

I. Create a brief test plan based on 1 of the following 2 scenarios:

1) Scenario 1: You are assigned to test a new e-commerce website. The website allows users to browse products, add items to their cart, and make purchases online.

Test Plan: E-commerce Website

2) Scenario 2: You are responsible for testing a new mobile app designed for task management. The app allows users to create tasks, set priorities, and receive notifications.

Test Plan: Mobile App Testing

II. Practice exercises for creating checklists

Write a checklist for a project of part I (can write some checklists for some functions in the project).

You can use any tool to write a project checklist or you can use the tool I introduce below. Below are suggestions for your reference.

=====END=====

I. Test Plan Template

<Company Name> [Group Name]

<Project Name>	Version: <1.0>
Test Plan	Date: <dd/mm/yy>
Prepared by: (Names of Preparers)	

Revision History

Date	Version	Author	Description

1. Introduction

- Purpose: To ensure the functionality, usability, and security of the e-commerce website.
- Scope: Testing will cover product browsing, cart management, and online transactions.

2. Test Strategy

- Testing Levels: Functional, usability, security.
- Testing Types: System, acceptance.
- Test Environment: Web browsers (Chrome, Firefox, Safari), different devices.

3. Test Planning

- Test Deliverables:
 - + Test cases
 - + test data
 - + defect reports.
- Test Schedule: Start date - MM/DD/YYYY, End date - MM/DD/YYYY.
- Resource Planning: Testers, developers, and product owners.

4. Test Design

- Test Scenarios: User login, product search, add to cart, payment processing.
- Test Cases: Detailed cases for login, search, cart operations, and transactions.
- Test Data: Valid and invalid user credentials, product data, and payment methods.

5. Test Execution

- Test Execution Schedule: Phased approach, starting with basic functionality.
- Test Execution Environment: Various browsers and devices.

- Defect Reporting: Use a bug tracking system, and categorize by severity.

6. Test Closure:

- Criteria for Test Completion: All critical and high-priority test cases executed without major defects.
- Test Summary Report: Summary of test results, issues, and recommendations.

7. Risks and Contingencies

- Identify risks: Server downtime, payment gateway issues.
- Contingency Plans: Regular backups, alternative payment methods.

8. Approval:

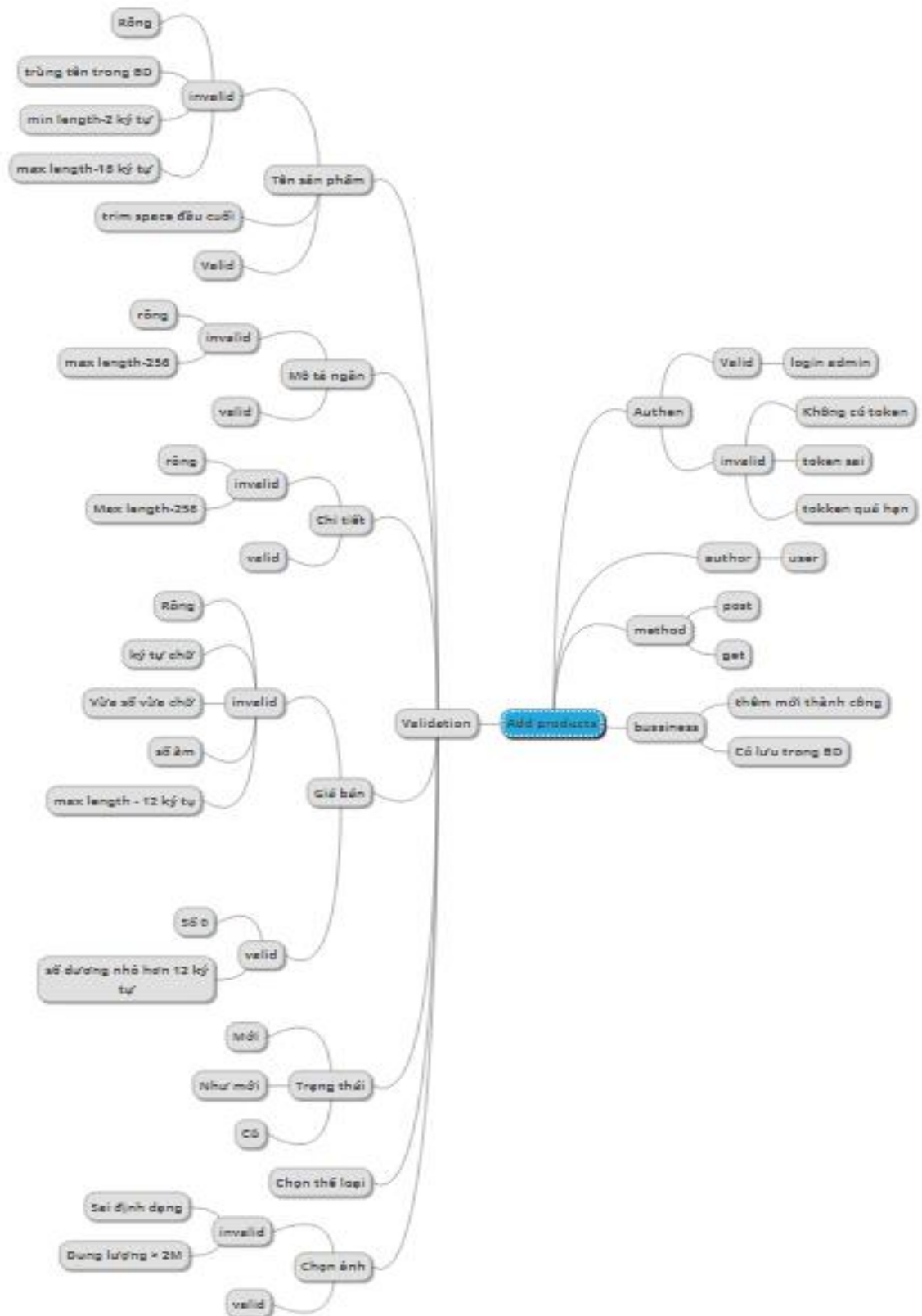
Approvals: QA Lead, Project Manager.

