SSN COLLEGE OF ENGINEERING(Autonomous)
Affiliated to Anna University,Chennai
DEPARTMENT OF CSE
UCS 1312-DATA STRUCTURES LABORATORY
MINI PROJECT: WATER TRANSPORT MANAGEMENT

LIKHITHA VERMA -185001084
PRASANNA KUMARAN D -185001110
PREETHI THIYAGRAJAN -185001116

Project Abstract : We live in a world where water is a scarce resource and not all citizens are met with the basic water needs. Our project aims to deliver water to the citizens at the shortest time and cheapest way possible. We have used Djikstra's Algorithm to solve the problem of finding the shortest route to the destination . We have the user module in which one can book water, view booking status and print the bill. The Admin module contains two validation options such as validating registration and confirmation of order, the Admin can add a route between two locations which provide water delivery services.

ADTS Involved:

" **djkstra_list_heap.h** " :- This ADT is used to process the data and calculate the shortest distance from the origin and every other location where a path exists.We have implemented Djikstra's Algorithm using adjaceny list and Minheap to find the minimum distance in every iteration.

" **queue.h** " :- This ADT is used to queue the user onto a list.Queue data structure is used as it is first in first out data structure and the user who booked first is given the top priority and the priority decreases progressively as the queue expands.

" **mini.c** " :- This file contains the main function, user/admin modules and uses files to manage the data that is being entered and processed.

Core concepts and Data Structures Involved:

* Linked List
* Graph Data Structure
* Priority Queue using MinHeap
* File Handling
* Queue
* Pointers and Arrays

<u>Functions Description :</u>

“djkstra_list_heap.h “

Structures:

AdjacenyNode	**:** distance(int) , destination(int) , self referential structure variable.
Graph	: nodes(int) , AdjList(AdjListNode**)
MinHeapNode	: distance(int), vertex(int)
MinHeap	: size(int), capacity(int), pos(int*), array(MinHeapNode**)

AdjacencyNode structure stores two variables distance,destination of type int and self referential structure (next) to point out to the next connected node in the network.

Graph structure stores the number of nodes in a network and an array of AdjListNodes.

MinHeapNode structure stores the value of the distance to the next vertex.

MinHeap structues defines the number of nodes present in the heap. Size to keep track of the current size and array to store the MinHeapnodes at a position.

Functions:

AdjListNode* newAdjListNode(int, int) : This function is used to create a new node in the network by assigning distance and destination to the node.

Graph* createGraph(int) : Creates a graph by defining number of nodes in the network and allocating size of each AdjList and initializing it to NULL.

Void addEdge(Graph*,int ,int,int): This is used to point the nodes to the corresponding adjList.Since the graph is undirected the source node is pointed to the destination adjList and vice versa.

MinHeapNode* newMinHeapNode(int, int) : Initializes the value of distance and destination for a new node in MinHeap.

MinHeap* createMinHeap(int) : Defines the size of the heap and initialize the variables in the structure.

Void swapNode(MinheapNode** ,MinheapNode**) : swaps two nodes in a heap.

Void MinHeapify(MinHeap* , int) : processes data based on the index passed and the heap properties are retained .

int checkEmpty(MinHeap*) : returns 0 if Heap is empty,1 otherwise.

MinHeapNode* extractMin(MinHeap*) : extracts the minimum value (root) from the heap, swaps the last node with the first and calls MinHeapify() to Heapify the resulting heap.

Void DecreaseKey(MinHeap*, int, int) : The child nodes are repeatedly swapped with the parent node if the child node is smaller than the parent node(distance).

Int* Djikstra(Graph* , int) : Uses the heap that was constructed for the given nodes and implements Djikstra's algorithm to calculate the minimum distances from the source to various destinations . The minimum distance in every iteration is obtained from the minheap array.

" **Queue.h** "

Structures:

User : stores user details.
Queue : front,rear,size,capacity(int), user* .

Functions:

Queue* createQueue(int) : initializes the Queue.

Int isFull(Queue*) : returns whether the queue is full or not.

