

Functional Requirements Document for Tip Calculator

Contents:

1. Introduction

- 1.1. Purpose
- 1.2. Scope
- 1.3. Intended Audience
- 1.4. Definitions, Acronyms, and Abbreviations

2. Overall Description

- 2.1. Product Perspective
- 2.2. User Classes and Characteristics
- 2.3. Operating Environment
- 2.4. Design and Implementation Constraints

3. Functional Requirements

4. Non-Functional Requirements

5. External Interface Requirements

- 5.1. User Interface

6. System Features

7. Technologies Used

Team Name: Precision Pros

1. Introduction

1.1 Purpose

This project aims to simplify the process of calculating a tip based on service quality, splitting the total bill among multiple people, and enabling fast digital payments via QR code.

1.2 Scope

The "Tip Calculator" is a web-based mini-project designed using HTML, CSS, and JavaScript. It allows users to:

- Enter the bill amount
- Rate the quality of service (on a scale of 1 to 10)
- Calculate the appropriate tip
- Split the total amount among multiple people
- Generate a QR code for convenient digital payment

The system is ideal for use in restaurants, cafés, or any group setting where a bill needs to be shared easily and fairly.

1.3 Intended Audience

- Project evaluators/instructors
- Developers

1.4 Definitions, Acronyms, and Abbreviations

- **QR:** Quick Response
 - **UI:** User Interface
-

2. Overall Description

2.1 Product Perspective

The application is a lightweight, browser-based standalone solution with no need for back-end support. It is client-side only.

2.2 User Classes and Characteristics

- **Casual Users:** Users looking to calculate and split tips quickly with minimal technical knowledge.
- **Restaurant Patrons:** Individuals or groups dining together and wanting to split the bill.

2.3 Operating Environment

- Modern web browsers: Google Chrome, Mozilla Firefox, Microsoft Edge
- Desktop or mobile devices
- Internet connectivity (for QR payment functionality)

2.4 Design and Implementation Constraints

- Built using only front-end web technologies (HTML, CSS, JavaScript)
- QR code generation may use an external API

3. Functional Requirements

Requirement ID Description

FR1	The user shall input the bill amount
FR2	The user shall select a service rating (1–10 scale) using a slider
FR3	The system shall display an emoji reaction that changes dynamically based on user's service rating
FR4	The system shall calculate the tip based on the rating
FR5	The user shall enter the number of people to split the bill
FR6	The system shall calculate and display the total bill and per person share
FR7	The system shall generate a QR code with the total bill amount for payment

4. Non-Functional Requirements

Type	Description
Performance	The calculator should provide real-time feedback upon user interaction
Usability	The UI shall be simple, responsive, and mobile-friendly

Type	Description
Portability	The application must work on all modern browsers and screen sizes
Security	The QR link (if using a third-party API) must be secure and protected
Reliability	QR generation and calculations must consistently return correct values

5. External Interface Requirements

5.1 User Interface

The interface will consist of:

- Input fields for:
 - Bill Amount
 - Number of People
- A slider for Service Rating (1–10)
 - Emoji based slider
 - Emoji reaction changes based on rating
- A **"Calculate"** button
- Output fields showing:
 - Tip Amount
 - Total Bill Amount
 - Per Person Share
- A **"Show QR to Pay"** button
 - A pop-up of QR code will be displayed

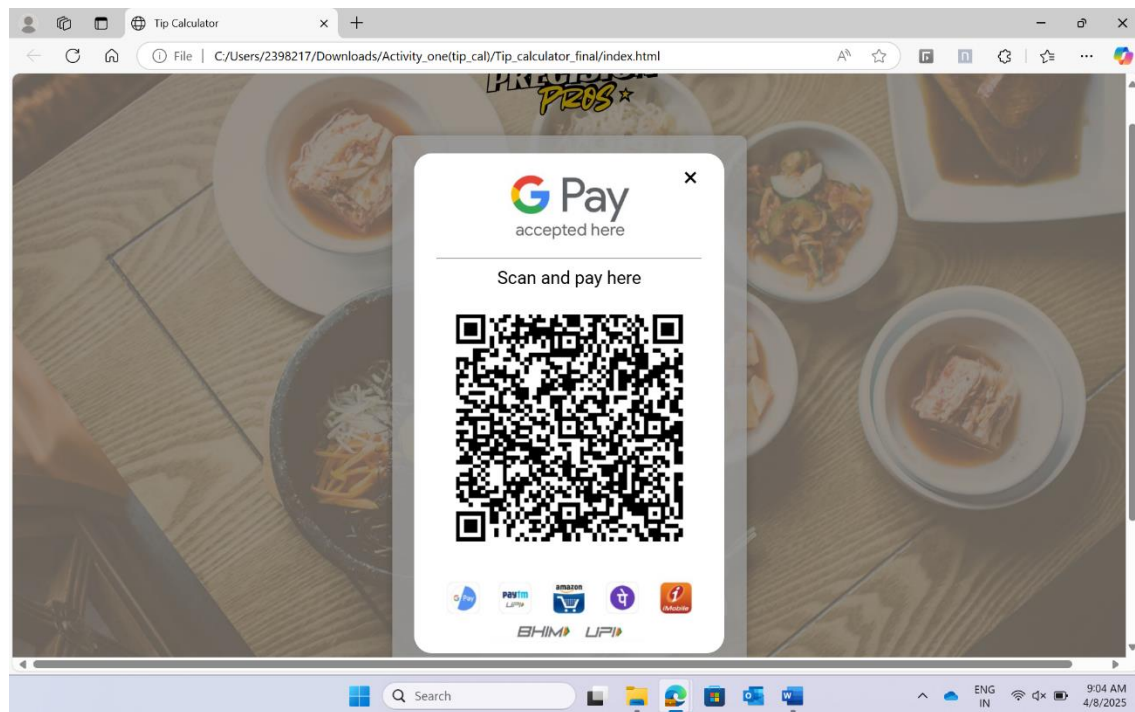
Screenshot 1: *Initial Web Page (before input is provided)*

