Testing Analyser Document

Team 33 - InHobi

Unit Testing

Unit Testing is the process where each individual unit or component of a software is tested. This is almost always done by the developers themselves, but could be done by independent testers as well.

The main usage of this testing is to validate that each unit of the software performs as required.

This testing was performed for each of the frontend & backend components separately. For the backend, each API method was tested for functionality via Postman requests. For the frontend, each dynamic page was tested for operability via user-interface components.

Integration Testing

Integration Testing is the process in which individual units are combined and tested as a group. This occurs right after unit testing has been done, and all the units are thoroughly examined. This is usually done by developers themselves or independent testers.

The main usage of this testing is to reveal any faults in the interaction between integrated units.

This testing was performed on a webpage basis for the application, where each webpage was tested independently once the backend and frontend were tested & connected. This was mainly done in coherence to the **Test Planner** document where each functionality was tested through user-interface components.

System Testing

System Testing is the process wherein the complete and integrated software is tested. This is usually done by the testing team, to ensure functionality of the product.

The main usage of this testing is to evaluate the system's compliance with the specified requirements.

After all of the unit & integration testings were completed, the complete system testing was done by the development team. This was done by adhering to the testing guidelines we've declared in

the **Test Planner** document. Majority of this level of testing was done by directly interacting with the product through the user-interface, and noticing the system's reactions.

Acceptance Testing

Acceptance Testing is the process of testing the user's needs against the application's performance & functionality. This is usually done by the user themselves.

The main usage of this testing is to evaluate the system's compliance with the client's requirements and assess whether it is acceptable for delivery.

The client performed this testing using **black box testing** for the R1 & R2 MVPs of our product created separately. He navigated and manually went through all the implemented features, and gave relevant feedback. Apart from this, each user-interface screen that was created for the frontend was separately sent to the client for feedback & validation. Only upon client's satisfaction was it introduced into the application and integrated with the backend.