Int isEmpty(Queue*) : returns 1 if queue is empty , 0 otherwise.

Void Enqueue(Queue*, user*) : enqueues the user structure into a queue.

User* Dequeue(Queue *) : performs dequeue operation and removes the first element in the queue.

" mini.c "

Functions:

void Bill(user) : Calculates the bill of the user .

}

Void put_details(user) : displays user details.

Void admin(Graph*) : the admin uses this module to procees registration , add edge between two locations and confirm order. The user data is written onto a file after every operation .

Void getUser(char*): stores the information of the new user.

Void login(Graph*) : validates the username and password entered by the user. Admin is a given username and password as admin.

Int regis() : registers a new user and writes the user details onto a file.

PROGRAM:

CONTENTS OF djkstra_list_heap.h:

```
// ADT FOR DJIKSTRAS ALGORITHM USING ADJACENY LIST AND MIN
HEAP !
#include<stdio.h>
#include<stdlib.h>
struct AdjListNode
{
	int distance;
	int destination;
	struct AdjListNode * next;
};
typedef struct Graph
{
	int no_of_nodes;
	struct AdjListNode** AdjList;
}Graph;
struct AdjListNode* newAdjListNode(int dest , int dist)
{
	struct AdjListNode* newNode= (struct AdjListNode*)malloc(sizeof(struct
AdjListNode*));
	newNode->destination = dest;
	newNode->distance = dist;
	newNode->next = NULL;
	return newNode;
}
struct Graph* createGraph(int nodes)
```

```
{
    struct Graph* graph = (struct Graph*)malloc(sizeof(struct Graph*));
    graph->no_of_nodes = nodes;
    graph->AdjList = malloc(sizeof(struct AdjListNode*)* nodes);
    for(int i = 0; i<nodes; i++)
        graph->AdjList[i] = NULL;
    return graph;
}
void addEdge(struct Graph* graph,int source, int destination, int distance)
{
    struct AdjListNode* newNode = newAdjListNode(destination, distance);
    newNode->next = graph->AdjList[source];
    graph->AdjList[source] = newNode;

    newNode = newAdjListNode(source, distance);
    newNode->next = graph->AdjList[destination];
    graph->AdjList[destination] = newNode;

}
struct MinHeapNode
{
    int distance;
    int vertex;
};
struct MinHeap
{
    int size;
    int capacity;
    int* pos;
    struct MinHeapNode **array;
};
struct MinHeapNode* newMinHeapNode(int v,int dist)
{
    struct MinHeapNode* newNode = (struct
MinHeapNode*)malloc(sizeof(struct MinHeapNode));
    newNode->vertex = v;
    newNode->distance = dist;
    return newNode;
}
struct MinHeap* createMinHeap(int capacity)
{
    struct MinHeap* heap = (struct MinHeap*)malloc(sizeof(struct MinHeap));
    heap->size = 0;
    heap->capacity = capacity;
    heap->pos = (int*)malloc(sizeof(int)* capacity);
    heap->array = (struct MinHeapNode**)malloc(sizeof(struct
MinHeapNode*)* capacity);
    return heap;
}
void swapNode(struct MinHeapNode** a , struct MinHeapNode** b)
{
    struct MinHeapNode* temp = *a;
    *a = *b;
    *b = temp;
}
void MinHeapify(struct MinHeap* heap,int index)
{
    int small,left,right;
    small = index;
    left = (2 * index) + 1;
    right = (2* index) + 2;
    if( left < heap->size && heap->array[left]->distance < heap->array[small]-
>distance)
        small = left;
    if( right <heap->size && heap->array[right]->distance < heap-
>array[small]->distance)
        small = right;
    if(small != index)
    {
        struct MinHeapNode *smallestNode = heap->array[small];
        struct MinHeapNode *indexNode = heap->array[index];
        heap->pos[smallestNode->vertex] = index;
        heap->pos[indexNode->vertex] = small;
        swapNode(&heap->array[small], &heap->array[index]);
        MinHeapify(heap,small);
    }
}
int checkEmpty(struct MinHeap* heap)
{
    return heap->size == 0;
}
struct MinHeapNode* extractMin(struct MinHeap* heap)
```

```
{
	if(checkEmpty(heap))
		return NULL;
	struct MinHeapNode* root = heap->array[0];
	struct MinHeapNode* last = heap->array[heap->size - 1];
	heap->array[0] = last;
	heap->pos[root->vertex] = heap->size -1 ;
	heap->pos[last->vertex] = 0;
	--heap->size;
	MinHeapify(heap, 0);
	return root;
}
void DecreaseKey(struct MinHeap* heap ,int vertex, int dist)
{
	int i = heap->pos[vertex];
	heap->array[i]->distance = dist;
	while(i && heap->array[i]->distance < heap->array[(i-1)/2]->distance)
	{
		heap->pos[heap->array[i]->vertex] = (i-1)/2;
		heap->pos[heap->array[(i-1)/2]->vertex] = i;
		swapNode(&heap->array[i], &heap->array[(i-1)/2]);
		i = (i-1)/2;
	}
}
int isInMinHeap(struct MinHeap* heap,int vertex)
{
	if(heap->pos[vertex] < heap->size)
		return 1;
	return 0;
}
void Printresult(int dist[], int n)
{
	printf("Vertex Distance From Source \n");
	for(int i=0; i<n; i++)
		printf("%d \t\t %d \n",i,dist[i]);
}
int * Djikstra(struct Graph* graph, int source)
{
	int v =graph->no_of_nodes;
	static int dist[9];
	struct MinHeap* heap = createMinHeap(v);
```

```
    for(int k=0; k<v;k++)
    {
        dist[k] = 9999;
        heap->array[k] = newMinHeapNode(k,dist[k]);
        heap->pos[k] = k;
    }
    heap->array[source] = newMinHeapNode(source, dist[source]);
    heap->pos[source] = source;
    dist[source] = 0;
    DecreaseKey(heap, source, dist[source]);
    heap->size = v;

    while(!checkEmpty(heap))
    {
        struct MinHeapNode* minNode = extractMin(heap);
        int u = minNode->vertex;
        struct AdjListNode* traverse = graph->AdjList[u];
        while(traverse != NULL)
        {
            int v = traverse->destination;
            if(isInMinHeap(heap,v) && dist[u]!=9999 && (traverse-
>distance + dist[u] < dist[v]))
            {
                dist[v] = dist[u] + traverse->distance;
                DecreaseKey(heap,v,dist[v]);
            }
            traverse = traverse->next;
        }
    }
    //Printresult(dist,v);
    return dist;
}
```

