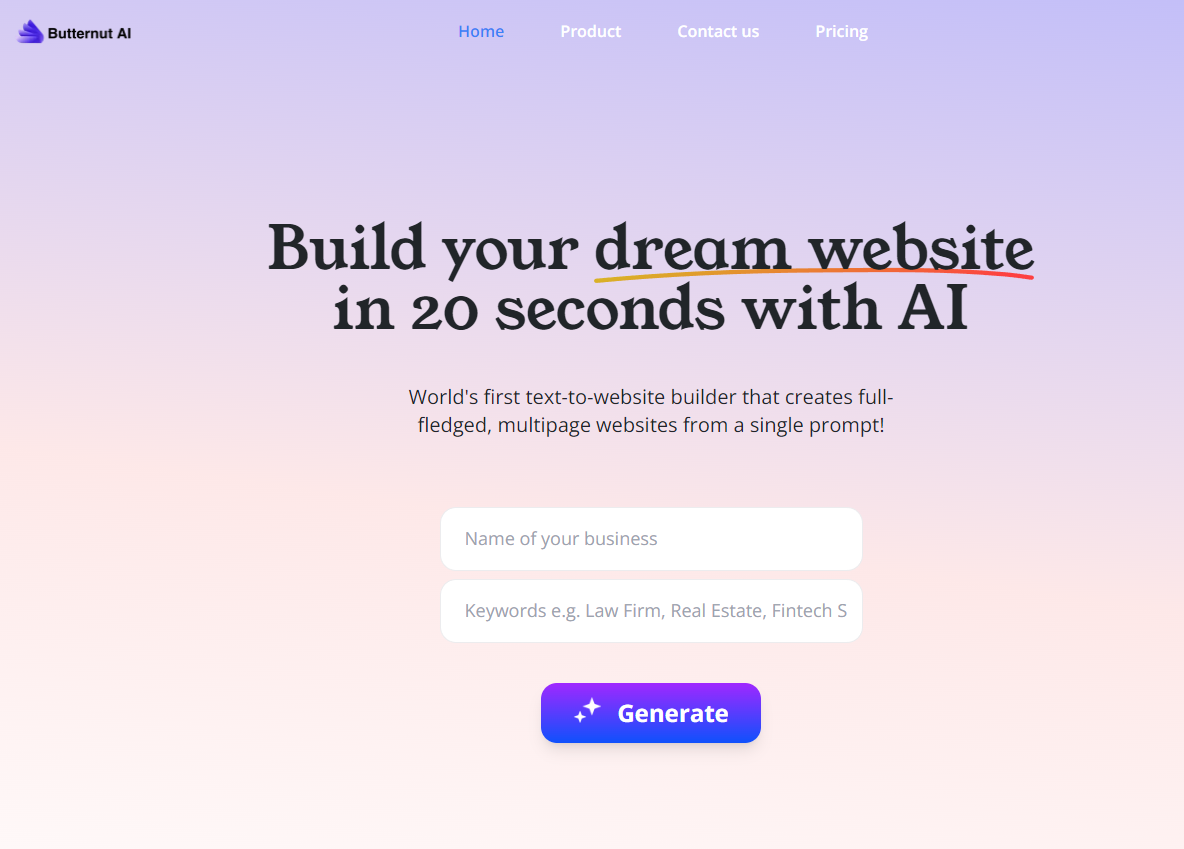
**Test Progress Report**

**Butternut AI [Web Application]**



| **Version** | **Date** | **Created By** |
| --- | --- | --- |
| V2 | 24/03/2024 | Aryan Vashishth |
|

**Report Outline:**

[**Introduction**](#_23ckvvd)

[**Test Case Execution Status**](#_ihv636)

[**Bug Report Status for Main and Generated Website**](#_32hioqz)

[**Testing suggestions for next period**](#_1hmsyys)

# **Introduction**

This Test summary report provides insights into the testing activities conducted for Butternut AI's homepage and its generated website features. The purpose is to provide an overview of:

* Test case execution status
* Defects found & their status
* Suggestions for improvement in the next testing phase

# **Test Case Execution Status**

This is the status of all the test cases that are run on the **Web Application** version:

| **Test Case Status** | **Current Build Results** |  |
| --- | --- | --- |
| Total Number of TCs | 94 |  |
| Passed TCs | 72 |  |
| Failed TCs | 18 |  |
| Blocked/Skipped TCs | 4 |  |

This is the status of all the test cases that are run on the **Android** version [There is no Android build]:

| **Test Case Status** | **Number of Test Cases** |
| --- | --- |
| Total Number of TCs | NA |
| Passed TCs | NA |
| Failed TCs | NA |
| Blocked/Skipped TCs | NA |

* Test Case Execution notes:
  + Test cases involving integrations were executed on Android devices due to the absence of an **Android** build.
  + Only 2 **critical bugs** were identified in this period.
  + 4 test cases were **blocked** due to dependency on previous cases.

