***Black Box Testing:***

the software program or system under test is viewed as a “black box”. Black

Box Testing: it is testing based on the requirements specifications and there is no need to examining

the code in black box testing. This is purely done based on customers view point only tester knows the

set of inputs and predictable outputs. Black box testing is done on the completely finished product

the software program or system under test is viewed as a “black box”. Black

Box Testing: it is testing based on the requirements specifications and there is no need to examining

the code in black box testing. This is purely done based on customers view point only tester knows the

set of inputs and predictable outputs. Black box testing is done on the completely finished product

* The software program or system under test, is viewed as a “black box”.
* It is the kind of testing based on the requirements and specifications.
* This can be applied to every level of software testing such as Unit, Integration, System and Acceptance Testing.
* There is no need in examining the code in black box testing.
* It is purely done based on customers view point only.
* The tester knows the set of inputs and predictable outputs.
* It is done based on end user perspective.
* The programmers and testers are independent of each other.
* Black box testing is done on the completely finished product.

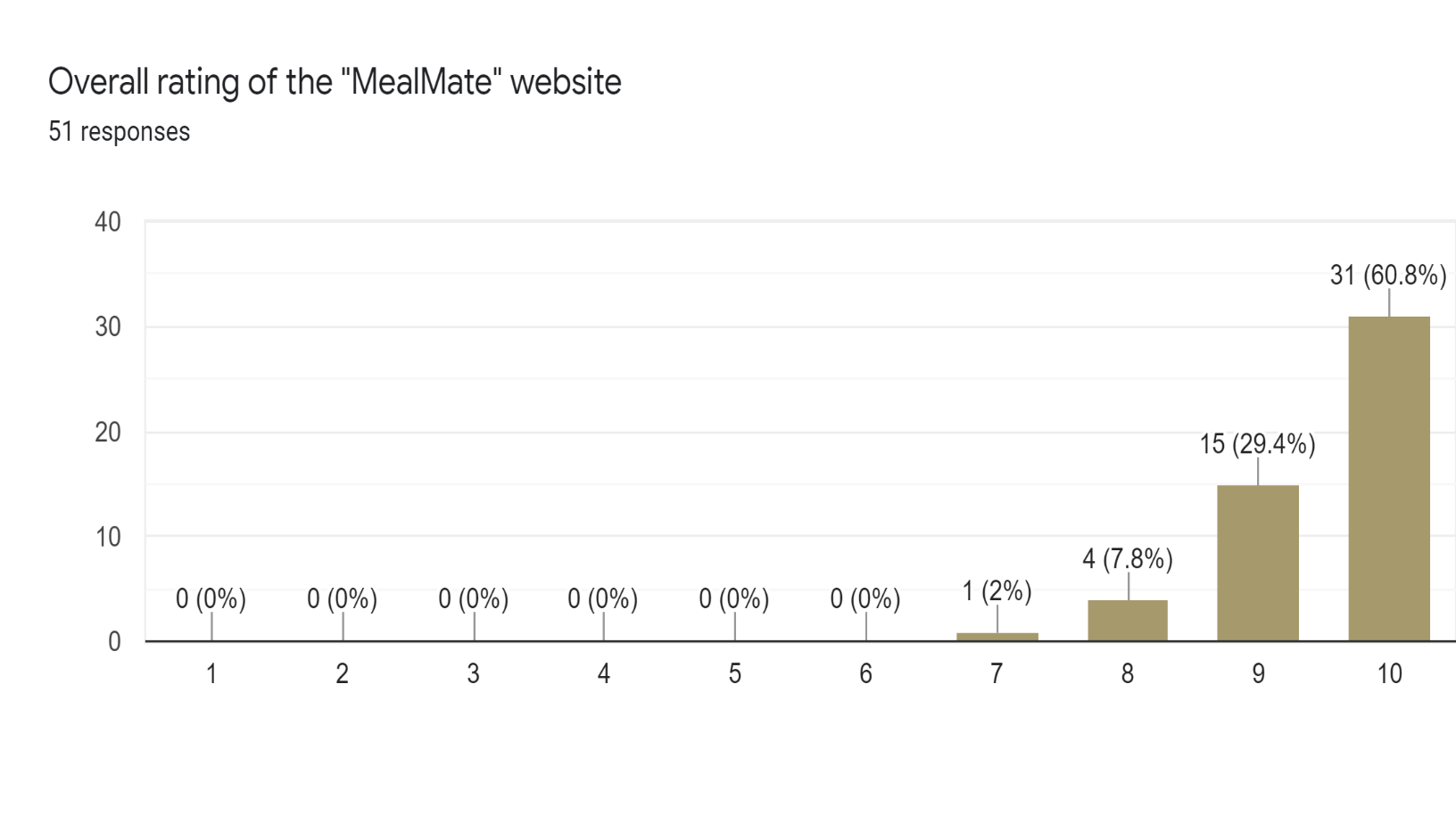
***Various parameters checked in Black Box Testing are:***