CONTENTS OF queue.h:

```
#include<stdio.h>
#include<stdlib.h>
typedef struct user
{
   char name[40],email[40],uname[40],pwd[40],notify[200];
   int cap,book,demand,v,location;
```

```
  float bill;
  long ph;
}user;
typedef struct Queue
{
    int front,rear,size;
    int capacity;
    struct user* user_array;
}Queue;
struct Queue* createQueue(int capacity)
{
    struct Queue* q = (struct Queue*)malloc(sizeof(struct Queue));
    q->front = 0;
    q->rear = capacity -1;
    q->size = 0;
    q->capacity = capacity;
    q->user_array = (struct user*)malloc(sizeof(struct user)* capacity);
    return q;
}
int isFull(struct Queue* q)
{
    if(q->size == q->capacity)
    {
        printf("Please try again later! Maximum orders reached \n");
        return 1;
    }
    return 0;
}
int isEmpty(struct Queue* q)
{
    return (q->size == 0);
}
void Enqueue(struct Queue* q, struct user* u)
{
    if(isFull(q))
        return;
    q->rear = (q->rear + 1)%q->capacity;
    q->user_array[q->rear] = *u;
    q->size = q->size + 1;
    //printf("Your request is being processed ! Please Wait...\n");
}
```

```
struct user* dequeue(struct Queue* q)
{
        if(isEmpty(q))
                printf("Queue Empty ! \n");
        else
        {
                struct user* u = &q->user_array[q->front];
                q->front = (q->front + 1)%q->capacity;
                q->size = q->size -1;
                return u;
        }
        return NULL;
}
struct user* front(struct Queue* q)
{
        if(isEmpty(q))
                printf("All requests have been satsified ! ");
        else
                return &q->user_array[q->front];
        return NULL;
}
```