# **Bug Report Status for Main and Generated Website**

This table lists the defects For Text-to-Website (**Main Website**):

| **Bug ID** | **Description** | **Status** |
| --- | --- | --- |
| TC\_TTW\_Home\_09 | Verify error message for random invalid text input | Failed |
| TC\_TTW\_Home\_10 | Verify absence of generated website for random invalid text input | Blocked |
| TC\_TTW\_Home\_11 | Verify error message for special characters in text input | Failed |
| TC\_TTW\_Home\_12 | Verify absence of generated website for special characters in text input | Blocked |
| TC\_TTW\_Home\_13 | Verify error message for numeric input in Name of the Business field | Failed |
| TC\_TTW\_Home\_14 | Verify absence of generated website for numeric input in Name of the Business field | Blocked |
| TC\_TTW\_Home\_15 | Verify error message for numeric input in keywords field | Failed |
| TC\_TTW\_Home\_16 | Verify absence of generated website for numeric input in keywords field | Blocked |

This table lists the defects For Comment Generation (**Generated Website**):

| **Bug ID** | **Description** | **Status** |
| --- | --- | --- |
| TC\_CG\_02 | Verify Maximum Length Limit | FAIL |
| TC\_CG\_03 | Verify Empty Input Handling | FAIL |
| TC\_CG\_04 | Verify Invalid Characters Handling | FAIL |
| TC\_CG\_09 | Verify Comment Generation Error Handling | FAIL |
| TC\_CG\_19 | Verify Handling of Special Characters | FAIL |
| TC\_CG\_20 | Verify Handling of Numbers | FAIL |
| TC\_CG\_21 | Verify Handling of Single Alphabets | FAIL |
| TC\_CG\_22 | Verify Handling of Invalid Words | FAIL |

This table lists the defects For Image Generation (**Generated Website**):

| **Bug ID** | **Description** | **Status** |
| --- | --- | --- |
| TC\_IG\_02 | Verify Customization Options | FAIL |
| TC\_IG\_09 | Verify Error Handling | FAIL |
| TC\_IG\_11 | Verify Handling of Special Characters in Input | FAIL |
| TC\_IG\_12 | Verify Handling of Numbers in Input | FAIL |
| TC\_IG\_13 | Verify Handling of Single Alphabets in Input | FAIL |
| TC\_IG\_14 | Verify Handling of Invalid Words in Input | FAIL |

# **Testing suggestions for next period**

1. **Address Text-to-Website Issues:** Focus on fixing issues related to the Text-to-Website feature. Improve how the system handles incorrect inputs and displays error messages. Ensure it works smoothly even with unusual inputs.
2. **Resolve Blocked Test Cases:** Investigate and resolve the blocked test cases for Text-to-Website. Make sure the feature works as expected in all scenarios, including when inputs are invalid.
3. **Improve Comment Generation:** Enhance the Comment Generation feature by fixing the problems identified in failed test cases. Make sure the system handles empty inputs, special characters, and numbers properly.
4. **Optimize Image Generation:** Review and optimize the Image Generation feature. Address issues related to customization options and handling of different types of inputs. Ensure the feature produces the desired images without errors.
5. **Thorough Regression Testing:** Conduct comprehensive regression testing to verify that fixes for identified defects work correctly and do not introduce new issues. Make sure all previously working features remain functional.
6. **Explore Edge Cases:** Look into additional scenarios and edge cases for testing. This helps uncover any hidden defects or usability issues that may not have been detected earlier.
7. **Collaborate with Development:** Work closely with the development team to prioritize and fix critical bugs found during testing. Coordinate efforts to ensure a smooth resolution of issues.
8. **Implement Error Logging:** Set up robust error logging and monitoring mechanisms. This helps in tracking and diagnosing errors encountered during image and comment generation, facilitating quicker resolution.
9. **Consider Automation:** Explore the possibility of using automated testing solutions to streamline the testing process and increase test coverage. Automated tests can help catch regressions early and save time.
10. **Gather User Feedback:** Continuously collect feedback from users and analyze application usage data. This provides valuable insights into areas for improvement and helps prioritize future enhancements effectively.