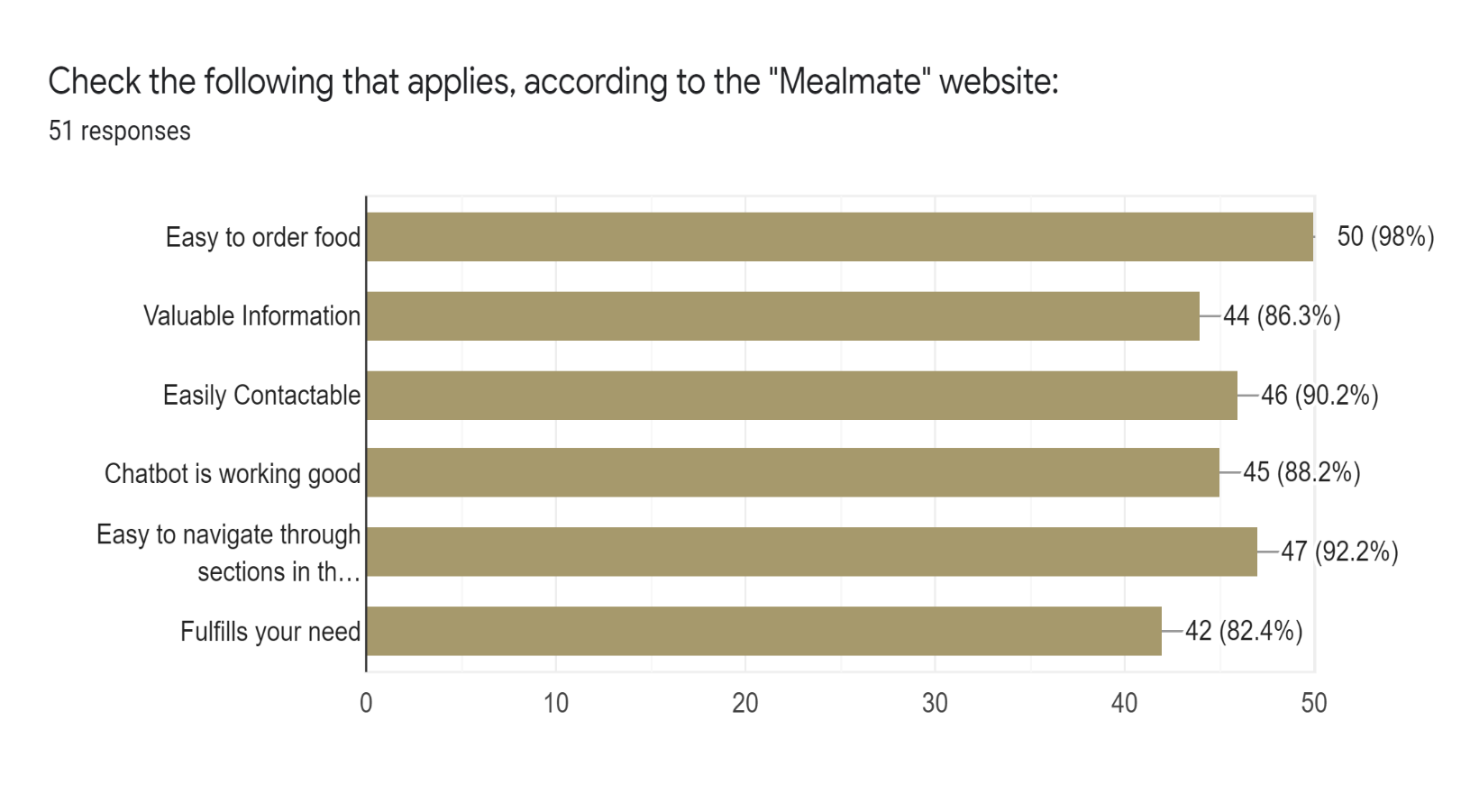
* Accurate actions performed by users
* System’s interaction with the inputs
* The response time of the system
* Use of data structures Issues in the user interface
* Usability issues
* Performance issues
* Abrupt application failure, unable to start or finish

