

AUTOMATION TESTING ACTITIME
Project based experience learning program

ACTITIME Automation Testing Project

ABOUT ACTITIME :

ActiTIME is a website deals with product for Time Tracking Software for cost-effective projects. actiTIME boosts performance, track work progress and don't leave any billable second untracked and records billable hours for a project resource for easy monitoring the work hours.

we can generate reports based on hours logged and also, we can integrate with other platforms like google calendar, Zapier etc.

actiTIME is not the only option for Time Tracking Software. Explore other competing options and alternatives. Time Tracking Software is a widely used technology, and many people are seeking productive, high quality software solutions with ease of completing timesheets, invoice creation and delivery, and time tracking. Other important factors to consider when researching alternatives to actiTIME include tasks and design. We have compiled a list of solutions that reviewers voted as the best overall alternatives and competitors to actiTIME, including Toggl Track, BigTime, QuickBooks Time, and Harvest.

The Actitime Manual Testing Project aims to thoroughly test the web-based time-tracking and project management software, Actitime. The project team will conduct comprehensive testing of Actitime's features and functionalities to identify any defects, inconsistencies, or usability issues that might impact user experience.

Table of Contents

- Introduction
- Project Overview
- Getting Started
 - ✓ Pre requisites
 - ✓ Installation
 - ✓ Project Setup
- Project Structure
- Writing Test Cases
- Running Tests
- Test Reporting
- Advantages & Disadvantages
- Application
- Troubleshooting
- Conclusion

Introduction

The actiTIME project is a Python automation project that utilizes the Selenium framework for automating web interactions. This documentation provides an overview of the project, instructions for installation and configuration, details about the project structure, and examples of test cases.

actiTIME is a web-based timesheet for companies of any size and any business type. It allows you to enter time spent on different work assignments, register time offs and sick leaves, and then create detailed reports covering almost any management or accounting needs.

Project Overview

The actiTIME project aims to automate various tasks on the Bluestone website. It leverages Selenium, a powerful web automation tool, to interact with web elements and perform actions such as clicking buttons, filling forms, and extracting data.

actiTIME manual testing is the process of verifying the functionality and quality of actiTIME software by executing test cases manually without using any automated tools. It involves checking the features such as time tracking, project management, billing, leave management, and integrations of actiTIME according to the user requirements and expectations. It also involves reporting and re-testing any bugs or issues that are found during the testing process. actiTIME manual testing helps to ensure that the software is bug-free and meets the customer needs.

Milestone 1

Getting Started

Explain the prerequisites and setup required to run the project.

Activity 1: Prerequisites

Before getting started with the actiTIME project, ensure you have the following prerequisites:

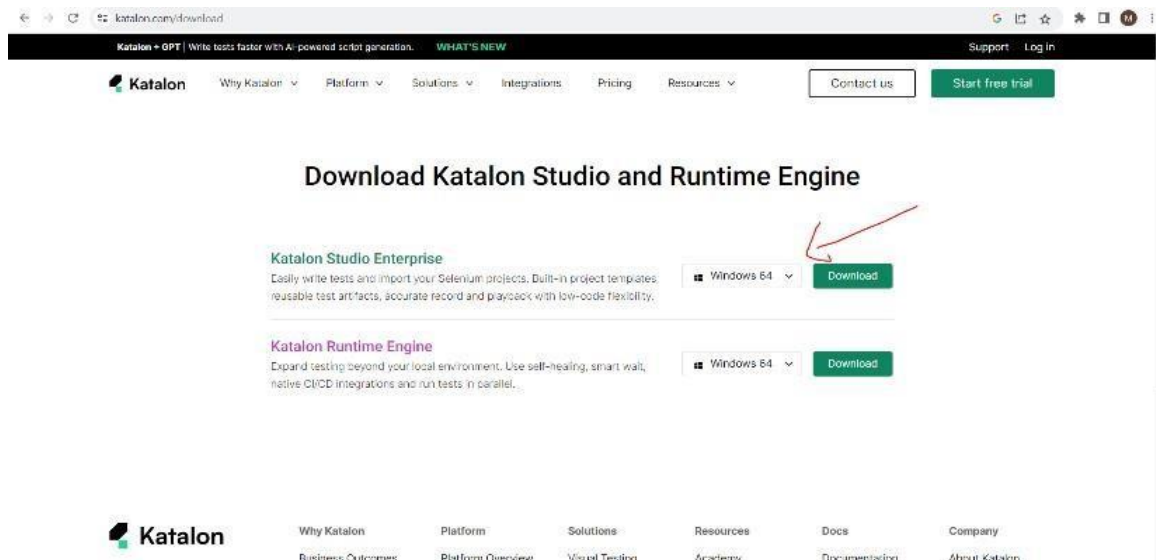
List the software and tools that need to be installed before running the project, such as:

- **Python (3.x)**
- **Selenium (3.x)**
- **Pytest (Latest Version)**
- **Any other dependencies**

Activity 2 : Installation

To install the necessary dependencies for the actiTIME automation project, follow these steps:

Activity 2.1:- Install Katalon Studio: Visit the Katalon Studio website (<https://katalon.com/download>) and download the latest version of Katalon Studio. Follow the installation instructions for your operating system.

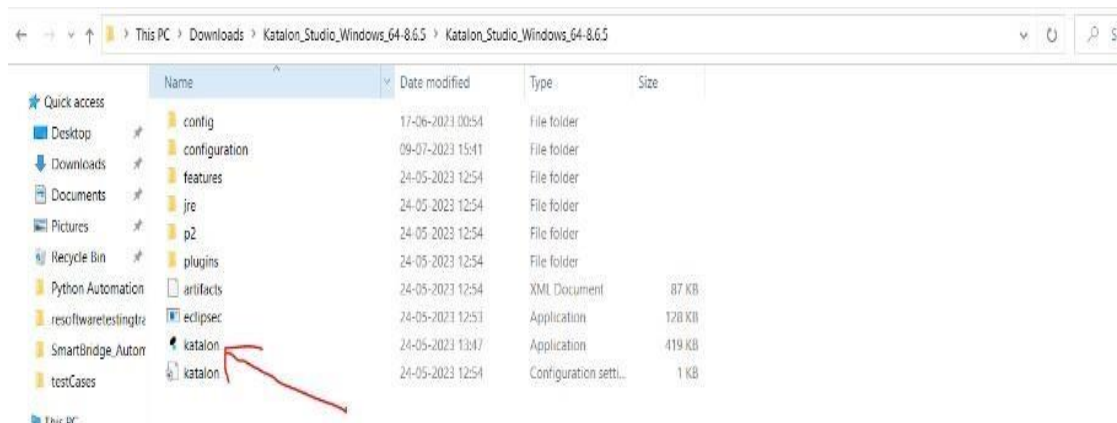


After downloading Katalon Studio and extracting the contents from the zipfile, locate the folder where you extract edit.

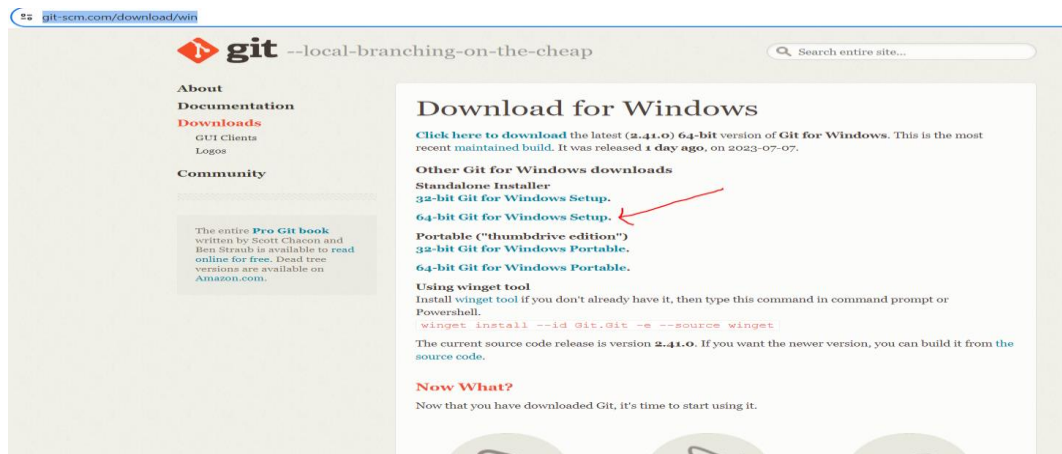
In that folder, look for the "katalon.exe" file. This file is the executable for Katalon Studio.

