

---

# SOFTWARE REQUIREMENTS SPECIFICATION

for

Book Shop Automation Software

Version 1.0 approved

Presented by

Rameshwar Bhaskaran (14CS30027)

Aditya Bhagwat (14CS30002)

Group 55

IIT Kharagpur

March 27, 2016

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Purpose . . . . .	4
1.2	Document Conventions . . . . .	4
1.3	Intended Audience and Reading Suggestions . . . . .	4
1.4	Project Scope . . . . .	4
1.5	References . . . . .	4
1.6	Definitions, Acronyms and Abbreviations Used . . . . .	5
<b>2</b>	<b>Overall Description</b>	<b>6</b>
2.1	Product Perspective . . . . .	6
2.2	Product Functions . . . . .	6
2.3	User Classes and Characteristics . . . . .	7
2.4	Operating Environment . . . . .	8
2.5	Design and Implementation Constraints . . . . .	8
2.6	User Documentation . . . . .	8
2.7	Assumptions and Dependencies . . . . .	8
<b>3</b>	<b>External Interface Requirements</b>	<b>9</b>
3.1	User Interfaces . . . . .	9
3.2	Hardware Interfaces . . . . .	9
3.3	Software Interfaces . . . . .	10
3.4	Communications Interfaces . . . . .	10
<b>4</b>	<b>System Features</b>	<b>11</b>
4.1	Query for book . . . . .	11
4.1.1	Description and Priority . . . . .	11
4.1.2	Stimulus/Response Sequences . . . . .	11
4.1.3	Functional Requirements . . . . .	11
4.2	Display details . . . . .	11
4.2.1	Description and Priority . . . . .	11
4.2.2	Stimulus/Response Sequences . . . . .	11
4.2.3	Functional Requirements . . . . .	12
4.3	Increment request field . . . . .	12
4.3.1	Description and Priority . . . . .	12
4.3.2	Stimulus/Response Sequences . . . . .	12
4.3.3	Functional Requirements . . . . .	13

4.4	Request for new book . . . . .	13
4.4.1	Description and Priority . . . . .	13
4.4.2	Stimulus/Response Sequences . . . . .	13
4.4.3	Functional Requirements . . . . .	13
4.5	Update inventory . . . . .	13
4.5.1	Description and Priority . . . . .	13
4.5.2	Stimulus/Response Sequences . . . . .	13
4.5.3	Functional Requirements . . . . .	13
4.6	Add to cart . . . . .	15
4.6.1	Description and Priority . . . . .	15
4.6.2	Stimulus/Response Sequences . . . . .	15
4.6.3	Functional Requirements . . . . .	15
4.7	View statistics . . . . .	15
4.7.1	Description and Priority . . . . .	15
4.7.2	Stimulus/Response Sequences . . . . .	15
4.7.3	Functional Requirements . . . . .	15
4.8	View requests . . . . .	15
4.8.1	Description and Priority . . . . .	16
4.8.2	Stimulus/Response Sequences . . . . .	16
4.8.3	Functional Requirements . . . . .	16
<b>5</b>	<b>Other Nonfunctional Requirements</b>	<b>17</b>
5.1	Performance Requirements . . . . .	17
5.2	Safety Requirements . . . . .	17
5.3	Security Requirements . . . . .	17
5.4	Software Quality Attributes . . . . .	17
5.5	Business Rules . . . . .	17

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the “Book Shop Automation Software” (BAS) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. Unless otherwise stated, all requirements specified here are high priority and committed for release 1.0.

## 1.2 Document Conventions

The document convention is very simple. All major section headings are in bold . Hyperlinks are indicated by blue bounding boxes.

## 1.3 Intended Audience and Reading Suggestions

This document is meant for the customers and the employees of the Book Shop. The document describes the workflow of the software and hence is extremely essential for all users to be well-acquainted with this document

## 1.4 Project Scope

In today’s world, there arises a need to get things done quickly and efficiently. Customers find it difficult at times to find a book from book shops close to them . At the same time, the book shop owner wouldn’t want to take a risk of losses due to stagnant stock. This issue is resolved by a neat mutually beneficial way by the software. It strives to be a fully independent entity with all necessary features to handle transactions and queries in a book shop. Also a need to automate tasks which need to be performed daily arises. In this case, the software reduces the risk and need of a third party to manually perform these mundane tasks.