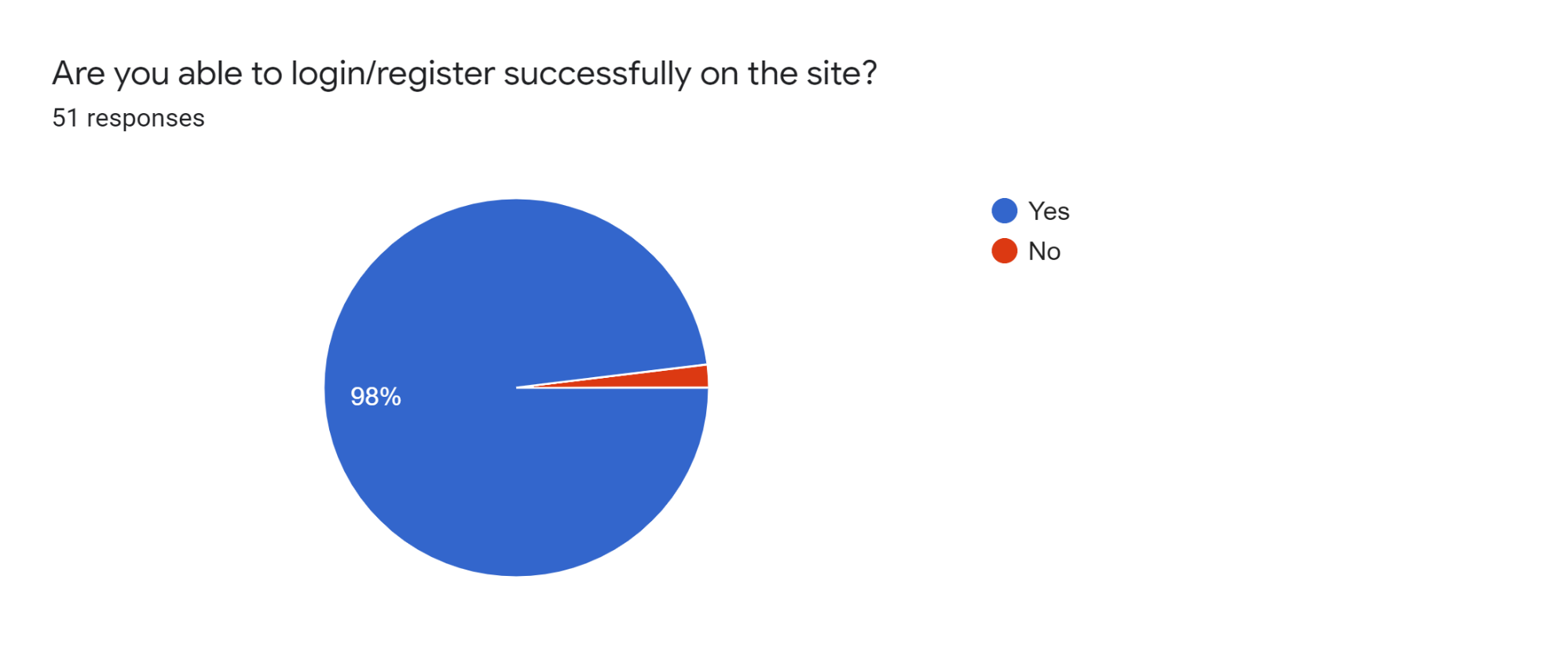
***When do we do Black Box Testing?***

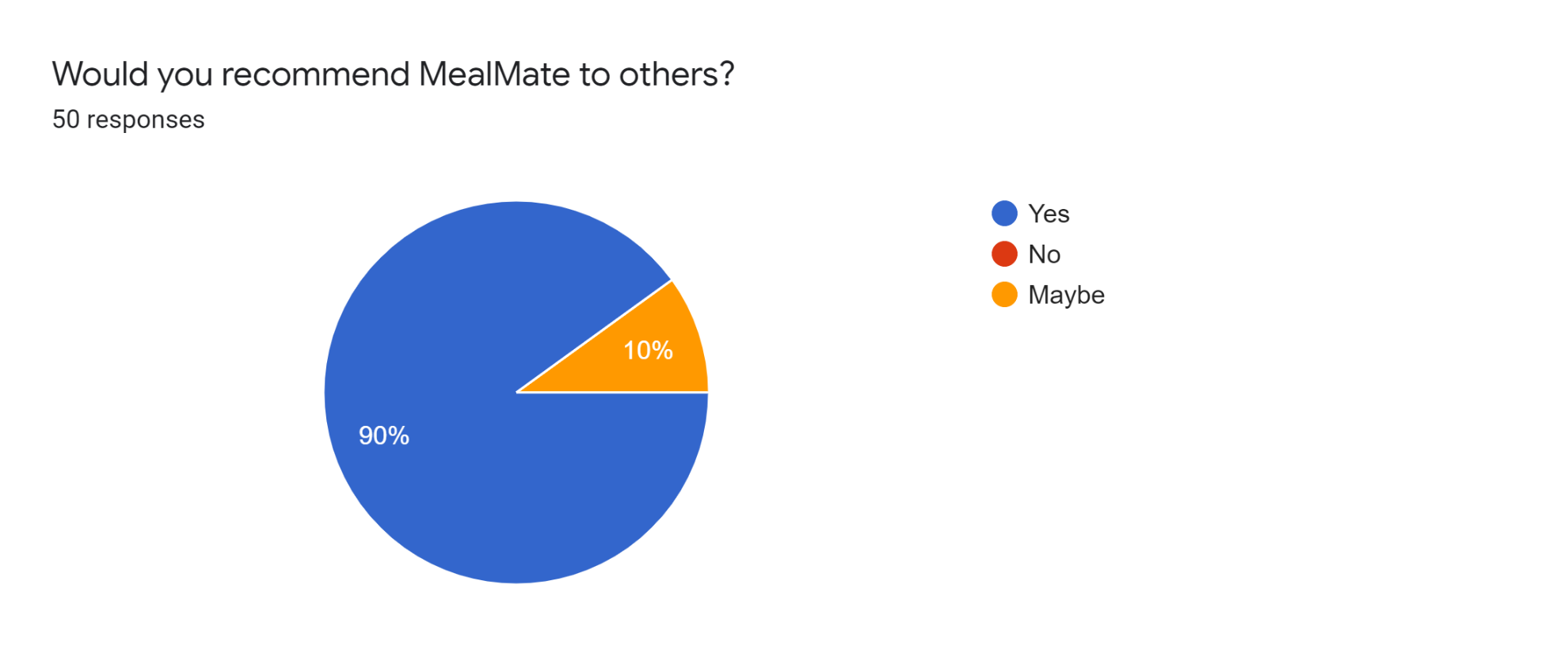