The screenshot shows a web browser window with the title 'Tip Calculator'. The address bar shows the file path 'C:/Users/2398217/Downloads/Activity_one(tip_cal)/Tip_calculator_final/index.html'. The main content area features a background image of a wooden table with several bowls of food. Overlaid on this is a white modal box titled 'Tip Calculator' in orange. Inside the modal, there are three input fields: 'Bill Amount' with the placeholder 'Enter bill amount', 'Service Rating' with a value of '1' and a sad face emoji, and 'Number of People' with the placeholder 'Enter number of people'. Below these fields is an orange 'Calculate' button. The Windows taskbar at the bottom shows the search bar and several application icons, with the system clock indicating 9:01 AM on 4/8/2025.

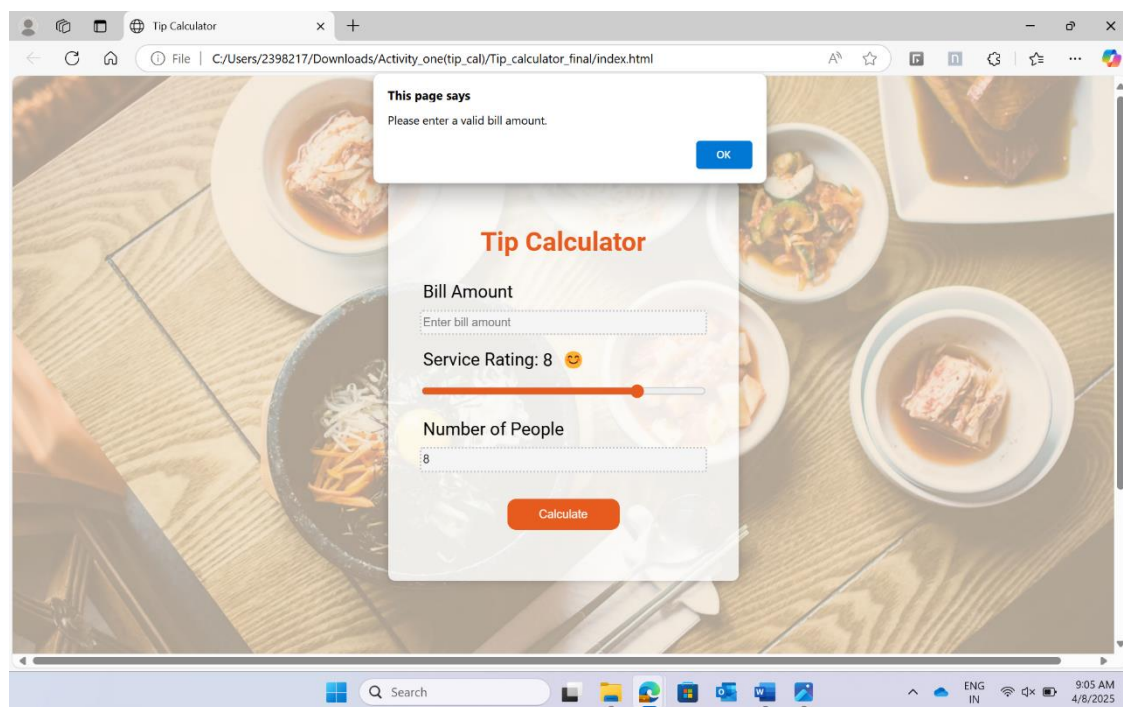
Screenshot 2: *After calculation (showing calculated values)*

The screenshot shows the same web browser window as Screenshot 1, but the modal box now displays calculated values. The 'Bill Amount' field contains the number '400'. The 'Service Rating' field shows '8' with a happy face emoji. The 'Number of People' field contains the number '5'. Below these fields, the 'Calculate' button is still present. Underneath the button, the calculated results are displayed: 'Tip Amount: 32.00', 'Total Bill Amount: 432.00', and 'Per Person Share: 86.40'. At the bottom of the modal is a new orange button labeled 'Show QR to pay'. The Windows taskbar at the bottom shows the search bar and application icons, with the system clock indicating 9:04 AM on 4/8/2025.

Screenshot 3: *QR Code screen (generated for payment)*



Screenshot 4: *Alerts for invalid input values*



6. System Features

- **Calculation Logic:** Based on service rating, the system calculates a tip (for example, rating 9 = 9% tip).
 - **Dynamic UI Feedback:** Displays calculated values without page reload.
 - **Payment Integration:** Generates a scannable QR code with total amount pre-filled.
-

7. Technologies Used

- **Frontend:** HTML, CSS, JavaScript
- **Libraries:** QRCode.js JavaScript library