## 1.5 References

- IEEE standard [template](#) for IEEE standard
- Fundamentals of Software Engineering by R.Mall

## 1.6 Definitions, Acronyms and Abbreviations Used

- **BAS** Book Shop Automation Software
- **SRS** Software Requirements Specifications
- **ISBN** International Standard Book Number is a unique 13 digit code for each book. Contains information related to Title, Publisher and Group etc. Different Editions will have different ISBN and ISBN for different copies of same book will be same.

## 2 Overall Description

### 2.1 Product Perspective

The BAS is an independently functioning system that automates manual tasks in a Book Shop like answering customer queries, billing, making sales statistics, planning for further orders etc. The system handles and stores all necessary records of books in a database. The user interface is simple for any user to understand when he/she uses it for the first time itself . The software also analyses data and presents it in a useful form for any important stakeholder like the owner and his business associates to take important decisions.

### 2.2 Product Functions

The BAS performs these major tasks:

- **Query for books:**

The customer can query for availability of the book by entering its title or the author's name. On availability , the BAS shows the rack number and the number of copies in stock.

- **Request for book:**

If the query for the book fails, then the customer has an option of requesting for the book shop to order the book in question by entering the full details of the book (Name, Author's name, ISBN code, Publisher's name ). Depending on the number of requests, the Manager may decide to order the book.

- **Update stock and inventory:**

When the customer confirms the book, the BAS updates the inventory. Also in case of new arrivals/defective pieces, the BAS updates accordingly.

- **View requests:**

The Manager can view the number of requests for the books and decide on the course of action to be taken.

- **Generate sales receipt:**

To complete the transaction, the BAS generates a sales receipt for endorsing the transaction.

- **Generate sales statistics:**

The BAS generates sales statistics based on transactions between any given period. This feature is used by the owner and any other authorised person. This also helps in finding the inventory level.

- **Print the list of books to be bought depending on the inventory level:**

Every day the book shop owner would give a command for the BAS to print the books which have fallen below the threshold and the number of copies to be procured along with the full address of the stockist.

## 2.3 User Classes and Characteristics

Main class has the access to information as it contains aggregations of many class objects and is the most privileged in the sense of access. Cart is used very frequently in conjunction with SalesClerk to buy books and print receipts. This is the most general function of the software.

The user classes in this project are :

- **Book :**

This class is central to this project. It stores those fields that characterize a Book object such as it's name, author's name, price, publisher, which rack it is kept on and so on. This implies that there can only be one book object per name - there will be one instance of the Book object "The Alchemist" irrespective of the number of copies of that book available. It's member functions give access to it's fields and allow us to change some mutable fields like number of copies available. This is the biggest class and holds the most amount of information.

- **Employee :**

This class is derived multiple times to create sub-classes like sales clerk, manager and owner . These sub-classes allow us to implement the organizational hierarchy of the book shop.

- **SalesDay :**

This class allows us to maintain an account of all the sales that take place during working day.

- **NotInCollection :**

NotInCollection is the order request that contains the details of the book as specified by the customer.

- **Cart :**

A customer may choose more than one book to buy in one transaction (and more than one copy per book). The cart maintains the list of books he/she wishes to buy.

- **Main:**

This is the main application class that binds everything together. It contains the arrays of Book objects and registered employees. It also contains information regarding the books that have fallen below the threshold stock level and the books that are in demand but not a part of the collection. The sales statistics are also maintained in this class.

- **SalesDay:**

This is a very important class that stores the sales statistics for the day. Useful to get an overall picture of the sales statistics of a period and derive useful information from it.

- **Other auxiliary classes :**

Publisher and vendor classes have been introduced to better implement the Book class. Each vendor may house books of multiple publishers. Many books may have the same publisher.