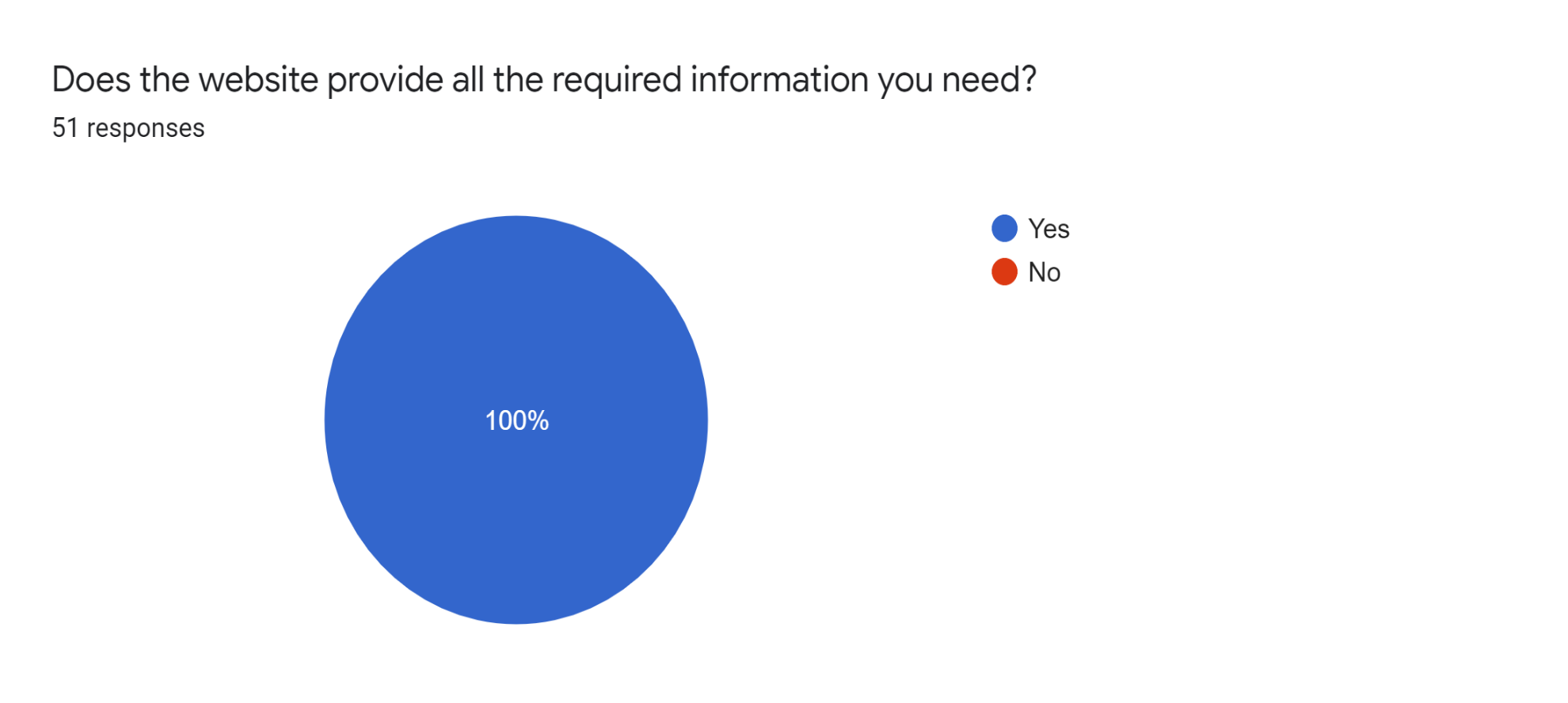
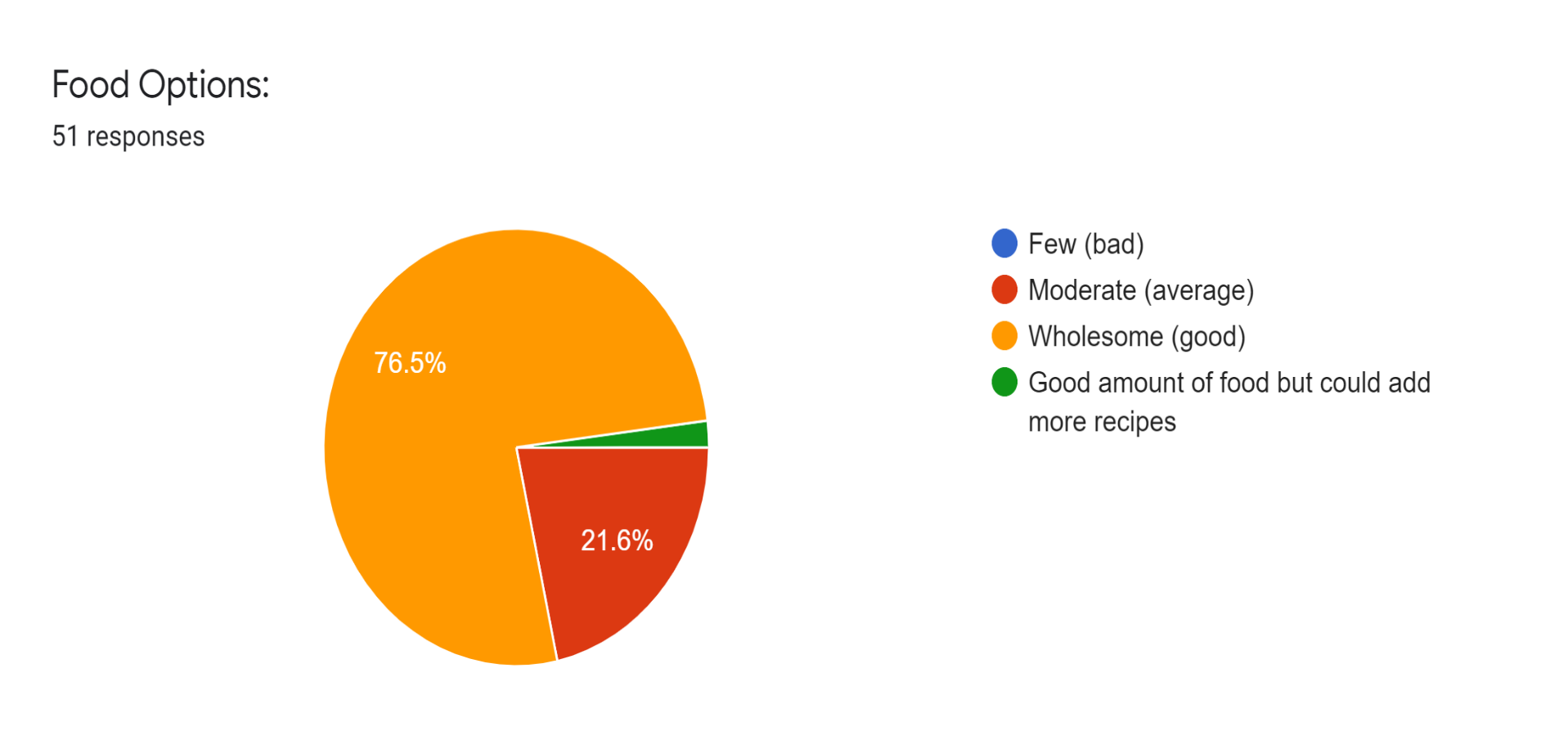
* Unlike traditional white box testing, black box testing is beneficial for testing software usability.
* The overall functionality of the system under test
* Black box testing gives you a broader picture of the software.
* This testing approach sees an application from a user’s perspective.
* To test the software as a whole system rather than different modules.

