

Software Test Plan - STP

“Hackampus.com”

Current Version

Title	Software Test Plan - STP
File	https://www.hackampus.com/
Author	Tal Guez , Ofir Vahav , Peleg Vorsht , Efrat Adler , Jaber Shaker , Wafa Saad
Version	---
Version Date	7/1/2024

1 Document Overview

1.1 Introduction

This document serves as the Software Test Plan for the "Hackampus.com" website.

The purpose of this STP is to define the framework and Strategy for the testing of "Hackampus.com" website.

The plan is tailored to support the Agile Scrum methodology, emphasizing on flexibility, and team work development.

Our objective is to validate the High Quality of "Hackampus.com" website.

We will verify that the "Hackampus.com" website behaves as expected by testing its features and functionality.

In alignment with Scrum principles, this document will try to stay as short and focused on testing needs so it could be easily updated and evolve throughout project iterations.

1.2 Objectives

At a high level The primary objectives of this Software Test Plan for hackampus.com are as follows:

- Ensure Product Quality:

To uphold the high standards of quality for which Hackampus.com is known, verifying that all features work as intended and meet user and students requirements.

- Enable Efficient Development Cycles:

To align testing activities with Scrum sprints, facilitating swift identification and resolution of defects, and supporting the development team in quick iterations.

- Support Business Goals:

To ensure that the testing process aligns with the overarching business objectives, contributing to the sustained success and growth of hackampus.com.

1.3 Scope

The scope of this document is only for the “Hackampus.com” product.

This STP won't include the test planning and test execution of “Hackampus.com” on the following OS: Linux, MacOS.

This STP will include the test planning and test execution of “Hackampus.com” on the following Windows versions: 10, 11.

2 Scope of testing

2.1 Features to be tested

<Note that because its evolving document that some features/Modules could be added / deleted while the project is on process depends on timetables and complexity>

- Register to the website.
- Login to the website.
- Search Box / “Fast search”.
- Support and bug report feature. - (only interface test)
- Inbox
- Change language.
- My profile.
- Homepage tab.
- Books tab.-
- Meetings tab.
- VOD tab.
- Tests tab.
- Grades tab.
- Favorites tab.
- Classes tab.

2.2 Features not to be tested

- Webinars tab.
- Tasks tab.

- Feedbacks tab.
- Forum tab.

2.3 Testing Types

Outlined below are the test types that will be planned and performed during this project:

Smoke Level/Test:

We will ensure sign in and connection to the student's system, smart management of the studied content, automatic transfer to the ZOOM application during class/meeting.

Sanity level:

- Functionality Verification:

To ensure all features of “Hackampus.com”, such as query input, search execution, register form, login form and tabs.

We will ensure that the tests function as intended on different devices and browsers.

- Usability Assessment:

To evaluate the user interface for intuitiveness and ease of use.

This includes ensuring the search page, homepage and the rest of the tabs are easily navigable and that the interface elements are responsive to user interactions.

- Compatibility Testing:

To confirm that “Hackampus.com” works seamlessly across different browsers (e.g., Chrome, Firefox, Safari, Edge), operating systems (Windows, Android, iOS), and devices (desktops and smartphones).

- Localization and Internationalization Verification:

To ensure that "Hackampus.com" provides accurate results and a user-friendly experience in different languages and regions.

- Search Result Accuracy:

To validate the relevance and accuracy of search results provided by the search algorithms.

- Integration/Interface Testing:

- We will ensure once we join an online/recorded meeting that the ZOOM app will open and function properly.

- Once accessing the "Books" tab, specifically for books uploaded through the PowerPoint app, we will verify that when selecting a book, the integration with the PowerPoint app functions correctly.

- Once accessing the "Support and bug report" button, we will verify that the integration with "Jira service management" platform functions correctly.

- GUI Testing:

We will make sure that all the visual elements are correct, legible and working, for example buttons.

- Negative Testing:

We will perform tests that challenge the system and tests that do not challenge the system.

We will check that the system meets its limits and there are no exceptions.

- Error Handling Testing:

We will make sure when entering details for example in the registration form that as soon as we have not met the site's requirements a clear and accurate error message will be displayed.

- Security testing :

We will check when it comes to typing a password, details that need to be disguised that the system is indeed secure enough for its users.

We will use devtools to confirm.