Double-click on the "katalon.exe" file to launch Katalon Studio. It may take a few moments to startup.

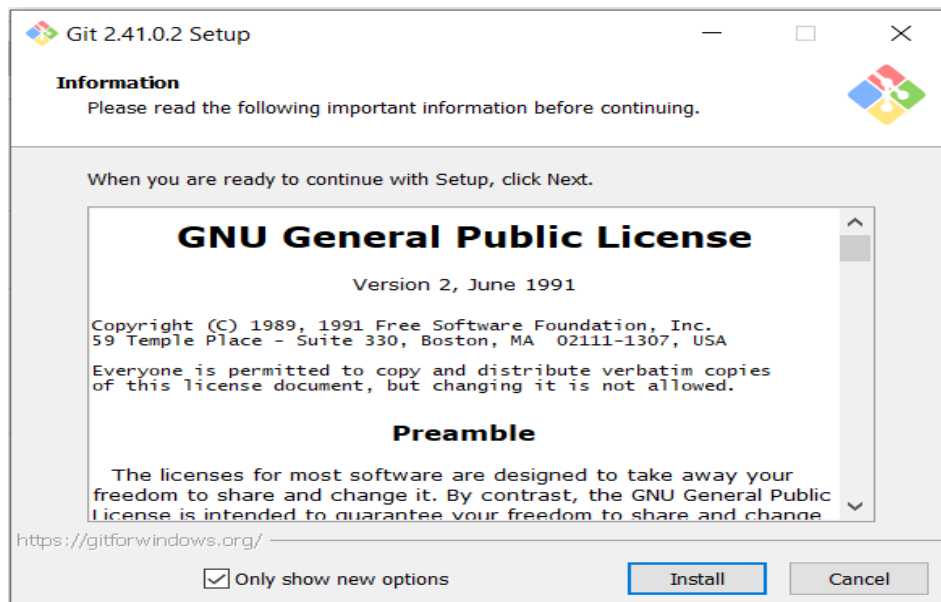
Once Katalon Studio is launched, you can start creating or opening your project by selecting "File" from the menu bar and choosing either "New Project" or "Open Project" based on your requirements.



Download and install **gitbash** from <https://git-scm.com/download/win>



Activity 2.3 :