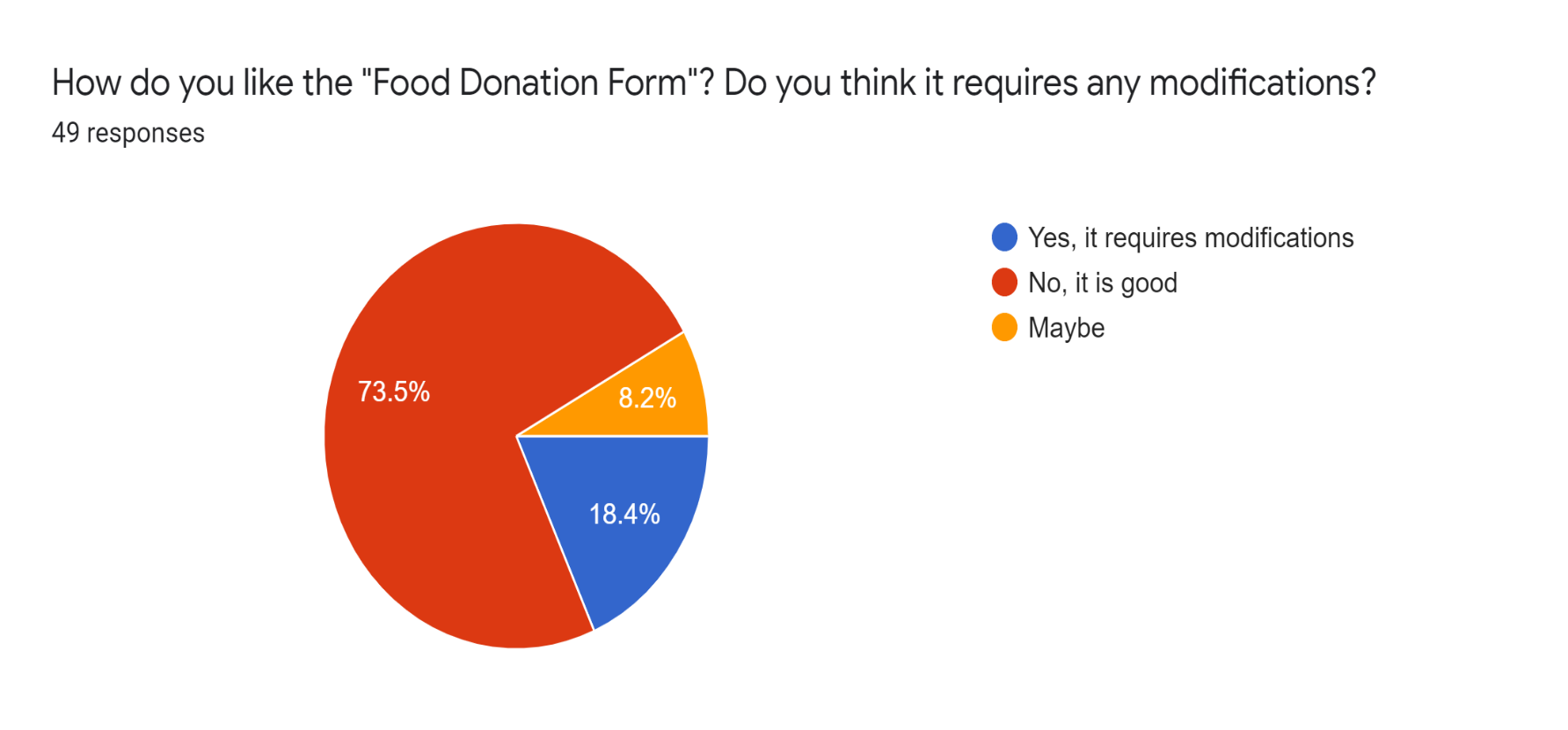
*We have conducted a survey on to check the above parameters and here are the results:*