CONTENTS OF mini.c:

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include"queue.h"
#include"djkstra_list_heap.h"

Queue *q;
int *dist;
char loc[]
[30]={"Adyar","kilpauk","Chetpet","Nungambakkam","Kodambakkam","Mylapor
e","Royapettah","Mogappair","Parrys"};
void bill(user obj)
{
   if(strcmp(obj.notify,"\nYour booking has been confirmed... ")==0)
   {
      /*transport: dist*10; rs.10 per km
```

```
            water: 5 rs per litr
            additional cost ..5% of total*/
        printf("\n-----Bill-----\n");
        printf("\nDistance from %s to %s :
%d",loc[0],loc[obj.location],dist[obj.location]);
        printf("\nDemand : %d",obj.demand);
        printf("\nTransportation charges : %d",dist[obj.location]*10);
        printf("\nWater charges: %d",obj.demand*5);
        float tot=dist[obj.location]*10+obj.demand*5;
        printf("\nAdditional Charges: %.2f",0.05*tot);
        tot+=0.05*tot;
        printf("\nTotal Cost : %.2f",tot);
    }
    else
    {
        printf("\nNo current bookings!!!!");
    }
}

void put_details(user obj)
{
    printf("\nName: %s \t\tPhone No.: %lu\nEmail Id: %s\tLocation: %s\tStorage :
%d",obj.name,obj.ph,obj.email,loc[obj.location],obj.cap);
}
void admin(struct Graph* graph)
{
    printf("\n\n-----Admin-----");
    FILE *fp;
    user obj,*u;
    char uname[40],str[100];
    int ch,op;
    do
    {
        printf("\nMENU");
        printf("\n1. Process Registrations\n2. Add Edges\n3. Confirm booking\n4.
Exit\nEnter your choice : ");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:
                fp=fopen("record.dat","r+b");
    fseek(fp,sizeof(user),0);
    while(!feof(fp))
    {
        fread(&obj,sizeof(user),1,fp);
        if(obj.v==0)
        {
            put_details(obj);
            printf("\n1-Accept\t0-Reject : ");
            scanf("%d",&op);
            if(op==1)
                obj.v=1;
            else
                obj.v=2;
            fseek(fp,-sizeof(user),SEEK_CUR);
            fwrite(&obj,sizeof(user),1,fp);
        }

    }
    fclose(fp);
    break;
case 2:
for(int i=0;i<9;++i){
    printf("\n\t%d - %s",i,loc[i]);
}
int s,e,dis;
printf("\nSelect starting vertex and ending vertex: ");
scanf("%d",&s);
scanf("%d",&e);
printf("\nEnter distance between them: ");
scanf("%d",&dis);
addEdge(graph,s,e,dis);
break;

case 3:
    while(!isEmpty(q))
    {
        u=dequeue(q);
        printf("\nno: %s,book: %d",u->uname,u->book);

        put_details(*u);
        printf("\nDemand: %d",u->demand);
```

```
          printf("\n1-Confirm\t0-Dismiss : ");
          scanf("%d",&op);
          if(op==1)
          {
            u->book=0;
            strcpy(u->notify,"\nYour booking has been confirmed... ");
            printf("%s\n",u->notify );

                      dist =Djikstra(graph, 0);
             /*
            call dijstra here
            get distance and calculate bill and assign it to bill variable..
            */
            int d=dist[u->location];
            if(d==9999)
            {
            strcpy(u->notify,"\nServices not provided to the given area... ");
            }
            //printf("\nDistance: %d",d);
          }
          else
          {
            u->book=0;
            strcpy(u->notify,"\nYour booking has been cancelled due to
unavailibility of resources...\nsorry for the inconvinienice. ");
            //printf("%s\n",u->notify );
          }
          {
            fp=fopen("record.dat","r+b");
            if(!feof(fp))
              fread(&obj,sizeof(user),1,fp);
            while(!feof(fp))
            {
              if(strcmp(obj.uname,u->uname)==0)
              {
                fseek(fp,-sizeof(user),SEEK_CUR);
                fwrite(u,sizeof(user),1,fp);
              }
              fread(&obj,sizeof(user),1,fp);
            }
            fclose(fp);
```

