Business requirement document

Project Name: Lucky Bricks ltd.

Document history:

Version	Date	Author	Description
1.0	26-07-2024	Quba	Initial draft of the BRD
2.0	-	-	-

Table of content

Contents

1. Introduction

1.1 Purpose of the Document

The purpose of this Business Requirements Document (BRD) is to define the business objectives, scope, and requirements for the project.

1.2 Scope of the project

Lucky Bricks Ltd. is currently operating in a completely offline environment, and they have to do everything manually, such as production management, stock-management, expense management, purchase-sales management, and financial year data, which makes it difficult for them to do everything in one place, so they want to create an application that allows them to do all of their operations in one place.

2. Business Objectives

2.1 Business Goals

2.1.1 Improve operation efficiency: Streamline all business processes by automating production management, stock management, expense management, purchase-sales management, and financial year data management.

2.1.2 Enhance data accuracy and reporting: Ensure accurate and real-time data entry and reporting.

2.2 Project Objectives

2.2.1 Develop a Centralized Application: Create an integrated application to manage production, stock, expenses, purchases, sales, financial data, and vehicle management.

2.2.2 Automate Key Business Processes: Implement automation for production scheduling, inventory tracking, expense logging, and financial reporting and vehicle management through vehicle numbers.

2.2.3 Enhance Data Management Capabilities: Improve data accuracy and accessibility through real-time data synchronization and advanced reporting features.

2.3 Success Criteria

The application should consolidate all operations into a multiple platform accessible to authorized multiple person.

Enable real-time data updates and generate comprehensive reports on demand.

Achieve compliance with relevant data protection regulations and maintain high security standards.

The application should be able to successfully switch from one phase of the production cycle to another.

Enable real-time employee data, track vehicles using their registration numbers and then automatically enter the information into the sales ledger.

Trigger an automated alert when a customer's purchasing capacity is exceeded based on historical data, as well as offer the ability to customize an Excel report.

User-friendly mobile application for required certain operation.

3. Current Business Environment

3.1 Current Processes: Lucky bricks Ltd. currently operates its business processes manually. This includes selling bricks to customers, managing inventory, recording sales transactions, assigning vehicle numbers and rental details, entering fiscal year data, and booking tenders for raw materials with the government or brokers, after which payments are made via checks or cash.

3.2 challenges and Issues: Since Lucky Bricks Ltd. now runs its operations manually, occasionally the process can be extremely drawn out, time-consuming, or misleading.

The majority of their data entries are book-oriented, they encounter difficulties in viewing data entries, and the procedure takes a long time.

When they have view the data entries, they must browse through books, and occasionally some books or data are missing.
They have different brick productions at different locations therefore, it would have been very difficult for them to stock count, sales, and purchase data.

Additionally, they have to use a ticket system to supervise their vehicle, which can be a very tricky and deceptive process at times.

They have a large number of workers in a variety of positions, such as supervisor, labourer, and driver, making it difficult to keep track of their data.

They have to calculate manually customer threshold based on the customer's past sales history.

4. Proposed Solution

4.1 Description of the Solution: a distribution system allowing numerous users to view all the data of various productions in one location in real time,

a centralized system for stock counts and sales and purchase data, and a centralized database for desktop and mobile applications

effectively view credit debit reports, customer-focused debt reports, purchase expense reports other than balance sheet reports, annul reports of sales,

Individual cycle planning for their production and a one-click system for switching production batch cycle and its expenses. Total command over the production cycle,

Modules for sales and purchases include buying reports, purchasing history management, threshold alerts for clients, vehicles tracking by registration number, sales report, and more.

Automate a customer threshold alert based on historical data,

Generate automated expenditure reports with just one click that include employee personal data, pay, vehicle expenses, and other expenses,

Print all customized reports in excel format with ease.

