Imagine that you are a Team Lead on one of the teams that work on a new e-commerce solution for the client. You are responsible for the creation of a new site. The site has the following very high-level description for now: On the main page, you should see all products (50 per page). You should be able to filter products by brand and category, search options should be available as well. By clicking on the product, you should see its details and a list of similar products. You should be able to add a product to the shopping cart from the detail page or the main page. You should see all the products in the shopping cart and the total amount of money. After successful payment user should get a confirmation email with order details.

**Tasks**

1. Split all work on EPICs
2. Estimate all Epics using the T-shirt sizes technique
3. Create User Stories for each EPIC
4. Estimate all User Stories with proper approach considering you have a new team
5. Estimate all User Stories with proper approach considering you have established team
6. Estimate how many Sprints your team may need to deliver the work, explain why – 10 points

Split all work on EPICs

1. **Product catalog and search**

*This EPIC focuses on creating and managing the product catalog, including listing, filtering, and searching for product*

1. **Product Detail Page**

*This EPIC involves displaying detailed information about each product and showing similar products.*

1. **Shopping Cart Management**

*This EPIC covers all functionalities related to the shopping cart, including adding, updating, and removing items*

1. **Payment Process**

*This EPIC includes the entire checkout process, from user information collection to payment*

1. **User Account and Authentication**

*This EPIC handles user authentication and account management features.*

1. **Testing**

*This EPIC includes all activities related to testing and ensuring the quality of the application.*

Estimate all Epics using the T-shirt sizes technique

1. **Product catalog and search**

I’d say thatit should be **L** (Large). This Epic includes listening and filtering. Listening is relatively simple task. You have to just get this data from backend and present this data in UI. But filtering is pretty complex task. Different kind of filters should be implemented.

1. **Product Detail Page**

It’s not difficult a task at all. The data have to be retrieved from the backend and present in UI somehow. It should be **M** (Medium)

1. **Shopping Cart Management**

The same as the previous one. This Epic includes different kind of activities with shopping cart like adding, updating and etc. It should be S (Small)

1. **Payment Process**

It’s a complex task includes activities related to integration with an external payment service, handling sensitive data securely. **XL** (Extra Large)

1. **User Account and Authentication**

It’s a complex task as well includes integrating with Authentication and Authentication provider.

**XL** (Extra Large)

1. **Testing**

It should be **L** (Large) need to cover all possible cases to make sure that the app is done and ready to deliver

Create User Stories for each EPIC

1. **Product catalog and search**
   1. User Story 1: Implement product listing with pagination (50 products per page).
   2. User Story 2: Develop filter functionality to filter products by brand.
   3. User Story 3: Develop filter functionality to filter products by category.
   4. User Story 4: Implement search functionality to find products by name or keywords.
2. **Product Detail Page**
   1. User Story 1: Create a product detail page to display detailed information (images, description, price, reviews).
   2. User Story 2: Implement functionality to show similar or related products on the product detail page.
3. **Shopping Cart Management**
   1. User Story 1: Implement "Add to Cart" functionality from the main product listing page.
   2. User Story 2: Implement functionality to update the quantity of products in the cart.
   3. User Story 3: Develop the ability to remove items from the shopping cart.
   4. User Story 4: Display the total amount for all items in the shopping cart.
4. **Payment Process**
   1. User Story 1: Develop a checkout page to *capture* user details
   2. User Story 2: Integrate payment gateway to securely process payments.
   3. User Story 3: Implement functionality to display a confirmation message after successful payment.
5. **User Account and Authentication**
   1. User Story 1: Implement user registration and login functionality.
   2. User Story 2: Develop a user profile page where users can view and edit their personal details.
   3. User Story 3: Implement password recovery and reset functionality.
6. **Testing** 
   1. User Story 1: Develop unit tests for components and services.
   2. User Story 2: Implement integration tests for end-to-end workflows.
   3. User Story 3: Perform security and performance testing to ensure application robustness.

Estimate all User Stories with proper approach considering you have a new team

***Product Catalog and Search***

* **User Story 1**: Implement product listing with pagination (50 products per page).  
  **M (**Medium**)** *This requires fetching data from the backend, implementing pagination on the frontend, The team might need some time to understand the backend data structure and how to efficiently implement pagination.*
* ***User Story 2****: Develop filter functionality to filter products by brand.***M** (Medium)

*The complexity is moderate because it involves understanding both backend and frontend aspects.*

* ***User Story 3****: Develop filter functionality to filter products by category.***M** (Medium)

*Like filtering by brand*

* ***User Story 4****: Implement search functionality to find products by name or keywords.***L (**Large**)**

*This could be more complex due to needing a more sophisticated backend search mechanism, possibly involving text indexing or integrating a search service. The team is new and may need time to get familiar with search optimization techniques.*

***Product Detail Page***

* ***User Story 1****: Create a product detail page to display detailed information (images, description, price, reviews).***M** (Medium) *This involves designing the product detail page and making backend calls to fetch detailed information.*
* ***User Story 2****: Implement functionality to show similar or related products on the product detail page.***L *(***Large***)***