## 2.4 Operating Environment

This software is built in Java and will run on Linux ,Windows and any other platforms which have a JVM pre-installed.As for hardware, it may require a printer to print out the sales receipt generated.

## 2.5 Design and Implementation Constraints

As the BAS needs to handle large number of books, that may lead to lagging of the system to handle data from database. A linear time algorithm may itself lose efficiency in such cases. Logarithmic time algorithms can be used ,but they will compromise on the memory used.

## 2.6 User Documentation

As the product is presently in prototype stage, the documentation is not available. Once the full software is built, the user documentation may be presented In html or pdf formats.

## 2.7 Assumptions and Dependencies

We have assumed that the computer used in the book shop is perfectly working and has a perfectly working Java runtime environment and JVM installed. As the software is built using Java, it is safely assumed that appropriate software is installed.

Coming to the problem statement, in order to give a more general perception to the



problem in hand, there are other types of employees other than the Manager and Sales Clerk.

## 3 External Interface Requirements

### 3.1 User Interfaces

The Book Shop Automation software is implemented on a single application window. There are multiple simultaneous users of the software so the interface necessarily has to cater to both the customer and the employee logged in currently. On the (default) main screen the logged in employee's name and job profile is displayed at the top right. There are 2 buttons for login and logout. Clicking the login button results in a login window popping up. Logging in makes us return to the main screen. The main screen will, most importantly, consist of a search bar on the top with a label that asks the user to enter either the author's name or the book title. There is space for displaying relevant search results below this search bar. Depending upon the employee logged in there will be buttons on the bottom to the left (outside the search result space) pertaining to his/her responsibilities. Clicking these buttons may pop windows (or change the panel in the same window) to perform required activities.

After the user enters the search query and results are displayed he has the option to view the book which redirects him to a window where he can view it's details. Here he can choose the number of copies he wishes to buy, and he returns back to the search page. The books chosen are added to the 'My Cart' sub-panel on the right of the main screen. There will be a button to finalize cart and proceed to payment which will change the screen. At this point the sales clerk will take over. Here, he will enter the ISBN code of each book and finally press the button to complete the transaction and print out the bill and return to the main screen again.

If the queried book has run out of stock or has insufficient stock to complete the transaction, the customer can create a new book request or increment a pre-created request. If the search result is empty, then the customer is given the option to enter the relevant details of the desired book and create a new-book-request. The manager can view both these requests by clicking a button visible to him on the main screen, which redirects to a new screen. The owner can view sales statistics similarly.

### 3.2 Hardware Interfaces

As indicated before ,the only external hardware the software needs is a printer. This can be accessed using the Java 2D printing API or any other suitable API.

### **3.3 Software Interfaces**

This software is a standalone application with the its own user interface handling user interaction fully. When necessary data needs to be retrieved/added/modified, then a connection is made to the database via a driver. All tools and libraries used in the software are open-source.

### **3.4 Communications Interfaces**

The software doesn't require internet to function for its requirements. However it may require internet to implement certain features like a global signin/signup option wherein the users may be spread over. In such cases, the internet connection must be present and must allow TCP (Transfer Control Protocol) and HTTP.

## 4 System Features

The BAS is a multi-faceted software that supports multiple features. The following are some of the important features that are implemented.

### 4.1 Query for book

The main objective/highlight of the software is to enable query for the books.

#### 4.1.1 Description and Priority

High priority feature. The search used is linear ,hence the software may lag behind for large number of books.

#### 4.1.2 Stimulus/Response Sequences

The user(customer) will enter the name of the author/title of the book to get the list of matching books.

#### 4.1.3 Functional Requirements

The feature requires a database connection and provision to get user input .The user needs to enter the appropriate keywords to get the book. To help with suggestions, the minimum edit distance algorithm and various string matching algorithms are deployed.

### 4.2 Display details

This feature helps display the details of the book like title,rack number and the number of copies left.