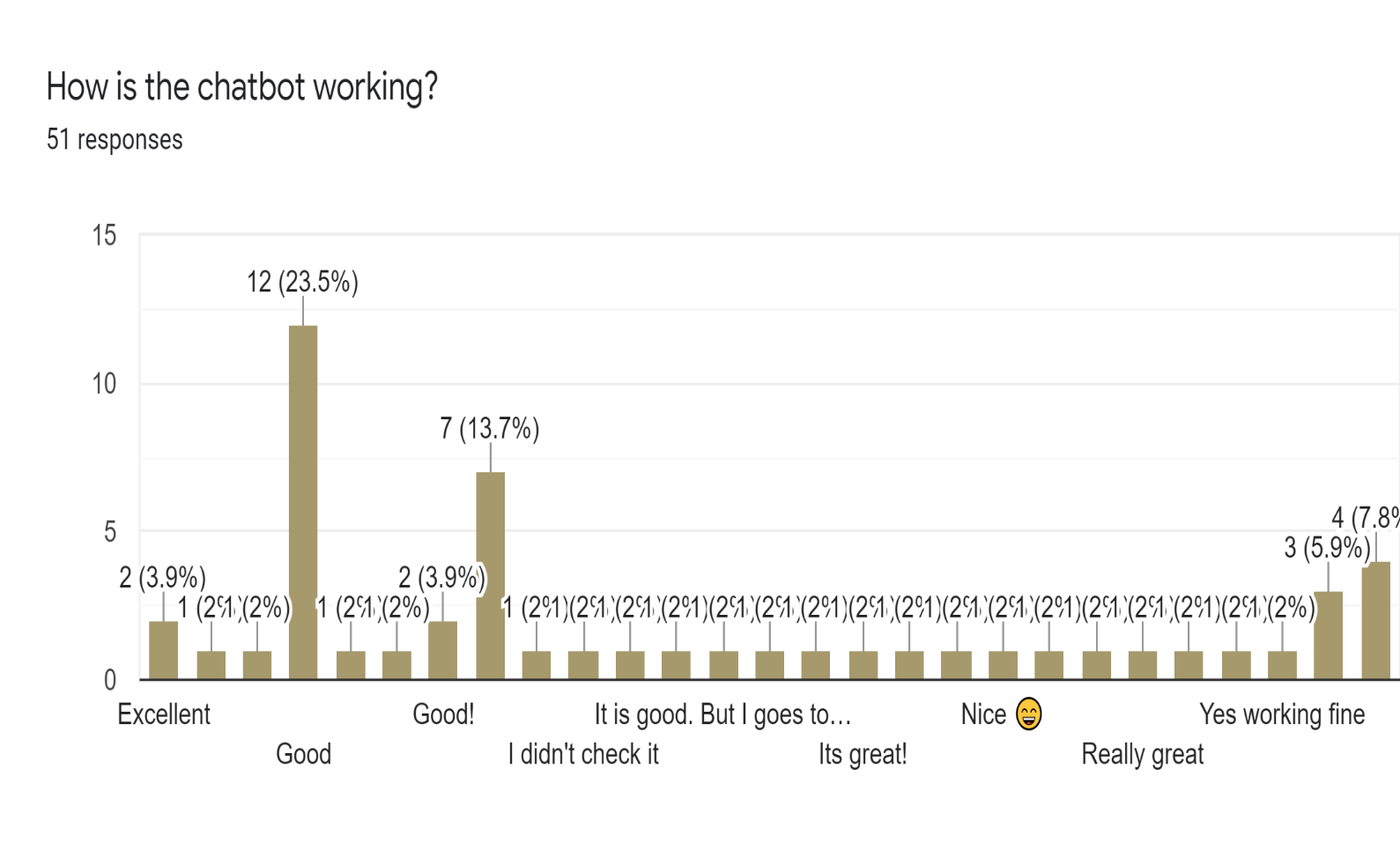










***White Box Testing:***

* **White box testing** refers to a scenario where (as opposed to [**black box testing**](https://en.wikipedia.org/wiki/Black-box_testing)), the tester deeply understands the inner workings of the system or system component being tested.
* Gaining a deep understanding of the system or component is possible when the tester understands these at program- or code-level.
* Tester will be able to design and execute test cases that cover all possible scenarios and conditions that the system component is designed to handle.
* By performing testing at the most granular level of the system, we are able to build a robust system that works exactly as expected, and ensure it will not go wrong in any way.

***Key Principles:***

* Statement Coverage – ensure every single line of code is tested.
* Branch Coverage – ensure every branch (e.g. true or false) is tested.
* Path Coverage – ensure all possible paths are tested.

***When do we do White Box Testing?***

* White box Testing is usually reserved for mission critical systems *(Example: Core Banking System for safe & secure transactions or account maintenance)* and components, because, well, such systems simply deserve the attention to detail that this technique can bring.
* The rigour that white box Testing employs is quite useful – yes, but not all the time.

***Steps to follow for White Box Testing:***

### Step 1: Identify the feature, component, program to be tested

### Step 2: Plot all possible paths in a flowgraph using:

* User journeys, use cases
* Program specifications
* Technical specifications, pseudocode

### Step 3: Identify all possible paths from the flowgraph

### Step 4: Write Test Cases to cover every single path on the flowgraph

### Step 5: Execute, rinse, repeat

***Flowchart:***