```
                    }
                }
                break;
            case 4:
                return;
            default:
                printf("\nInvalid choice...");
                break;
        }
    }while(ch!=4);
}

user user_menu(user obj)
{
    int ch;
    do
    {
    printf("\n\n------User Menu------");
    printf("\n\n1. Book water\n2. View Booking Status\n3. View Bill\n4.
Exit\nEnter your choice: ");
    scanf("%d",&ch);
    switch(ch){
        case 1:
            printf("\n-----Booking-----");
            printf("\nEnter Demand(in ltrs.): ");
            scanf("%d",&obj.demand);
            while(obj.demand>obj.cap)
            {
                printf("\nDemand cannot be more than your storage capacity...");
                printf("\nRe-enter Demand(in ltrs.): ");
                scanf("%d",&obj.demand);
            }
            obj.book=1;
            Enqueue(q,&obj);
            strcpy(obj.notify,"Your booking is yet to be confirmed...");
            printf("\n%s ",obj.notify);
            break;
        case 2:
            printf("\n\n-----Booking Status-----");
            printf("\n %s",obj.notify);
            break;
        case 3:
           bill(obj);
            break;
        case 4:
            return obj;
        default:
            printf("\nInvalid choice....");
    }
    }while(ch!=4);
    return obj;
}
void get_user(char uname[])
{
    char re[40];
    FILE *fp;
    user obj;
    strcpy(obj.uname,uname);
    fp=fopen("record.dat","ab");
    printf("\nEnter your Name: ");
    scanf(" %[^\n]",obj.name);
    do
    {
        printf("\nEnter new password: ");
        scanf("%s",obj.pwd);
        printf("\nRe-enter password: ");
        scanf("%s",re);
        if(strcmp(obj.pwd,re))
            printf("\nNot matching!!!");
    }while(strcmp(obj.pwd,re));
    printf("\nEnter Email Id:");
    scanf("%s",obj.email);
    for(int i=0;i<9;++i){
        printf("\n\t%d - %s",i,loc[i]);
    }
    printf("\nSelect location(0-8): ");
    scanf("%d",&obj.location);
    printf("\nEnter phone Number : ");
    scanf("%li",&obj.ph);
    printf("\nEnter water storage capacity: ");
    scanf("%d",&obj.cap);
    obj.book=0;
    obj.demand=0;
    obj.v=0;
    strcpy(obj.notify,"No bookings done...");
    fwrite(&obj,sizeof(user),1,fp);
    printf("\nRegister Successfully!!!!");
    printf("\nWait for admin verification...");
    fclose(fp);
}
void login(struct Graph* graph)
{
    printf("\n------Login------\n");
    char uname[40],pwd[40];
    FILE *fp;
    user obj;
    fp=fopen("record.dat","rb");
    printf("\nEnter username: ");
    scanf("%s",uname);
    int f=0;
    fread(&obj,sizeof(user),1,fp);
    while(!feof(fp))
    {
        if(strcmp(uname,obj.uname)==0)
        {
            f=1;
            if(obj.v==1)
            {
                printf("\nEnter password: ");
                scanf("%s",pwd);
                if(strcmp(pwd,obj.pwd)==0)
                {
                    printf("\nLogin Successful......");
                    fclose(fp);
                    if(strcmp(uname,"admin")==0)
                        admin(graph);
                    else
                        {
                        obj=user_menu(obj);
                        fseek(fp,-sizeof(user),SEEK_CUR);
                        fwrite(&obj,sizeof(user),1,fp);
                        fclose(fp);
                    }
                    return;
                }
                else
                {
                    printf("\nLogin failed....");
                    break;
                }
            }
            else if(obj.v==0)
            {
                printf("\nAccount not yet verified!!! Try some time later...");
                fclose(fp);
                return;
            }
            else if(obj.v==2)
            {
                printf("\nYour Registration has been rejected by the admin... try
registering again...");
                fseek(fp,0,SEEK_SET);
                FILE *ftemp=fopen("temp.dat","wb");
                fread(&obj,sizeof(user),1,fp);
                while(!feof(fp))
                {
                    if(strcmp(obj.uname,uname)!=0)
                        fwrite(&obj,sizeof(user),1,ftemp);
                    fread(&obj,sizeof(user),1,fp);
                }
                fclose(ftemp);
                fclose(fp);
                remove("record.dat");
                rename("temp.dat","record.dat");
                return;
            }
        }
        fread(&obj,sizeof(user),1,fp);
    }
    fclose(fp);
    if(f==0)
        printf("\nEnter valid user name...");
}
```