#### 4.2.1 Description and Priority

High priority feature. This will decide whether the user needs to buy the book and carry out the transaction.

#### 4.2.2 Stimulus/Response Sequences

This feature is used only when the user presses the query button as well as the database has a matching entry with the user input.

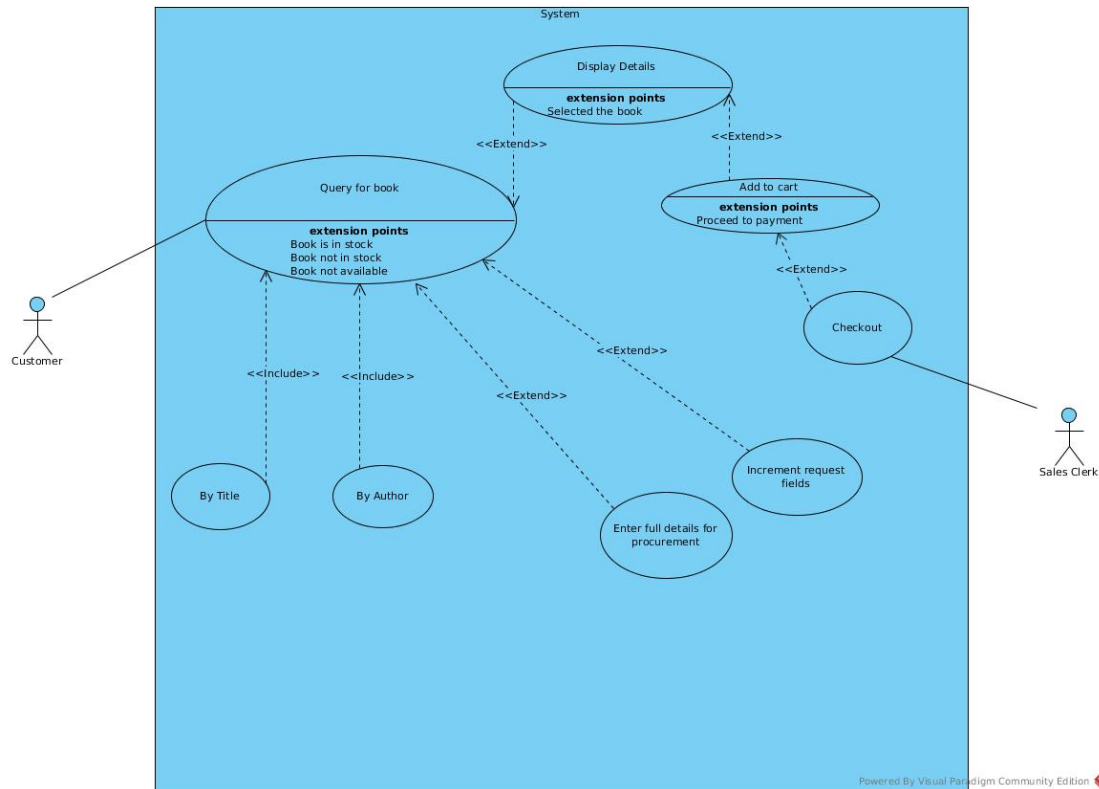


Figure 4.1: Use case diagram of the software.

### 4.2.3 Functional Requirements

The feature requires a database connection and an appropriate UI wherein the details can be displayed.

## 4.3 Increment request field

This feature helps requesting the manager to order more copies of the book that is presently not in stock.

### 4.3.1 Description and Priority

High priority feature. This would not only help the shop improve its business, but also would help prevention of stagnant stock.

### 4.3.2 Stimulus/Response Sequences

This feature is used only when the user presses the query button and the database has the book record but number of its copies is 0. Then the UI further shows a dialog box

wherein the user is offered to increment request field.

#### **4.3.3 Functional Requirements**

The feature requires a database connection and an appropriate UI wherein the details can be displayed.

### **4.4 Request for new book**

This feature helps requesting the manager to order more copies of the book that hasn't been sold yet.

