# **Full stack development – Activity 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OVERVIEW** | | | | |
| In this activity, participants learn what full stack development is. The activity begins with a discussion on the definition of full stack development and its three main components (front end, back end, and the database). The next discussion introduces the various responsibilities that a full stack developer has and the benefits of hiring/being a full stack developer. This discussion is followed by a mini activity where participants form groups of 4 (2 front end developers, and 2 back end developers) and select an application idea from the list of provided ideas. Then, for the selected application idea, participants must brainstorm to make a note of the list of required and nice-to-have functionalities that the application would have. For each functionality, they must explicitly mention what tasks would the front end developer have to complete and what tasks would the back end developer have to complete. Next, comes the discussion about the gender disparity in tech, which also includes a specific discussion surrounding gender disparity in front end vs back end. This discussion also sheds light into what steps can be taken to mitigate this disparity with time. The last part of the activity delves into introducing the MERN stack, the different components of MERN stack, and the benefits of learning the MERN stack as a full stack developer.  **Curriculum competencies**  **(Taken from (BC Curriculum) –** [**Applied Design, Skills, and Technologies**](https://curriculum.gov.bc.ca/curriculum/adst)**;** [**Mathematics**](https://curriculum.gov.bc.ca/curriculum/mathematics); [**BC Science**](https://curriculum.gov.bc.ca/curriculum/science)**)** | | | | |
| **Topic** | **Grade Level** | **Cost** | **Time** | **Complexity** |
| - Computer Science | Grades 8-12, all girls | $0.00/students | Intro: 00 min  Project: 00 min | Preparation: 2/5  Execution: 2/5  1-5 (5 = hardest) |
| **Comments from a past instructor:** Put in your pitch about how great this activity was! Or something that should be done differently if someone wants to resurrect it. | | | | |

|  |  |  |
| --- | --- | --- |
| **MATERIALS** | | |
| **Non-Consumable** | **Consumable** | **Other** |
| - Laptop (1 per group of 4 participants)  - 1 Laptop for instructors  \* A laptop with a Notepad app is required to write down ideas for the full stack application. |  |  |
| **PREPARATION** | | |
| N/A | | |

|  |  |
| --- | --- |
| **PROCEDURE** | |
| **10-15 min**  **25 min**  **40-55 min**  **15-25 min** | 1. **Introduce full stack development and its components.** 2. **Discuss the responsibilities of a full stack developer and the benefits of being one.** 3. **Reinforce the concepts of how back end and front end functionalities work together to build features with the mini activity.** 4. **Address the gender gap in tech and in front end vs. back end and discuss ways to mitigate the same.** 5. **Introduce the MERN stack, and discuss its components and benefits.** |
| **Extend** | Ask the groups of 4 to discuss how would they apply the MERN stack to their selected full stack application. |
| **SAFETY** | |
| General internet/computer safety | |
| **TALKING POINTS AND RESOURCES** | |
| **Discussion 1 – What is full stack development?**   * Full stack development refers to the end-to-end application software development, including the front end and back end. The front end consists of the user interface, and the back end takes care of the business logic and application workflows. The main components of a full stack development are the front end and the back end which uses the database. * Consider an online shopping website. Users can browse or purchase specific items, delete or add items in cart, change their profile, and do many other things. All these actions require a front-end user interface (UI), as well as some business logic, written in the back end. * The website UI can be built using various, front-end technologies like HTML, CSS, Javascript and have functionalities such as presenting product listings and allowing users to search and use filters for search.   + **Ask the students:** What are some other places where front-end development is necessary?     - Possible answers: Add products to their cart and view current products in the cart, display promotions, show product details, etc. * The back end, written in programming languages like Java or Python can be used to store purchase history and for user login and authentication.   + **Ask the students:** What are some other places where back end development is necessary?     - Possible answers: Managing and processing orders, inventory management, sending order status updates, implementing algorithms for searching and filtering products based on user input, etc.   + A good web application would need scalability, event handling, and routing, which are usually handled by back end libraries and frameworks like SpringBoot or Django.   + Scalability: In case there is increased user traffic (e.g., Black Friday sales)   + Event handling: For customer service chat, notifications, or live updates on limited-quantity products   + Routing: Mapping URLs so that users can navigate to different parts of the website by specifying URLs etc.   **Discussion 2 – What is a full stack developer?**   * Full stack developers must have knowledge of an entire technology stack, i.e., the set of technologies that are used to build an end-to-end application quickly and efficiently. A technology stack is a combination of software tools, programming languages, frameworks, libraries, and infrastructure components that are used together to build and run a software application. For example, if they want to build an application using the MEAN stack, they should know how to work with MongoDB, Express, Angular, and Node. * Full stack developers should be able to judge whether the selected technologies are the right choice for their project during the early phases. They should also help in choosing the right technologies for the project development and testing both on the front end and the back end. * They should be able to write clean code across the technology stack by following the best practices of the tools used. * They should be up to date with the latest technologies and tools to make the best technology usage decisions. * There are many advantages of hiring full stack developers for web application development:   + Complete ownership and understanding of the project.   + Saves both project time and cost and enhances productivity.   + Faster bug fixing due to knowledge of the complete system.   + Easy knowledge transfer to other team members   + Better division of work among team members   **Activity 1) Role exploration time**  *Purpose: This outlines an interactive exercise instructors can do with students to reinforce the concepts of front end and back end development.*  **Activity Logistics**   * Place students in groups of 4. * Each group has a laptop where they can take notes in the Notepad app.   **Procedure**   * 2 participants in a group act as front end developers and 2 participants act as back end developers. * Each group selects a full stack application to develop from the slide of ideas. * They are given 15 minutes to discuss and make a note of:   + The functionalities that the chosen full stack application should have (required and nice-to-have)   + The role of the front and back end developer for each functionality * After 15 minutes, each group provides a summary of the list of functionalities (required and nice-to-have), and the role of each developer for each functionality.   **Discussion 3 – The gender gap in tech**   * In the general landscape of tech, gender disparity is a very real issue:   + Women made up just 16% of bachelor’s degree recipients in computer and information services, 21% in engineering, 27% in economics, and 38% in physical sciences.   + Women hold less than 20% of tech leadership roles.   + Only 19% of senior vice presidents and 15% of CEOs are women.   + 39% of women in tech see gender bias as a hindrance to getting a promotion.   + 34% of Apple’s employees are female but only 24% of their technical roles are held by women.   Gender disparity in front end vs back end:   * A few years ago, a judge rejected a class-action lawsuit against Google for pay discrimination. The lawsuit was part of a larger case about pay inequality in the tech industry, specifically focusing on whether front-end development, a subfield, is gender-segregated and pays less. The distinction between frontend and backend largely harms women. In the lawsuit, Kelly Ellis, now an ex-employee of Google, said that despite her background in backend development, Google assigned her to be a frontend developer, which had significant consequences for her career. She also said, “Google pays backend engineers more than frontend and fast tracks them for promotion. On the teams I worked with and observed at Google, almost all backend software engineers were men. Almost all female software engineers, however, were frontend engineers.” * The darker subtext of the divide between frontend and backend is that frontend is considered by many to be the “pretty window dressing”, whereas the back end is the “real” development. This is reflected in salaries as well because as front end jobs are considered to be “easier”, front end developers find it harder to get promoted and paid more. Glassdoor reports that the average web developer (a title given to backend developers) makes $87,661, and a frontend developer makes $76,300. Once you are in your career, it gets worse – senior front end developers make an average of $98,560, and regular senior web developers make $113,601. It is also generally recognized that a common practice is to hire women as backend software developers and then only assign them only frontend work, which is known as the “frontend bait and switch”.   Mitigating the disparity:   * It can be daunting to take that first step into tech, especially as a woman, due to the existing disparity. But one of the ways to address this disparity is to work in tech and build a strong network of women who can support each other and help contribute to each other’s career growth. Some things that you can do are:   + Be persistent and resilient: there will be challenges along the way and you might feel discouraged at times but remember to ask for help, keep learning, and continue pursuing your goals.   + Remember that you belong in technology: women are meant to be in tech and every field. Even if you can’t see as many, they’re there and eager for you to join.   + Advocating for yourself: know your worth and ask for promotions, raises, new opportunities, and any other thing you want on the job. You belong in tech and can achieve anything.   + Building your self-confidence: women tend to apply to jobs only where they are sure they meet at least 90% of the requirements; men apply even if they don’t meet them. This could stem back to societal expectations; men are encouraged to take risks and not be afraid of failure, and women are more cautious. So, it is important to allow yourself to fail sometimes when facing uncertain circumstances because only then you will know what you are truly capable of and what you have the potential to become better at.   **Discussion 4 – Building a full stack application with MERN**   * Now, we will focus on how to build a full stack application using the MERN stack. * MERN is one of several variations of the MEAN stack (MongoDB Express Angular Node), where the traditional Angular.js front-end framework is replaced with React.js. * MERN stands for MongoDB, Express, React, Node, after the four key technologies that make up the stack.   + MongoDB — document database   + Express(.js) — Node.js web framework   + React(.js) — a client-side JavaScript framework   + Node(.js) — the premier JavaScript web server * The MERN architecture allows you to easily construct a three-tier architecture (front end, back end, database) entirely using JavaScript and JSON. * The top tier of the MERN stack is React.js, the declarative JavaScript framework for creating dynamic client-side applications in HTML. React lets you build up complex interfaces through simple components, connect them to data on your back-end server, and render them as HTML. * React’s strong suit is handling stateful, data-driven interfaces with minimal code and minimal pain, and it has all the bells and whistles you’d expect from a modern web framework: great support for forms, error handling, events, lists, and more. * The next level down is the Express.js server-side framework, running inside a Node.js server. Express.js bills itself as a “fast, unopinionated, minimalist web framework for Node.js,” and that is indeed exactly what it is. Express.js has powerful models for URL routing (matching an incoming URL with a server function), and handling HTTP requests and responses. * By making XML HTTP Requests (XHRs) or GETs or POSTs from your React.js front end, you can connect to Express.js functions that power your application. Those functions, in turn, use MongoDB’s Node.js drivers, either via callbacks or using promises, to access and update data in your MongoDB database. * If your application stores any data (user profiles, content, comments, uploads, events, etc.), then you’re going to want a database that’s just as easy to work with as React, Express, and Node. And that’s where MongoDB comes in: JSON documents created in your React.js front end can be sent to the Express.js server, where they can be processed and (assuming they’re valid) stored directly in MongoDB for later retrieval.   Why choose a MERN stack:   * Let’s start with MongoDB, the document database at the root of the MERN stack. MongoDB was designed to store JSON data natively (it technically uses a binary version of JSON called * BSON), and everything from its command line interface to its query language (MQL, or MongoDB Query Language) is built on JSON and JavaScript. * MongoDB works extremely well with Node.js, and makes storing, manipulating, and representing JSON data at every tier of your application incredibly easy. * For cloud-native applications, MongoDB Atlas makes it even easier, by giving you an auto-scaling MongoDB cluster on the cloud provider of your choice, as easy as a few button clicks. * Express.js (running on Node.js) and React.js make the JavaScript/JSON application MERN full stack, well, full. Express.js is a server-side application framework that wraps HTTP requests and responses and makes it easy to map URLs to server-side functions. React.js is a front end JavaScript framework for building interactive user interfaces in HTML and communicating with a remote server. * The combination means that JSON data flows naturally from front to back, making it fast to build on and reasonably simple to debug. Plus, you only have to know one programming language, and the JSON document structure, to understand the whole system! * MERN is the stack of choice for today’s web developers looking to move quickly, particularly for those with React.js experience. | |
| **MODIFICATIONS AND ADAPTATIONS** | |
| This activity can be a little harder for those who have a hard time working with partners. Try to have the participants join into groups with their friends if they have them in the club or keep a close eye on those who have a harder time in a group setting. | |
| **BACKGROUND INFORMATION** | |
| * + N/A | |

# 

# **Last Modified - 03/10/23**