```
int regis()
{
   printf("\n----REGISTER----\n");
   char uname[40],pwd[40],re[40];
   user obj;
   FILE *fp;
   fp=fopen("record.dat","rb");
   printf("\nEnter user name(0-quit): ");
   scanf("%s",uname);
   while(!feof(fp))
   {
      fread(&obj,sizeof(user),1,fp);
      if(strcmp(uname,"0")==0)
      {
         fclose(fp);
         return 0;
      }
      if(strcmp(uname,obj.uname)==0)
      {
         printf("\nUsername already taken!!!\nRe-Enter!!!");
         fclose(fp);
         return 1;
      }
   }
   fclose(fp);
   get_user(uname);
   return 0;
}
int main()
{
   int ch;
   q=createQueue(9);
   struct Graph* graph = createGraph(9);
   {
      remove("record.dat");
      remove("complaints.txt");
      FILE *fp=fopen("record.dat","wb");
      user obj;
      strcpy(obj.uname,"admin");
      strcpy(obj.pwd,"admin");
      obj.v=1;
```

```
        fwrite(&obj,sizeof(user),1,fp);
        fclose(fp);
    }
    do
    {
        printf("\n\n------WATER MANAGEMENT SYSTEM------");
        printf("\n\n1.Register\n2.Login\n3.Exit\nEnter your choice : ");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:
                if(regis())
                    regis();
                break;
            case 2:
                login(graph);
                break;
            case 3:
                    break;
            default:
                if(ch!=3)
                    printf("\nInvalid choice....");
                break;
        }
    }while(ch!=3);
    return 0;
}
```

OUTPUT:

```
/*initially the option is chosen from the menu 1.Register
option is for new users to register and details such as
email id,password,their location,their phone number and
their water storage capacity is entered. After
registering their has to be verified by the Admin. If the
registration is canceled they will have to re-register.
2.Login option is for the verified users to login to
their profile.
*/
```

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

----REGISTER----

Enter user name(0-quit): ram

Enter your Name: ramarajan

Enter new password: 123

Re-enter password: 123

Enter Email Id:coloursattai@gmail.com

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select location(0-8): 3

Enter phone Number : 9877626713

Enter water storage capacity: 1000

Register Successfully!!!!
Wait for admin verification...

------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

----REGISTER----

Enter user name(0-quit): rana
Enter your Name: ranadagupati

Enter new password: 123

Re-enter password: 123

Enter Email Id:ranaonehitwonder@gmail.com

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select location(0-8): 7

Enter phone Number : 987623131

Enter water storage capacity: 500

Register Successfully!!!!
Wait for admin verification...

------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

----REGISTER----

Enter user name(0-quit): siva

Enter your Name: sivasamy

Enter new password: 123

Re-enter password: 123

Enter Email Id:asuranbb100days@gmail.com

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select location(0-8): 5

Enter phone Number : 9878763424

Enter water storage capacity: 5000

Register Successfully!!!!
Wait for admin verification...
```

/*option 2 is for logging in into the respective account using username and password. Admin is default to process registrations, confirm water booking by other users and add new locations to deliver water.