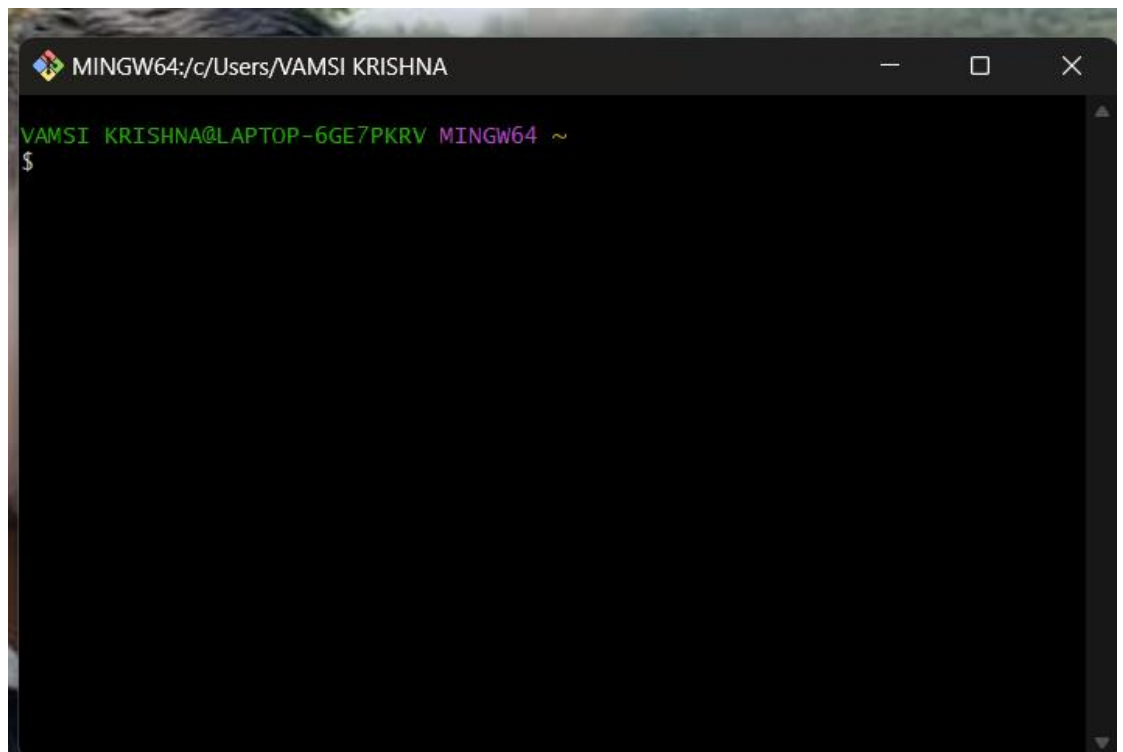


Activity 2.4 :

Go to the desired location where you want to create the folder, such as Desktop.

Folder name like eg...(Automation_Testing_Project)

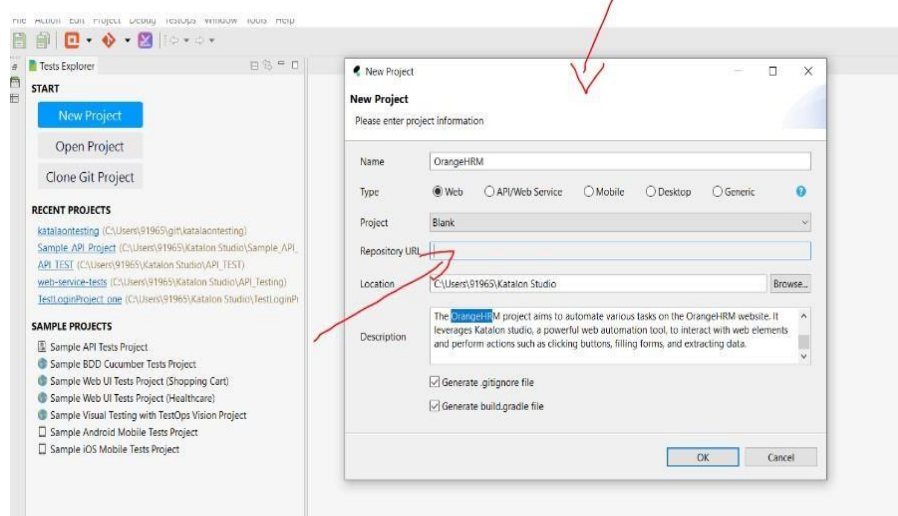
Open the folder-→ Right click with in the folder and select open gitbash here, you will get pop up like below and clone the repository based on the next steps provided be



Activity 3:- Project Setup

Explained how to set up the project locally, including steps like:

Before cloning, the project create project like below in Katalon Studio.



Activity 3.1:- Clone the repository from github.

Activity 3.2:- Clone the actiTIME project repository from [repository URL].

