**Business Requirements Document (BRD)**

**1. Introduction**

**1.1. Purpose:**

The purpose of this system is to simplify the process of calculating and sharing bills between flatmates based on the number of days each person has stayed in the flat during the billing period. It will also generate a PDF report with details of each flatmate’s payment contribution. This system is intended to automate manual calculations, reduce errors, and improve transparency among flatmates.

**1.2. Scope:**

The system is intended for flatmates who share household expenses such as rent, utilities, or other shared services. It will take inputs from the users about the total bill, the billing period, and the days each flatmate has stayed in the flat, and it will output:

* The amount each flatmate owes.
* A PDF document summarizing the information.

**1.3. Stakeholders:**

* **End Users**: Flatmates who will use the system to split bills.
* **Developers**: The software team that will build the system.
* **Project Manager**: Responsible for overseeing the project development and ensuring the requirements are met.

**2. Business Objectives**

1. **Simplify Bill Sharing**: The system must allow users to input bill amounts and automatically split the bill based on the number of days each flatmate has stayed.
2. **Automate Report Generation**: Generate a PDF report summarizing each flatmate’s payment contribution.
3. **Error Handling**: The system must validate input data (e.g., days stayed must be positive, bill amount must be a valid number, names should not be empty).
4. **Usability**: Provide a simple and user-friendly interface to enter flatmate and billing information.
5. **Modularity**: The system should be flexible enough to add more flatmates if needed in future versions.

**3. Business Requirements**

**3.1. Functional Requirements:**

* **User Input**:
  + The system must prompt the user to input the following details:
    - Total bill amount.
    - Billing period (in days).
    - Names of the flatmates.
    - Number of days each flatmate has stayed in the flat.
* **Bill Calculation**:
  + The system must calculate each flatmate’s share based on their stay duration relative to the total days of stay by all flatmates.
* **PDF Generation**:
  + The system must generate a PDF that summarizes the bill amount, the names of the flatmates, and the amount owed by each.
* **Directory Management**:
  + The system should create a directory for saving PDF files using the current date as the folder name, ensuring unique file paths.

**3.2. Non-Functional Requirements:**

* **Performance**: The system should handle input processing and PDF generation within a few seconds.
* **Error Handling**: The system must handle incorrect inputs, such as negative days or non-numeric bill amounts, and provide clear feedback to the user.
* **Reliability**: The system should create correct outputs every time, with no data loss.
* **Portability**: The system should be compatible with multiple operating systems (Windows, macOS, Linux).
* **Scalability**: The system should allow for additional flatmates in future versions.

**4. Assumptions and Constraints**

**4.1. Assumptions:**

* The system assumes that the bill amount is split based solely on the number of days each flatmate stayed.
* Users are responsible for entering correct billing period values (e.g., the total days).
* Flatmates are expected to provide truthful inputs regarding the number of days stayed.

**4.2. Constraints:**

* The initial version of the system supports only two flatmates.
* The PDF will be generated in a predefined format without user customization.

**5. Acceptance Criteria**

* The system calculates the correct bill amount for each flatmate based on their days stayed.
* The system generates a correctly formatted PDF file containing the flatmates' names and their corresponding payment amounts.
* The system handles invalid inputs (e.g., negative values, empty names) with appropriate error messages.

**6. Deliverables**

1. A working Python application that accepts user inputs and calculates bill shares.
2. A PDF generation module that creates reports for each bill.
3. Validation and error-handling functionality.
4. A user manual to guide end users through the process.