Username and password is given as admin in the next step for admin to login*/

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: admin

Enter password: admin

Login Successful......
```

/* in the admin menu, process registrations can access the details of the users registered and accept or reject their registration. The registrations are displayed in queue giving first preference to the user registered first*/

```
-----Admin-----
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 1

Name: ramarajan         Phone No.: 9877626713
Email Id: ramrajancoloursattai@gmail.com  Location:
Nungambakkam  Storage :1000
1-Accept  0-Reject : 1

Name: ranadagupati          Phone No.: 987623131
Email Id: ranaonehitwonder@gmail.com  Location: Mogappair
     Storage :500
1-Accept  0-Reject : 1

Name: sivasamy          Phone No.: 9878763424
Email Id: asuranbb100days@gmail.com   Location:
Royapettah    Storage :5000
1-Accept  0-Reject : 0

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: ram

Enter password: 123

Login Successful......

------User Menu------
```

```
1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

------Login------

Enter username: rana
Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: siva

Your Registration has been rejected by the admin... try
registering again...
```

/*another registration is done that will be accepted by the admin*/

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

----REGISTER----

Enter user name(0-quit): sivasam

Enter your Name: sivasamy

Enter new password: 123

Re-enter password: 123

Enter Email Id:asuranbb100days@gmail.com

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select location(0-8): 3

Enter phone Number : 2873648723

Enter water storage capacity: 1000

Register Successfully!!!!
Wait for admin verification...

------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------
```

```
Enter username: admin

Enter password: admin

Login Successful......

-----Admin-----
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 1

Name: sivasamy          Phone No.: 2873648723
Email Id: asuranbb100days@gmail.com   Location:
Nungambakkam  Storage :1000
1-Accept  0-Reject : 1
```

/*option 2 is add edges is to add routes to the present locations to which the water can be delivered, the input is starting vertex and the end vertex from the locations menu and the distance between them is taken from the input*/

```
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 0
1

Enter distance between them: 4
```

```
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 0
7

Enter distance between them: 8

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 1
2

Enter distance between them: 8

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
```

```
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 1
7

Enter distance between them: 11

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 2
3

Enter distance between them: 7

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 2
8

Enter distance between them: 2

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 2
5

Enter distance between them: 3 4

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 3
4

Enter distance between them: 9

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 3
5

Enter distance between them: 14

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 4
5

Enter distance between them: 10
```

```
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 5
6

Enter distance between them: 2

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 6
7

Enter distance between them: 1

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
```

```
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 6
8

Enter distance between them: 6

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 2

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select starting vertex and ending vertex: 7
8

Enter distance between them: 7

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 3

/*since there aren't any water bookings currently,it
returns to menu*/
```

```
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 4
```

/*by logging in as the user, the menu is displayed to book water, to view booking status and to view the bill. The water demand is specified in the booking. Validation is done to check that the demand is less than the storage capacity.*/

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: ram

Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 1

-----Booking-----
Enter Demand(in ltrs.): 10000

Demand cannot be more than your storage capacity...
Re-enter Demand(in ltrs.): 100

Your booking is yet to be confirmed...
```

/*View Booking Status option gives the current status updated by the admin */

```
------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 2


-----Booking Status-----
 Your booking is yet to be confirmed...

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: rana

Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 1

-----Booking-----
Enter Demand(in ltrs.): 90
```

```
Your booking is yet to be confirmed...