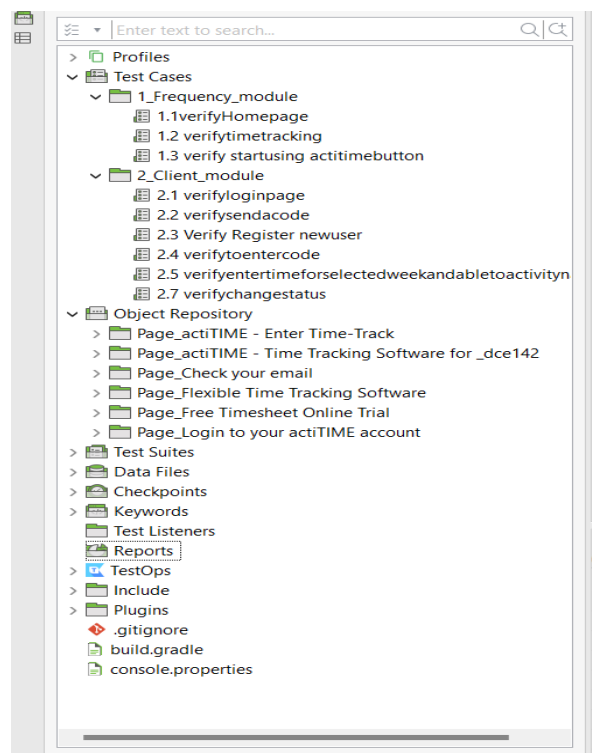
<https://github.com/venkyrevalla/Software-Testing-Automation/tree/main/projects%20file>

Activity 3.3 : Configuration settings (if any)

Milestone 2

Project Structure

Activity 1: Explained the structure of the project, including directories and files, and their purposes. For example:



Tests cases/: Directory containing test scripts

Object Repository/: Directory containing page object models

Test Suites/: Directory containing run multiple testcases reports/:

Directory containing test reports

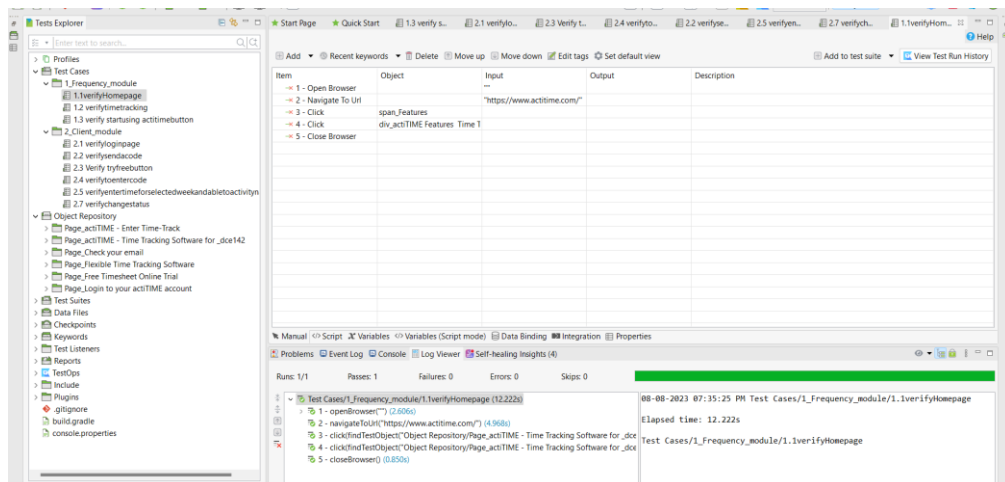
datafiles/: Directory containing test data (ex: excel, csv ,text ,json files) related to project Run:-Option to run all the test cases from the top quick aces tool bar.

Activity 2 : Writing Test Cases

Explained how to write test cases using Python, Selenium, and pytest.
Provide examples of different types of test cases, such as:

