TUTORIAL 4

4.Analyze a project and list modules that are best suited for automation testing. Objective:

To help students evaluate a real-world software project and identify modules where automation testing will provide maximum benefit in terms of efficiency, coverage, and accuracy.

Project Used (Sample):

Online Shopping Web Application
Features include:

- User login/registration
- Product search and filters
- Shopping cart and checkout
- Order history
- Admin product management

(This can be a local project or from GitHub. Recommended: https://github.com/pranavmalvawala/Simple-Ecommerce-PHP or any Flask/Django/Node-based sample project)

Tools Required:

- Browser (Chrome/Firefox)
- Local Server (XAMPP/WAMP for PHP, Flask for Python)
- Spreadsheet (Excel or Google Sheets)
- Pen & Paper or Document editor for observations
- Optional: Selenium IDE/WebDriver (for future testing)

Step-by-Step Instructions for Students:

🔹 Step 1: Understand the Application

- Launch the project on localhost.
- Explore all features as an end user.
- Navigate through each page and list its functionality.

Example:

Page Name	Functionality
Login Page	User login with email/password
Registration Page	New user signup
Home Page	Displays all products
Product Details	Shows product info, add to cart button
Cart Page	View/edit/remove products in cart
Checkout	Address input, payment simulation
Order History	List of past orders
Admin Panel	Add/edit/delete product and view orders

🔹 Step 2: Identify All Functional Modules

Based on your navigation, note down key functional modules and what they do.

Module	Description
User Login	Authenticates user credentials
User Registration	Validates and stores new user data
Product Search	Allows user to filter products by name/category
Add to Cart	Allows adding product with quantity

Checkout	Captures address, confirms order
Order Summary	Displays list of ordered items
Admin Product CRUD	Create, read, update, delete products

🔹 Step 3: Analyze Each Module for Automation Suitability

Check each module using these criteria:

- 💡 Is the feature **repetitive** (used often)?
- 💡 Does the feature use **form inputs or validations**?
- 💡 Are results **predictable and consistent**?
- 🚫 Does it involve **CAPTCHA**, **third-party tools**, or **image uploads**? (These are **not ideal** for automation)

🔹 Step 4: Create an Automation Suitability Table

Sr.	Module	Description	Suitable for Automation	Reason
1	Login	User login form with validations	✅ Yes	Repetitive, predictable
2	Registration	Email and password form	✅ Yes	Form-based, good for field testing
3	Product Search	Keyword/category search	✅ Yes	Result matching and filters easy to automate
4	Add to Cart	Add, remove, quantity update	✅ Yes	Data-driven actions, good for repeat tests
5	Checkout	Address input and order summary	✅ Yes	End-to-end flow check with static input
6	Order History	Displays previous orders	✅ Yes	Display checks, test with known data
7	Admin Product CRUD	Admin adds/edits/deletes products	✅ Yes	Form testing, DB change validation

8	Theme Layout	Color, font, responsiveness	❌ No	Requires human eyes, not logic-based
9	CAPTCHA (if present)	Image verification	❌ No	Needs manual check
10	Payment Gateway	Simulated or real payment	❌ No	Sensitive, 3rd party, sandbox required

🔹 Step 5: Justify Your Choices

Provide 2–3 sentences **justifying why you chose** a module for automation.

Example:

> "Login Module is best suited for automation because it is used in almost every session. Valid and invalid login cases can be easily automated using Selenium, and results are deterministic (either success or error)."

🔹 Step 6: Document the Output

Prepare the following in your lab report:

- Title Page
- Objective
- Application Overview
- Full List of Modules
- Automation Suitability Table
- Justifications for at least 3 modules
- Conclusion: Why automation helps and what tools you'd use (Selenium, TestNG, etc.)

Sample Conclusion:

> After analyzing the online shopping application, we found that modules like Login, Registration, and Product Search are ideal for automation because of their repetitive use and structured input/output. Automation will save time and reduce manual effort in regression testing.

Submission Guidelines:

Each student/group must submit:

- 📄 A detailed report in Word/PDF
- 📊 Excel sheet with module analysis
- 🖼️ Screenshots of the project interface
- 🎯 Optional: One automated test using Selenium