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4
```

/* the water booking can be confirmed only by the admin. The admin is logged in and from the menu confirm bookings option is chosen. The water bookings are displayed by priority queue to be confirmed by the admin.*/

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: siva

Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 1

-----Booking-----
Enter Demand(in ltrs.): 200

Your booking is yet to be confirmed...

------User Menu------

1. Book water
2. View Booking Status
```

```
3. View Bill
4. Exit
Enter your choice: 4



------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: admin

Enter password: admin

Login Successful......

-----Admin-----

/* Admin can confirm or reject bookings and notify the
user. If the booking is canceled the user has to re-
book.*/

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 3

no: ram,book: 1
Name: ramarajan        Phone No.: 9877626713
Email Id: ramrajancoloursattai@gmail.com  Location:
Nungambakkam  Storage :1000
Demand: 100
1-Confirm 0-Dismiss : 1

Your booking has been confirmed...

no: rana,book: 1
Name: ranadagupati          Phone No.: 987623131
Email Id: ranaonehitwonder@gmail.com  Location: Mogappair
     Storage :500
```

```
Demand: 90
1-Confirm 0-Dismiss : 1

Your booking has been confirmed...

no: siva,book: 1
Name: sivasamy          Phone No.: 2873648723
Email Id: asuranbb100days@gmail.com   Location:
Nungambakkam  Storage :1000
Demand: 200
1-Confirm 0-Dismiss : 0

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: ram

Enter password: 123

Login Successful......
```

/*the booking status and the bill is being viewed by all the users, in case if the booking has been canceled by the admin, a message will be displayed. If no bookings is done appropriate message will be printed for when option 3.View Bill option is selcted.*/

```
------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
```

```
Enter your choice: 2


-----Booking Status-----

Your booking has been confirmed...

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 3

-----Bill-----

Distance from Adyar to Nungambakkam : 19
Demand : 100
Transportation charges : 190
Water charges: 500
Additional Charges: 34.50
Total Cost : 724.50

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: rana

Enter password: 123

Login Successful......
```

```
------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 2


-----Booking Status-----

Your booking has been confirmed...

------User Menu------

/* Bill is calculated based on the distance, demand and
additional charges*/

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 3

-----Bill-----

Distance from Adyar to Mogappair : 8
Demand : 90
Transportation charges : 80
Water charges: 450
Additional Charges: 26.50
Total Cost : 556.50

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
```

```
3.Exit
Enter your choice : 2

------Login------

Enter username: siva

Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 2


-----Booking Status-----

Your booking has been cancelled due to unavailibility of
resources...
sorry for the inconvinience.

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 3

No current bookings!!!!

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------
```

```
1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: prasa

Enter valid user name...

------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 1

----REGISTER----

Enter user name(0-quit): sam

Enter your Name: 123

Enter new password: 123

Re-enter password: 123

Enter Email Id:sam@yahoo

     0 - Adyar
     1 - kilpauk
     2 - Chetpet
     3 - Nungambakkam
     4 - Kodambakkam
     5 - Mylapore
     6 - Royapettah
     7 - Mogappair
     8 - Parrys
Select location(0-8): 10

Enter phone Number : 38124243

Enter water storage capacity: 4000

Register Successfully!!!!
Wait for admin verification...
```

```
------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: admin

Enter password: admin

Login Successful......

-----Admin-----
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 1

Name: 123          Phone No.: 38124243
Email Id: sam@yahoo     Location:     Storage :4000
1-Accept  0-Reject : 1

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: sam
Enter password: 123

Login Successful......

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 1

-----Booking-----
Enter Demand(in ltrs.): 60

Your booking is yet to be confirmed...

------User Menu------

1. Book water
2. View Booking Status
3. View Bill
4. Exit
Enter your choice: 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: admin

Enter password: admin

Login Successful......

-----Admin-----
MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
```

```
Enter your choice : 3

no: sam,book: 1
Name: 123          Phone No.: 38124243
Email Id: sam@yahoo     Location:     Storage :4000
Demand: 60
1-Confirm 0-Dismiss : 0

Your booking has been confirmed...

MENU
1. Process Registrations
2. Add Edges
3. Confirm booking
4. Exit
Enter your choice : 4


------WATER MANAGEMENT SYSTEM------

1.Register
2.Login
3.Exit
Enter your choice : 2

------Login------

Enter username: sam

Enter password: 123

Your Registration has been rejected by the admin... try
registering again...
```