5. Functional Requirements

5.1 Production management

5.1.1 Ability to manage current cycle

5.1.2 Ability to create new entries of production

5.1.3 Ability to switch production cycle

5.1.4 Ability to start new cycle

5.1.5 Ability to manage labour employees

5.1.6 Ability to end cycle process

5.1.7 Ability to record production unit

5.2 Purchase management

5.2.1 Ability to manage new record of purchase (do/via agent)

5.2.2 Ability to manage purchase history

5.3 sales management

5.3.1 Ability to create sales data through graphs + reports

5.3.2 Ability to create new sales entries

5.4 Stock management

5.4.1 Ability to give yearly threshold alert

5.4.2 Ability to print reports

5.4 Expense management

2. Types expenses

1) employees expense

-5.4.1Ability to create salary entries

2) Vehicle expense entries

5.4.2 Ability to create number vies vehicle expense entries

5.4.3 Ability to enter vehicle driver expense

5.5 Financial reports management

5.4.1 Ability to provide balance sheet

5.4.2 Ability to manage Purchase data reports

5.4.3 Ability to manage sales data reports

5.4.4 Ability to manage expense reports

5.4.5 Ability to manage credit and debit reports

5.4.6 Customer oriented debit sheets

5.6 Vehicle and employee management

5.6.1 Ability to note vehicle number

5.6.2 Ability to note data of employees

5.7 Reports management

5.7.1 Ability to print reports

5.7.2 Ability to provide various type of available data reports

5.8 User creation by admin

5.8.1 Ability for the user to register

5.8.2 Ability for the user to login

5.8.3 Ability for the user change password

5.8.4 Ability for the user to retrieve user id

5.8.5 Ability for the user to retrieve password

6. Non-Functional Requirements

6.1 **Distributed System**

- **Requirement:** The system should be capable of functioning as a distributed system.
- **Details:**
    - **Scalability:** The system should support horizontal scaling, allowing additional nodes to be added as demand increases.
    - **Availability:** Ensure high availability with minimal downtime, using redundancy and failover mechanisms.
    - **Data Consistency:** Implement mechanisms to maintain data consistency across distributed nodes.

6.2 User-Friendly System

- **Requirement:** The application should be intuitive and easy to use for all users.
- **Details:**
    - **Ease of Navigation:** Provide a clear and consistent navigation structure.
    - **Accessibility:** Ensure compliance with accessibility standards (e.g., WCAG) to accommodate users with disabilities.
    - **Intuitive Interface:** Design the user interface (UI) with a focus on user experience (UX) principles.

6.3 Pixel-Perfect Development

- **Requirement:** The system's user interface should be meticulously crafted to match design specifications.
- **Details:**
    - **Accuracy:** Ensure the UI matches design mockups exactly, with precise alignment, spacing, and proportions.
    - **Consistency:** Maintain a consistent design language across all screens and components.
    - **Responsiveness:** Ensure the UI is fully responsive and adapts perfectly to different screen sizes and devices.

6.4 **Robust and Scalable**

- **Requirement:** The system should be robust and scalable to handle increasing loads.
- **Details:**
    - **Load Handling:** Capable of handling high transaction volumes and user concurrency.
    - **Performance Monitoring:** Implement monitoring tools to track system performance and automatically scale resources as needed.
    - **Fault Tolerance:** Ensure the system can recover gracefully from failures with minimal impact on users.

6.5 Performance

- **Requirement:** The system should perform efficiently under various loads.
- **Details:**
    - **Response Time:** The system should have a response time of less than 400 to 600ms for any user query or action under normal operating conditions. This ensures a smooth and efficient user experience.
    - **Throughput:** Maintain high throughput for processing transactions and data operations.
    - **Resource Utilization:** Optimize resource usage to maintain performance while minimizing costs.

7. Stakeholders

7.1 List of Stakeholders

NAME	RESPONSIBILITIES
Mr. Faruk Fatehgadh	Business owner
Mr. Anas Loriya	Developer / Manager
Mr. Fardin Mansuri	Business Analyst

8. Constraints

8.1 Budgetary Constraints

The project budget is capped at 65,000/- rupees. This budget includes all costs associated with the project, such as personnel, software, hardware, and any other resources required.

8.2 Timeline Constraints

The project must be completed within a 3-month timeframe. This includes all phases of the project lifecycle: initiation, planning, execution, monitoring and controlling, and closing.

8.3 Regulatory Constraints: N/A

9. Assumptions/Dependencies

9.1 UI Acceptance

The User Interface (UI) acceptance dependency entails the approval of the UI design and functionality by the designated stakeholders. This includes ensuring that all user interface elements are correctly implemented according to the design specifications and user feedback.

9.2 Server Configuration

The server configuration dependency involves setting up and configuring the servers to host the application. This includes installing necessary software, setting up databases, and ensuring network configurations are correctly implemented.

9.3 Calculations

The calculations dependency refers to the implementation and verification of all calculations within the application. This includes ensuring that all mathematical formulas, business rules, and logic are accurately coded and tested.

10. Risks

10.1 List of Risks: N/A

10.2 **Risk Mitigation Strategies**: N/A

11. Approval

11.1 Sign-off

Approval by	Date

Sign