#### **4.4.1 Description and Priority**

High priority feature. This feature is important for improving the shop's inventory.

#### **4.4.2 Stimulus/Response Sequences**

This feature is used only when the user presses the query button and the database doesn't have such a book record. In such a case, the software takes in customer input for details of the book like the ISBN code and name. This is forwarded.

#### **4.4.3 Functional Requirements**

The feature requires a database connection and an appropriate UI wherein provision for user input is present.

### **4.5 Update inventory**

This feature helps employees to update stock details once stock arrives/transactions are made.

#### **4.5.1 Description and Priority**

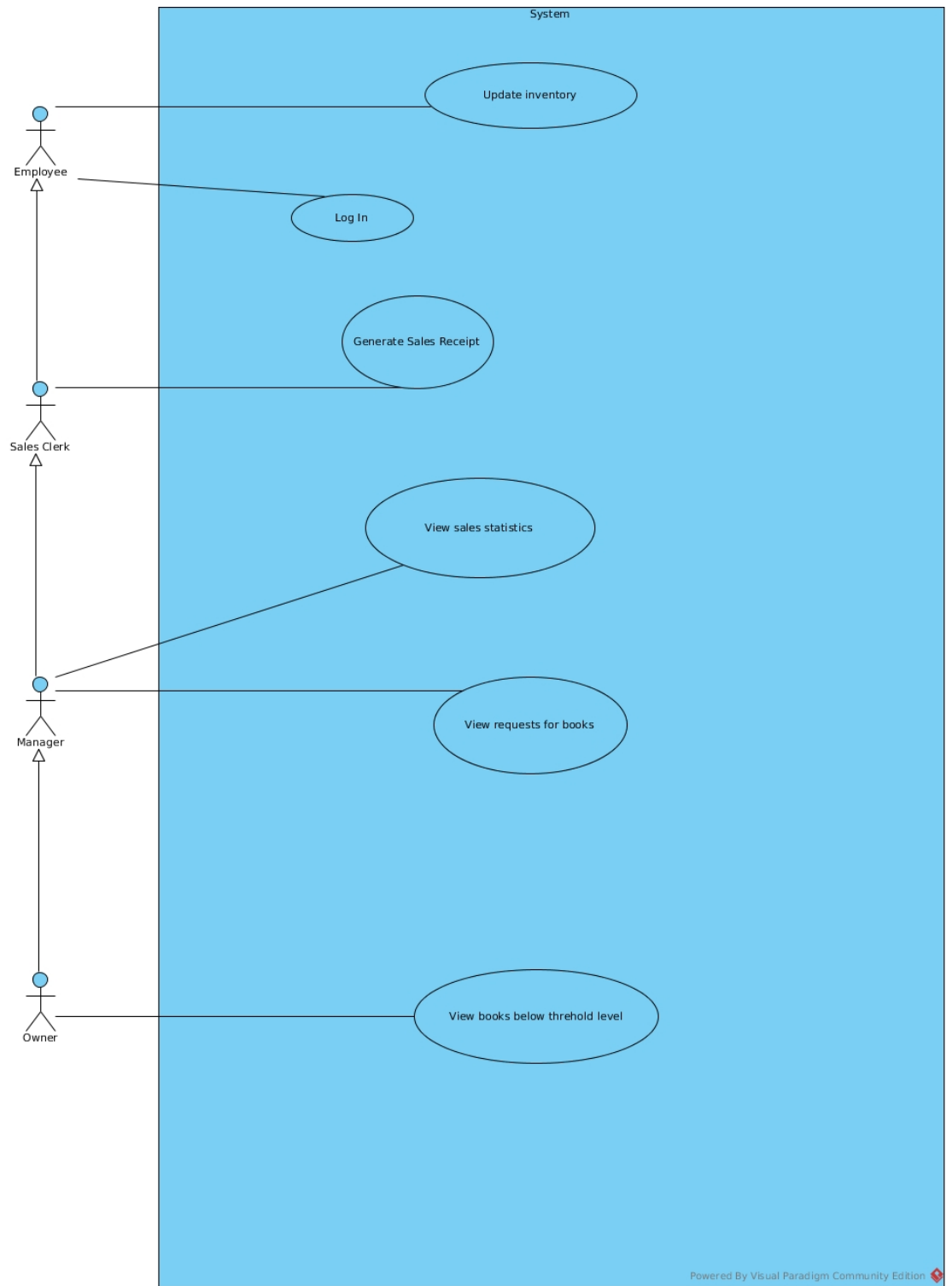
High priority feature. This feature is integral for getting the inventory updated. This feature is crucial for every other major feature to work right.