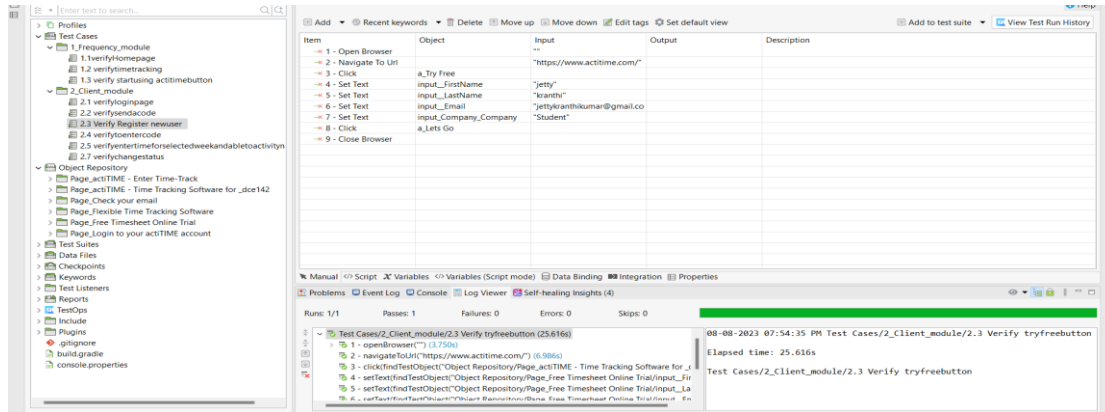
Test case 1: Home Page functionality

Screenshot of the code:



Test case 2: Register new user functionality

Screenshot of the code:



Milestone 3

Running Tests

Explained how to run the test cases using Katalon studio. Include information about command-line and flags that can be used to customize test execution.

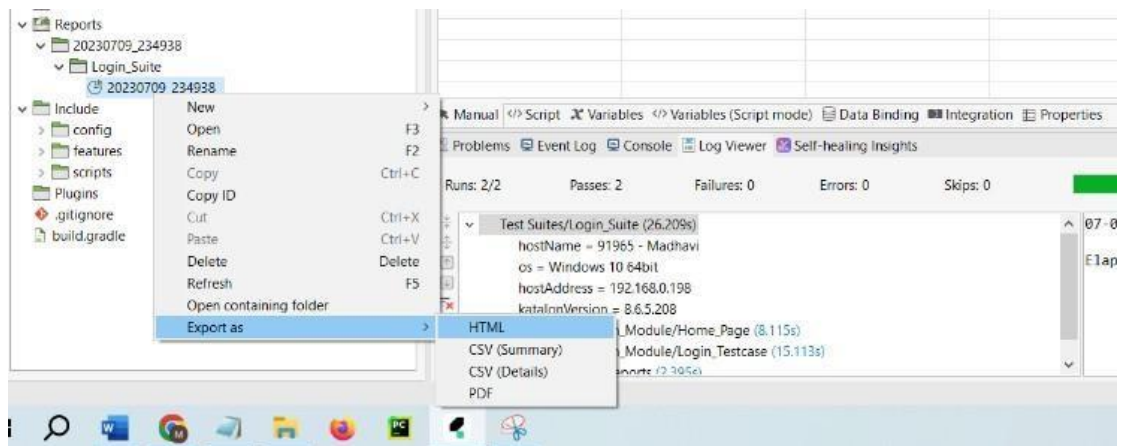
Activity1: Option to run all the test cases from the top quick access toolbar.

Milestone 4

Test Reporting

Explained how test reports are generated and where they are stored. Include information about any tools or libraries used for reporting, such as katalon reporting. -html.

Activity1: reports/: Directory containing test reports



Login Suite Test Log	
Execution Environment Host name: 91965 - Madhavi Local OS: Windows 10 64bit Katalon version: 8.6.5.208 Browser: Chrome 114.0.0.0	
Test Execution Log	
<div> <div>TEST SUITE: Login Suite</div> <div> Full Name: Login Suite Start / End / Elapsed: 2023-07-10 07:50:11.500 / 2023-07-10 08:00:27.103 / 00:01:15.603 Status: 2 Test total, 2 passed, 0 failed, 0 error, 0 warning, 0 skip, 0 suspend </div> </div>	Expand All
<div> <div>TEST CASE: Test Cases/Login_Module/Home_Page_Title</div> <div> Full Name: Login Suite/Test Cases/Login_Module/Home_Page_Title Description: Home page title validator Tag: Smoke Start / End / Elapsed: 2023-07-10 07:50:11.900 / 2023-07-10 07:50:26.104 / 00:00:14.204 Status: PASSED </div> </div>	Expand All
<div> <div>TEST STEP: openBrowser()</div> <div> Full Name: navigateToUrl("https://katalon.com") Test Step: initializeWebDriver Test Step: closeWebDriver Test Step: closeWebDriver </div> </div>	
<div> <div>TEST CASE: Test Cases/Login_Module/Login_Test_Case</div> <div> Full Name: Login Suite/Test Cases/Login_Module/Login_Test_Case Description: Login Test Case using valid credentials Tag: Smoke Start / End / Elapsed: 2023-07-10 07:50:26.121 / 2023-07-10 08:00:27.103 / 00:01:00.982 </div> </div>	Expand All