*Requires additional logic to determine "similar" products, which could involve machine learning algorithms or rule-based logic.*

***Shopping Cart Management***

* ***User Story 1****: Implement "Add to Cart" functionality from the main product listing page.***S** (Small)

*The complexity is low to moderate.*

* ***User Story 2****: Implement functionality to update the quantity of products in the cart.****S (***Small***)***

*This task requires additional backend calls and UI updates. The team should handle edge cases like maximum quantity limits.*

* ***User Story 3****: Develop the ability to remove items from the shopping cart.****S (***Small***)*** *This requires a backend request to remove an item and updating the UI accordingly.*
* ***User Story 4****: Display the total amount for all items in the shopping cart.****XS (***Extra Small)

*Calculating the total amount is a simple task involving summing up item subtotals. It's more of a frontend task with minimal backend changes.*

***Payment Process***

* ***User Story 1****:* Develop a checkout page to capture user details
* ***M (***Medium***)******Reason****: This involves creating a form for user input and handling form validation and data submission. The team needs to ensure that the data is securely transmitted to the backend.*
* ***User Story 2****:* Integrate payment gateway to securely process payments.  
  ***XL*** (Extra Large)

*Payment integration is complex, requiring third-party API integration, security considerations (e.g., PCI compliance), and thorough testing. Given the team's inexperience, this could take longer.*

* ***User Story 3****: Implement functionality to display a confirmation message after successful payment.***S** (Small)

*After payment is processed, the system needs to display a confirmation message. This is mostly a frontend task, with some backend support to confirm the transaction status.*

***User Account and Authentication***

* ***User Story 1****: Implement user registration and login functionality.***L** (Large)

*This involves developing user registration forms, login mechanisms, password hashing, and session management.*

* ***User Story 2****: Develop a user profile page where users can view and edit their personal details.***M** (Medium)

*A user profile page needs to display user data and allow for updates. It involves both backend and frontend work.*

* ***User Story 3****: Implement password recovery and reset functionality.***M** (Medium)

*The task requires secure handling of password reset requests, including generating and verifying tokens.*

**Testing**

* **User Story 1**: Develop unit tests for components and services.

**S** (Small)

Unit tests are small pretty easy test. I can’t say that it calls for extra effort

* **User Story 2**: Implement integration tests for end-to-end workflows.

L ( Large)

This kind of tests call for extra efforts. The team have to figure out how to work the system

* **User Story 3**: Perform security and performance testing to ensure application robustness.

L (Large)

The same as the prev one.

Estimate all User Stories with proper approach considering you have established team

When estimating tasks with an established team as opposed to a new team, several factors could influence the estimates. An established team is typically more familiar with the codebase, has better communication and collaboration practices, and has a clearer understanding of the project's domain, tools, and technologies. This often results in more accurate and generally **lower estimates** for the same tasks.

Established teams are more confident in their estimates, reducing the risk of unexpected challenges.

Estimate how many Sprints your team may need to deliver the work, explain why.

**Assumptions**

**Team Size and Roles**:

Assume a team of 5 members: 2 developers, 1 QA engineer, 1 UX/UI designer, and 1 Scrum Master/Product Owner.

**Sprint Length**:

2-week sprints are typical for Agile teams.

**T-Shirt Sizes to Story Points Conversion**:

* XS (Extra Small) = 1 point
* S (Small) = 3 points
* M (Medium) = 5 points
* L (Large) = 8 points
* XL (Extra Large) = 13 points

**Team Velocity**:

* For an established team, assume a velocity of 25 story points per sprint.
* For a new team, assume a velocity of 15 story points per sprint due to potential inefficiencies.

**Product Catalog and Search**

1. Product listing with pagination (M) - 5 points
2. Filter products by brand (M) - 5 points
3. Filter products by category (M) - 5 points
4. Search functionality (L) - 8 point  
   **Total Points**: **18 points**

**EPIC: Product Detail Page**

1. Product detail page (M) - 5 points
2. Show similar/related products (L) - 8 points  
   **Total Points**: **13 points**

**EPIC: Shopping Cart Management**

1. Add to cart from the main page (S) - 3 point
2. Update quantity in cart (S) -3 point
3. Remove items from cart (S) - 3 point
4. Display total amount in the cart (XS) - 1 point  
   **Total Points**: **10 points**

**EPIC: Checkout and Payment Process**

1. Checkout page (M) - 5 points
2. Payment gateway integration (XL) - 13 points
3. Confirmation message after payment (S) - 3 point  
   **Total Points**: **21 points**

**EPIC: User Account and Authentication**

1. User registration and login (L) - 8 points
2. User profile page (M) - 5 points
3. Password recovery/reset (M) - 5 points  
   **Total Points**: **18 points**

**Testing**

1. User Story 1: Develop unit tests for components and services. (S) - 3 points
2. User Story 2: Implement integration tests for end-to-end workflows. (L) - 8 points
3. User Story 3: Perform security and performance testing to ensure application robustness. (L) - 8 points

**Total Points**: **19 points**

**Total Story Points**: 99 points

For an established Team: 99/25=3.96 - (4 sprints)

For a New Team: 99/15=6.6 - (7 sptints )