Native.

React can use standard syntax with Javascript and HTML, but also has support for JSX – an XML-type of syntax. React can be used to create components, and set up applications as well as external plugins for a system.

React Tutorial: https://reactjs.org/tutorial/tutorial.html

Getting Started: https://reactjs.org/docs/getting-started.html

Express.js -

Express.js is a Fast, unopinionated, minimalist web framework for Node.js.

Web Applications

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

APIs

With a myriad of HTTP utility methods and middleware at your disposal, creating a robust API is quick and easy.

Performance

Express provides a thin layer of fundamental web application features, without obscuring Node.js features that you know and love.

Frameworks

Many popular frameworks are based on Express.

Main website: https://expressjs.com/

Express history and changes backlog for a more detailed description of the features: https://github.com/expressjs/express/blob/5.0/History.md

JavaScript -

Javascript is the brain of a webpage. It is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

Main website: https://www.javascript.com/

Tutorial: https://www.w3schools.com/js/

Azure DevOps -

DevOps is an incredibly useful tool for planning in an Agile framework. It holds many features that allow developers to easily collaborate and communicate the progress of their project.

Boards

Boards show all the work items for the project at hand. They can be neatly categorized for easier searching, and contain the framework for sprint planning within them. Sprints can be easily tied to the work items listed in the boards.

Boards also provide a set up for preparing queries for projects, and even contain support for creating a full delivery plan that the whole team can follow as they progress.

Backlogs

Backlogs display the list of all items that need work, are being worked on or are finished in a project. They contain the entirety of the stories and tasks created for sprints. Work items can be linked to different features, allowing them to be stored and easily accessed when necessary.

Sprints

Sprints are displayed on a wide board with all current tasks shown clearly. Each task will have one or many developers assigned to them, and are tracked by moving the task along the progression of the sprint. Tasks start in the Backlog, then are moved into the sprint board as a new task. Once the task is being worked on by the assigned parties, the task can be moved to Active. Finally, once the task has been finished, it is moved into the Completed column. This display of progression lets everyone on the team know where they stand in terms of completing their tasks.

Repositories

Repositories can be created in DevOps to give the team an easily accessible main code base for the project. Using github, repos are created, updated, and displayed to each team member. Branches of the main code base can be created for team members to have their own local version of the project.

Main site: https://azure.microsoft.com/en-us/products/devops/

HTML/CSS -

HTML is the fundamental technology used to define the structure of a webpage.

- Semantic HTML
- SEO best practices

- Main focus is on the content of the product.

HTML is easily assisted with technologies like JavaScript, which enhance its capability. HTML allows websites to be created, and is also the standard markup language for documents designed to be displayed in a web browser.

HTML tutorial: https://www.w3schools.com/html/

CSS defines the HTML document's presentation. CSS can also be used for web-page animations.

- CSS Box model
- CSS specificity
- Flexbox, grid
- Responsive designs
- Psuedo elements

Main focus is more on the visual side of the product.

CSS tutorial: https://www.w3schools.com/css/

Full Stack Development - (paraphrasing from Chris's pdf)

Frontend Development

More focused on the client side of the product. This pertains to the visible part of the website or web application that the user interacts with during their experience of the product. A user will use what's developed on the front end.

There are many technologies that can be used to create a functional product for clients to interact with. Such as HTML, CSS, JavaScript and React.

Backend Development

It refers to the server-side development of web application or website with a primary focus on how the website works. It is responsible for managing the database through queries and APIs by client-side commands.

A powerful technology for backend development is Node.js. It is a runtime environment for JavaScript and by extension, Express.js empowers it further. Express.js is the framework used for Node.js that contains features for the web application that will help in creating a well-supported API. It also makes creating applications in Node.js fast and easy.

Full Stack Development

Full Stack combines both worlds of front end, and back end development. With full stack development, an entire system is created and web applications or websites can be viewed as a whole. It's where all the support for front end and back end technologies merge to create a powerful unified product.