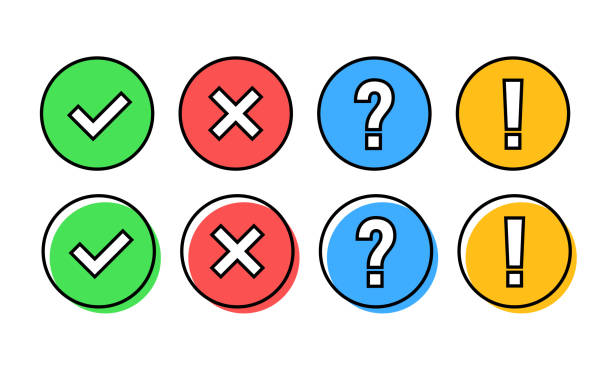
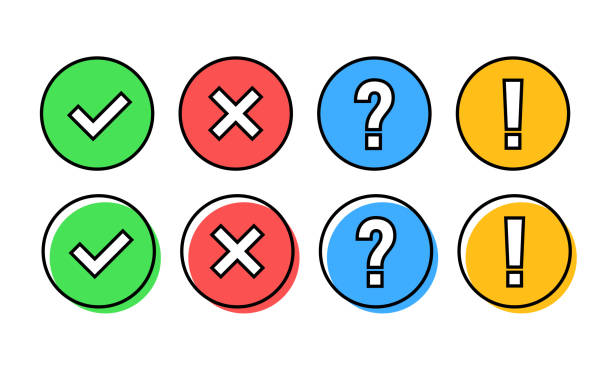
## **4c Restatement of User Req. as Design + Testing Objectives**

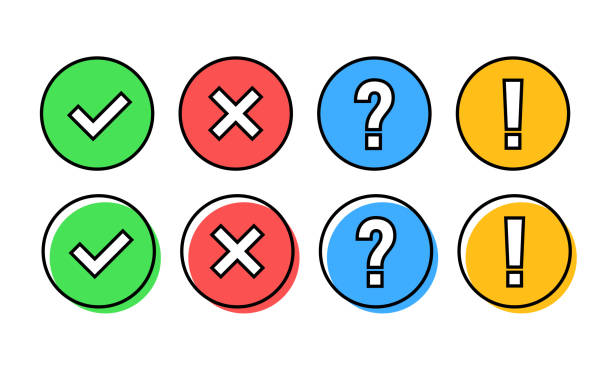
|  |  |  |
| --- | --- | --- |
| No. | Criteria | How it will be tested |
|  | ***Google API side*** |  |
| 1 | Google calendar API implementation is able to retrieve all of the correct and up-to-date data | View web interface and script output side-by-side |
| 2 | Data is updated periodically per unit time as decided eg. per hour | Demonstrate mechanism within code and if possible show periodic database entries |
| 3 | Database is used to store/ manage/ order the event data to ease further processing | Explained through written code and screenshots how the db is implemented |
| 4 | Information is retrieved at the accurate output time and text-to-speech functionality works | Video of pop-up output at specific expected time to show evidence |
|  | ***Gesture Recognition side*** |  |
| 5 | cv2 and mediapipe library function together to process each frame for landmark positions | Screenshots to demonstrate MP output on live camera feed |
| 6 | Gesture matrices are accurately built from test data and stored in accessible location | Screenshot filing and database system used |
| 7 | Error calculation method can reliably determine, and output, gesture received | Screenshots of input images, then expected and actual response comparison |
|  | ***Overall System*** |  |
| 8 | GUI is sufficient to demonstrate the functionality (not a direct objective) | Screenshot of working desktop UI running full main program |
| 9 | Text-to-speech and audio output resource is shared effectively between the two sides | Main program explained via code and then possibly further video evidence |



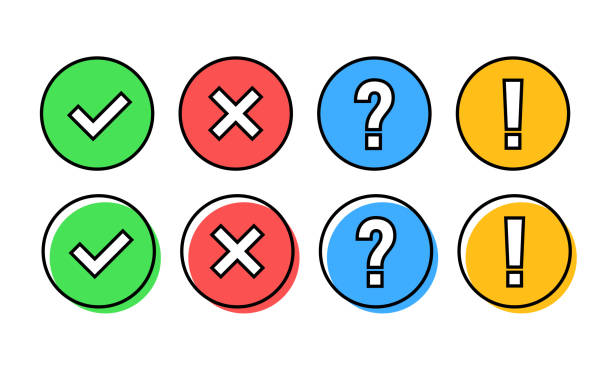
Fulfilled successfully



Fulfilled well however could be improved



Not fulfilled well and requires attention in future development



Not fulfilled at all