Advantages :

- Actitime is a web-based project management and time tracking software that helps businesses manage tasks, projects, and track employee work hours. When it comes to manual testing, Actitime offers some advantages that can be beneficial for users.
- User-Friendly Interface: Actitime's manual testing module typically has a user-friendly interface, making it easy for testers to navigate and perform their testing tasks efficiently.
- Easy Test Case Creation: Manual testing in Actitime allows testers to create and manage test cases easily. Testers can define the steps, expected results, and associated test data for each test case.
- Flexible Test Execution: Actitime provides flexibility in test execution. Testers can execute test cases in any order, allowing them to adapt to changing priorities and requirements during the testing process.

- **Test Progress Tracking:** Manual testing in Actitime offers features for tracking the progress of testing activities. Testers can update the status of each test case, indicating whether it's in progress, passed, failed, or blocked.

Disadvantages:

- While Actitime's manual testing module offers some advantages, it also has certain disadvantages that users should be aware of
- **Time-Consuming:** Manual testing can be time-consuming, especially for large and complex projects. Testers need to execute each test case step by step, which can be inefficient and slow down the overall testing process.
- **Prone to Human Errors:** Manual testing heavily relies on human involvement, making it susceptible to human errors and oversight. Testers might miss critical issues, leading to potential defects in the software.
- **Limited Test Coverage:** Manual testing may have limited test coverage compared to automated testing. Testers may not be able to execute all possible test scenarios due to time constraints or the repetitive nature of executing the same test cases repeatedly.
- **Resource-Intensive:** Manual testing requires skilled testers who can dedicate significant time and effort to execute test cases. This can lead to higher resource costs, especially for projects that demand extensive testing efforts.

Application

- ActiTIME is a web-based time tracking and management software, primarily used for project and task management, timesheet management, and workforce analysis. Manual testing plays a crucial role in ensuring the quality and functionality of the ActiTIME application. Here are some of the key applications of ActiTIME manual testing.

- **Functional Testing:** Ensure that all the functionalities of ActiTIME work as expected and meet the specified requirements. This includes testing features like time tracking, task creation, project management, reports generation, user management, etc.
- **User Interface (UI) Testing:** Verify the consistency and usability of the user interface across different browsers and devices. Focus on validating the layout, navigation, and visual elements.
- **Compatibility Testing:** Test ActiTIME on various web browsers, operating systems, and devices to ensure its compatibility and proper functionality across different environments.

Troubleshooting

List common issues that may occur during project setup or test execution and provide possible solutions.

- **Troubleshooting ActiTIME manual testing** involves identifying and resolving issues that arise during the testing process. Here are some common troubleshooting steps you can follow:
- **Identify the Problem:** First, carefully document and reproduce the issue you encountered during manual testing. Identify the exact steps that led to the problem, including any specific inputs or configurations used.
- **Check Test Environment:** Ensure that your test environment is set up correctly, including the browser version, operating system, and any required plugins. Verify that ActiTIME is running on the correct server and that all dependencies are properly configured.
- **Verify Test Data:** Double-check the test data you used during testing. Ensure that it is valid and properly set up to simulate real-world scenarios accurately.

Conclusion

The **actiTIME** automation project provides a framework for automating various tasks within the **actiTIME** web application using Python, Selenium, and Pytest. With this documentation, you should be able to set up the project, execute tests, and extend/customize it according to your needs.