#### **4.5.2 Stimulus/Response Sequences**

This feature is used only when new stock has arrived. Also it is used in cases where books are sold or is defective.

#### **4.5.3 Functional Requirements**

This feature requires a database connection and only employees of the shop are allowed to change the inventory.



Employee.jpg

Figure 4.2: Use case diagram of the software.

## **4.6 Add to cart**

This feature helps customers to add books to their cart to get the transactions done together.

### **4.6.1 Description and Priority**

Medium priority feature. This feature is a important add-on to improve user experience. However its absence won't affect the basic functioning of the software.

### **4.6.2 Stimulus/Response Sequences**

This feature is used only when new stock has arrived. Also it is used in cases where books are sold or is defective.

### **4.6.3 Functional Requirements**

This feature requires a database connection and only employees of the shop are allowed to change the inventory.

## **4.7 View statistics**

This feature helps the owner and manager to view sales statistics to make important decisions.

### **4.7.1 Description and Priority**

High priority feature. It is used to make the sales data in a more comprehensible and better fashioned way.

### **4.7.2 Stimulus/Response Sequences**

This feature is invoked when the manager/owner clicks the "View Statistics" button.

### **4.7.3 Functional Requirements**

This feature requires a database connection and only the managers and the owner are allowed to use this feature. Also this feature may require external graphing libraries or plot visualising tools.

## **4.8 View requests**

This feature helps the manager to review both incremental requests and well as requests for new books.



#### **4.8.1 Description and Priority**

High priority feature. It helps in analysing the stock needed for procurement.

#### **4.8.2 Stimulus/Response Sequences**

This feature is invoked when the manager/owner clicks the "View incremental requests" button and "View new book requests" button.

#### **4.8.3 Functional Requirements**

This feature requires a database connection and only the managers and higher authorities are allowed to use this feature.

## **5 Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

The software is supposed to be designed for a small to medium sized book shop. The book shop is assumed to house a few thousand books (or less) at any given point in time. Hence search results are supposed to be almost instantaneous. This goes for all the operations that a customer may perform, including updating the cart and requesting for a new book. The same can be said about printing sales receipt and viewing order requests. Displaying the sales statistics is the only computationally heavy operation and may potentially take a few minutes but only if the period entered is too large, say 10 years.

### **5.2 Safety Requirements**

All exceptions have to be properly handled so that the application doesn't crash under any circumstances and no data is lost. Because the software requirements necessitate the operation of a cron job, that operates on data generated during a day, crashes can potentially lead to a great losses of information to the owner. Safeguards must be put in place so that this never happens.

### **5.3 Security Requirements**

The BAS handles very sensitive data including personal details of employees and customers and details of transactions. Such data needs to be suitably 'abstracted' to prevent misuse. No external software should be given access to the internal parts/modules of the software, as that may lead to its misuse.

### **5.4 Software Quality Attributes**

The software must include user authentication for all types of operation besides operation by the customer. Different classes of employees have different accessibility privileges and this must be strictly upheld for the data to remain safe.

### **5.5 Business Rules**

The software must have wide adaptability and reusability. Many book shops choose to also sell music CDs, DVDs and cassettes. Some choose to also keep magazines and

journals depending on their customer base. The software must be modularized enough to allow for this. Since it will be operated on by non-professionals (both the lay employee as well as customers) ease of use is vital. It must be robust and maintainable as it is possible that it may be used continuously for many years.