**DUDE SESHU**

**Development Methodologies**

**Domain-Driven Design (DDD):** Imagine building software for a library. Through discussions with librarians (the domain experts), you discover key concepts like "Book," "Loan," "Patron," and "Reservation." You also learn about important business rules, such as how many books a patron can borrow, loan periods for different types of materials, and the process for handling overdue items. Your software's design would then directly mirror these concepts and rules. For instance, you might have software components specifically representing a "Book" with its attributes (title, author, ISBN) and behaviors (being loaned, being returned). The language you use when talking about the software with the librarians would be the same terms used in the system's design.

**Feature-Driven Development (FDD):** Consider developing an online shopping cart. One specific feature identified is "Adding an item to the cart." The team would focus solely on implementing this functionality. This involves designing the user interface elements (like an "Add to Cart" button), the underlying logic to store the selected item and quantity, and updating the cart display. Once this feature is complete and tested, the team would move on to the next feature, like "Viewing the cart" or "Proceeding to checkout."

**Test-Driven Development (TDD):** Think about creating a function that validates if an email address is in the correct format. Before writing the actual code to perform the validation, you would first define what "correct format" means by writing down specific test cases. For example, you might write down: "An email address should contain an '@' symbol," "An email address should have a domain name after the '@'," and "Spaces are not allowed in an email address." These test cases act as your initial requirements. Only then would you write the email validation function, constantly running your tests to ensure the code you write satisfies all the defined criteria.

**Behavior-Driven Development (BDD):** Let's take the example of a website's login process. Using BDD, you might describe the desired behavior like this:  
  
 **Scenario:** Successful login  
  
 **Given** the user is on the login page **And** the user enters a valid username "john.doe" **And** the user enters the correct password "password123" **When** the user clicks the "Login" button **Then** the user should be redirected to the dashboard **And** a welcome message "Welcome, John Doe!" should be displayed.