

Abstract

The Shares and Stock details is a mini project that provides a user-friendly, interactive Menu Driven Interface (MDI) based on local file systems. All data is stored in files on disk. This mini project uses file structures to access the files. A file structure is a combination of representations for data in files and operations for accessing the data. A file structure allows applications to read, write, and modify data.

This system is built for fast data processing and display for Companies or their Customers. The database is a collection of company name, customer name and other stock specific data. A customer, when added to the database contains his/her shares in the companies associated to those shares. The mini project is built to help companies and its customers keep track of their shares and stocks. It consists of an effective and easy interface to help the user store and view the data.

The identity of the fields is preserved by separating them with delimiters. Fixing the number of fields in a record does not imply that the size of fields in the record is fixed. The records are used as containers to hold a mix of fixed and variable length fields within a record.

We use a B-Tree of indexes to keep byte offsets for each record in the original file. The byte offsets allow us to find the beginning of each successive record and compute the length of each record. A B-Tree of simple indexes on the primary key is used to provide direct access to data records. Each node in the B-Tree consists of a primary key and reference pair of fields. The primary key field is the SID field while the reference field is the starting byte offset.

Acknowledgment

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management of RNS Institute of Technology** for providing such a healthy environment to carry out this mini project work. We would like to thank Our beloved Director **Dr. H N Shivashankar** for his confidence filling words and support for providing facilities throughout the course.

We would like to express our thanks to our Principal **Dr. M K Venkatesha** for his support and inspiring us towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. M V Sudhamani**, Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our mini project work.

We would like to express our profound and cordial gratitude to our guide **Mr. Chethan J**, Assistant Professor, Department of Information Science and Engineering for his valuable guidance in preparing the mini project report.

We would like to express our profound and cordial gratitude to our coordinator **Mrs. Vinutha G K**, Assistant Professor, Department of Information Science and Engineering for her valuable guidance, constructive comments and continuous encouragement throughout the mini project work.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped us carry out the mini project work.

And lastly, we would hereby acknowledge and thank our parents who have been a source of inspiration and also instrumental in carrying out this mini project work.

SHREYAS DL - USN: 1RN16IS098

TEJAS DS - USN: 1RN16IS113

TABLE OF CONTENTS

CERTIFICATE	
ABSTRACT	i
ACKNOWLEDGMENT	ii
TABLE OF CONTENTS	iii
LIST OF FIGURES	vi
LIST OF TABLES	vii
ABBREVIATIONS	viii
1. INTRODUCTION	1
1.1 Introduction to File Structures	1
1.1.1 History	1
1.1.2 About the File	2
1.1.3 Various kinds of storage of Fields and Records	3
1.1.4 Application of File Structure	7
2. SYSTEM ANALYSIS	8
2.1 Analysis of Program	8
2.2 Structure used to store the fields and records	8
2.2.1 Storing Fields	8
2.2.2 Storing Records	8
2.3 Operations Performed on a File	9
2.4 Indexing used	10
3. SYSTEM DESIGN	12
3.1 Design of fields and records	12
3.2 User Interface	12

3.2.1 Insertion of a Record	13
3.2.2 Display of a Record	13
3.2.3 Deletion of a Record	14
3.2.4 Searching of a Record	14
3.2.5 Modification of a Record	14
3.2.6 Design of Index	15
4. IMPLEMENTATION	16
4.1 About C++	16
4.1.1 Classes and Objects	16
4.1.2 Dynamic Memory Allocation and Pointers	16
4.1.3 File Handling	16
4.1.4 Character Arrays and Character functions	17
4.2 Pseudo code	17
4.2.1 Insertion Module Pseudo code	17
4.2.2 Display Module Pseudo code	19
4.2.3 Deletion Module Pseudo code	20
4.2.4 Search Module Pseudo code	21
4.2.5 Modify Module Pseudo code	23
4.2.6 Indexing Pseudocode	25
4.3 Testing	26
4.3.1 Unit Testing	26
4.3.2 Functional Testing	27
4.3.3 Integration Testing	28
4.3.4 System Testing	31
4.3.5 Acceptance Testing	32
4.4 Discussion of results	33

4.4.1 Menu Options	33
4.4.2 Insertion	34
4.4.3 Deletion	34
4.4.4 Searching a Record	35
4.4.5 Modification	35
4.4.6 Display	36
4.4.7 File Contents	36
4.4.8 Index File Contents	37
4.4.9 B-Tree Structure Display	37
5. CONCLUSION AND FUTURE ENHANCEMENT	39
REFERENCES	40

LIST OF FIGURES

Figure No.	Description	Page No.
Figure 1.1	Four methods for field structures	04
Figure 1.2	Making Records Predictable number of Bytes and Fields	06
Figure 1.3	Using Length Indicator, Index and Record Delimiters	07
Figure 3.1	Class Shares	12
Figure 3.2	User Menu Screen	13
Figure 3.3	Class Node	15
Figure 3.4	Class B-tree and Index	15
Figure 4.1	User Menu Screen	33
Figure 4.2	Insertion of a Record	34
Figure 4.3	Deletion of a record	34
Figure 4.4	Search for a record	35
Figure 4.5	Modification of a record	35
Figure 4.6	Modification of a record	36
Figure 4.7	Display	36
Figure 4.8	Data File Contents	37
Figure 4.9	Index File Contents	37
Figure 4.10	B Tree Structure	38
Figure 4.11	Tree Node	38

LIST OF TABLES

Table No.	Description	Page No.
Table 4.1	Unit Test Case for share_id input check	26
Table 4.2	Unit Test Case for Customer_Name input check	27
Table 4.3	Functional testing Test cases for Shares and Stocks management	27
Table 4.4	Integration testing Test cases for Shares and Stocks management	29
Table 4.5	System testing Test cases for Shares and Stocks management	31

ABBREVIATIONS

MDI	-	Menu Driven Interface
UI	-	User Interface
BST	-	Binary Search Tree
AVL	-	Adelson-Velskii and Landis
BT	-	Balanced Tree