9. Appendices:

- Glossary: Key terms related to e-commerce.
- References: User manuals, technical documentation.

II. You can use this tool to create checklists

<https://app.mindmup.com/map/new/1696494388594>



1. Checklist - is a list of tests that should be run in a definite procedure. It helps to understand if testing is fully run and how many failed. It also helps formalize testing separately taken functionality, putting tests in a list. Test order in the checklist may be strict as well as random.

2. When to create checklists? The most common reason to create a checklist is not enough time, the project requires testing in a short time.

3. Criteria for creating checklists

- Define the purpose of the checklist
- Define the list of test requirements
- Define test steps
- Sort requests in order of priority
- Check the checklist: Check the requirements on the checklist to make sure that the check steps are correct.

4. Example of some Checklists for a web

a. Functional Testing

Sample use cases to be considered

- Validate signup, sign-in, login flow, and functionality.
- Check app response during push notification.
- Ensure calls may be received/declined while the app is running.
- Check app response after intrusion.
- Measure power consumption.

b. Usability Testing

Sample use cases to be considered

- Create a sitemap and ensure ease of navigation.
- Ensure the web application's design is user-friendly.
- Ensure the app provides confidentiality when asking for crucial personal information.
- Ensure the app serves a diverse audience, both geographically and physically.
- Look for breadcrumbs in the app.
- Validate the app menu to ensure that it's easy to use
- Validate app rendering across devices and OS.

c. Compatibility Testing

Sample use cases to be considered

- Test buttons and links across devices and platforms.
- Test for any resolution, image, application cache, and performance issues across devices.
- Test for text alignment, dropdown, and issues with alignment of other elements.
- Test for web app's scalability to fit according to size, tables, and borders.
- Verify fields and their inputs.
- Check app behavior when autocomplete is on or off.
- Check app background color across all screens.

d. GUI Testing

Sample use cases to be considered

- Test the key GUI elements such as size, position, length, etc.
- Ensure the error messages are displayed correctly.
- Ensure the fonts are readable and aligned.
- Test the quality of the images used.
- Check the positioning of GUI elements for different screens.

e. Cross-browser Testing

Sample use cases to be considered

- Ensure the app loads on all browsers.
- Ensure the elements (such as buttons, forms, and menus) are visible.
- Test the clickable elements on the app page.
- Ensure the page content is visible on all devices.
- Measure the time taken to load the webpage on different browsers.

g. Database Testing

Sample use cases to be considered

- Ensure the data is logically organized.
- Ensure the heterogeneous databases are consistent with the overall application mapping.
- Test the length and naming convention of the database fields.
- Check the size and length of the required fields and indexes.
- Check the database server configurations as per the business requirements.

h. Performance Testing

Sample use cases to be considered

- Test app loading time.
- Test app performance over multiple devices and OS.
- Test battery consumption.

- Test network scalability.
- Test memory and CPU utilization.
- Test app performance under varying loads.
- Test app behavior during network fluctuations.

k. Security Testing

Sample use cases to be considered

- Ensure the app can handle SQL injections.
- Ensure the cookies do not store passwords.
- Ensure the session expires when the user clears the cache.
- Ensure the user is asked to re-login when the session expires.
- Test user roles.
- Test all third-party integrations and vulnerable files.
- Identify functional loopholes.
- Validate exposed sensitive data.
- Identify server-related vulnerabilities.
- Validate the process of data management and processing.
- Perform pen tests and red-team intervention.
- Test the verification capability of the app.

l. Accessibility Testing

Sample use cases to be considered

- Ensure the app design and layout is consistent.
- Ensure the text size is apt as per the end user's phone settings.
- Test the screen orientation after rotation.
- Ensure the informative text has accessible alternatives.
- Ensure the screen readers can read the headings.
- Test the non-text features like photos, diagrams, graphs, audio/video content, etc.
- Test the transcripts for the audio and video elements.
- Test the audio controls of the app.
- Test the contrast of app elements.
- Check whether the app displays the default language.

5. Checklist for some basic items on the website

a. Dropdown

- When to enable/disable

- When is it obligatory/optional to choose data
- What is the default value
- Values in the dropdown are sorted ascending or descending
- You can select a value in the dropdown by typing some characters and then pressing enter
- Multiple values can be selected at the same time if the dropdown allows

b. Button

- When to enable/disable
- What happens if click the button

c. Textbox

- When to enable/disabled
- Is it mandatory to enter data?
- What is the default value when opening the page?
- What is the valid/invalid input value?
- Format of input value
- Maximum length of input value

d. Checkbox

- When to use enabled/disabled
- Default value of checked/unchecked

e. Paging

- When to appear/not to appear paging
- What happens when clicking on the page numbers

g. Menu

- What items does the menu include?
- What happens when you click on an item on the menu

h. Hyperlink

- What happens when hover and click on hyperlink

i. Upload dialog file/image

- Can you upload all allowed file formats?
- Is the display correct after uploading?

l. UI

- Does the UI break when zooming in (150%)/zooming out (75%)
- Is the UI correct font/size/color/effect/position of items relative to the design?
- Does the UI break when opening in different browsers