



Online Feedback System

C++ Mini Project

Collecting and analysing user feedback efficiently.

Group Members:

Raj Rasal

Kartik Wagh

Ajit Singh

Pallavi Sarovar

The Need for Feedback



Understand User Experience

Identify strengths and weaknesses from a user's perspective.

Improve Product/Service

Iterate and enhance offerings based on direct input.

Boost Engagement

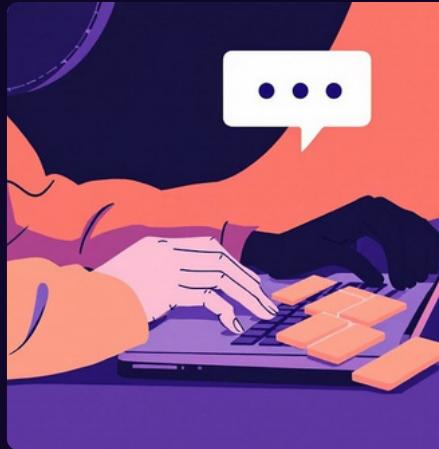
Show users their opinions are valued and heard.

Key Features in Focus



RatingInput (1-5)

Users provide a numerical score to quantify their experience.



Comment Submission

Detailed qualitative feedback to explain ratings.



Average Rating Display

Real-time calculation of overall user satisfaction.



Feedback Summary

Consolidated view of all submitted ratings and comments.

Code Walkthrough: Input Loop

The core logic handles user input and validates entries.

```
do{  
    // Get rating  
    cout<<"Enter your rating (1-5): ";  
    cin >> rating;  
    while(rating< 1 || rating > 5) {  
        cout<<"Invalid. Enter 1 to 5: ";  
        cin >> rating;  
    }  
    getline(cin,empty); // Clear buffer  
  
    // Get comment  
    cout<<"Enter feedback : ";  
    getline(cin, comment);  
  
    // Update totals  
    totalRatings+= rating;  
    numberOfRatings++;  
    feedbackSummary += to_string(numberOfRatings) + ". Rating: " + to_string((rating)) + " | Comment: " + comment + "\n";  
    //Askformore feedback  
    cout<<"More feedback? (y/n): ";  
    cin >> moreFeedback;  
    //...validation for moreFeedback ...  
  
}while(moreFeedback == 'y' || moreFeedback == 'Y');
```

1

Input & Validation

Ensures ratings are within the 1-5 range.

2

Data Accumulation

Aggregates ratings and stores comments.

3

Loop Control

Allows multiple feedback entries sequentially.

Code Walkthrough: Output & Summary

Presenting the collected insights.

Conditional Output

Only displays summary if feedback exists.

Average Calculation

Arithmetic mean of all valid ratings.

Formatted Summary

Presents individual ratings and comments clearly.

```
if (numberOfRatings > 0) {  
    double averageRating = totalRatings / numberOfRatings;  
    cout << "\n Summary\n";  
    cout << feedbackSummary;  
    cout << "Total: " << numberOfRatings << " ratings\n";  
    cout << "Average: " << averageRating << "\n";  
}  
return 0;
```

FUTURE ENHANCEMENTS

Expanding System Capabilities



Data Persistence

Store feedback in files or a database.



User Authentication

Allow registered users to submit feedback.



Advanced Analytics

Categorise comments; sentiment analysis.



Web Interface

Develop a graphical user interface (GUI).



Thank You!

This project demonstrates foundational C++ skills for practical applications.