**Employee Id:** 2150842

**Name:** Divya Udayakumar

**Project - Assignment I**

**SECTION 1**

1. Preferred Data Structure to hold all the necessary data for card even for add additional fields for efficiency is Linked List with singly linked list mechanism with hashing.

By this Linked list technique, we have Dynamic size, Efficient memory allocation/utilization and quick insertion of additional fields.

Head

Address of next node

Address of next node

NULL

Address of next node

Value

Value

Value

**Linked List Structure**

The list is not required to be contiguously present in the memory. The node can reside anywhere in the memory and linked together to make a list. This achieves optimized utilization of space. list size is limited to the memory size and doesn't need to be declared in advance. We can store values of primitive types or objects in the singly linked list.

3. \*Time Complexity:

Linked lists have most of their benefit when it comes to the insertion and deletion of nodes in the list. Unlike the dynamic array, insertion and deletion at any part of the list takes constant time.

|  |  |  |
| --- | --- | --- |
| Sl.no | Operation | Complexity |
| 1 | Accessing element | O() |
| 2 | Traverse all elements | O() |
| 3 | Insertion (Current) | O (1) |
| 4 | Deletion (Current) | O (1) |
| 5 | Insertion (Front) | O (1) |
| 6 | Insertion (End) | O() |

\*Space Complexity:

The Space Complexity of the above Linked List operations is O(1).This is because we do not need extra space beyond a fixed number of variables. For some operations, you may need extra space of the order of O(N). For example, sorting a Linked List using a sorting algorithm that is not in-place.

\*Efficiency of Memory:

The efficient way of representing a linked list is using the free pool of storage. During the creation of a linked list, whenever a node is required the request is placed to the memory; the memory will then search the memory space.

1. Data structure that will hold all customer data in a sorted order, the complexity will be less. I prefer linked list for this sorted order. Because during sorting the data in increasing order it is easy to accessing using singly linked list.

For example, the implementer cannot scramble the order to improve efficiency. An ordered list is a list in which the order of the items is significant. However, the items in an ordered list are not necessarily sorted. Consequently, it is possible to change the order O items and still have a valid ordered list.

Permanent Storage Area

Stack

Heap

|  |
| --- |
| Local Variables |
| Free memory |
| Global Variables |
| C Program Instruction |

**Memory Allocation Process**

1. Indexing criteria will be the account number because each user is provided with an account number the process of registration stage. If the indexing is name or card number there is a difficult with searching and process the data. Not naming and Card number because similar naming and card may be lost so it may cause a problem when the user buys new cards.
2. Banks use credit card number as in index my choice if they go with account number with bank it may not cause any problem over the indexing of the card. As we have implemented the data while the registration process. So, index can be may as account number.
3. Banks also use a person’s name to index because of this case if bank as naming as an indexing many persons have similar name it may create a data with duplicate records which may create a run-time error when it is implemented.
4. If you have to give a facility that a customer can be indexed with either his name or card number, in some cases if card number and naming becomes similar it may cause an issue at the run-time which may cause a duplicate record which may cause a crashing of data in the process. So, card number is maximum prefer for the linked list-based algorithms.

struct acc\_type

{

char acc\_holder\_name[50];

int Date\_of\_birth;

char acc\_holder\_address[100];

char acc\_email[30];

long card\_no;

int card\_issuse;

int card\_expiry;

int cvv;

};

**SECTION 2**

1. American express uses 15 digits for its credit card. If a hypothetical bank MeToo Bank issues credit cards that can be 15 digits or 16 digits. Yes, selected data structure provides a valid result over the process. Only few records of the credit card are made changed for the credit card number where the value of card from the bank side is changed for this type of cards.

|  |
| --- |
| + Retailer\_id  +Collect\_Detail  +Upload\_Details |
| +Upload  +View  +Store |

|  |
| --- |
| +Customer\_name  +Card\_number  +Purchase\_details  +View\_details |
| +Register  +Login  +View  +Download  +Upload  +Logout |

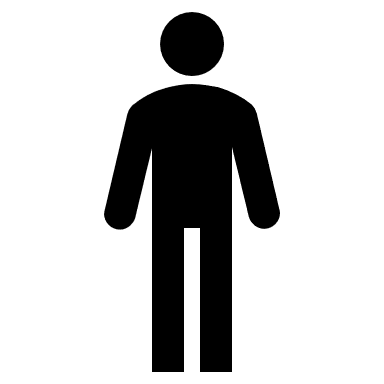
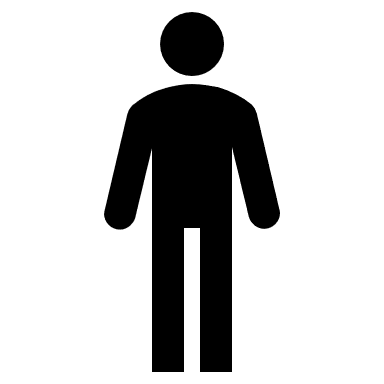
**Retailer**

|  |
| --- |
| + Banker  +Verify\_data  +Upload\_files |
| +Encryption  +Verification  +Upload |

**Customer Bank**

**SECTION 3**

1. If you have a memory constrained device, the data record of the currently developed structure member is reduced to some memory of space where the data of unnecessary information is cut-off and the necessary information is put to the memory constrained devices.

**Block of the project**

Paying Bills

Prepare the Bills

Purchase the item

Issuing Card

Bank Enquiry

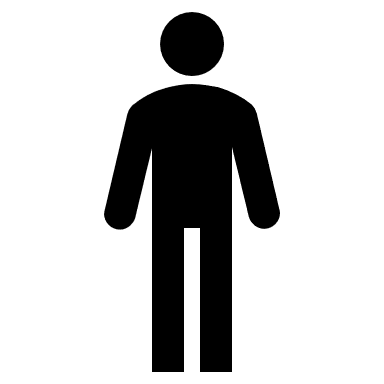
Credit Card Request

Creating Account

Retailer

Bank

Customer

****