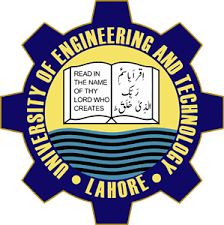
**DSA Project Report**

****

**Project Title CLI Based Facebook App**

**Grp Member Muhamad Huzzifa & Hafiz Abdul Rehman**

**Reg No 2020-CS-606 & 2020-CS-624**

**Grp No Grp-01**

**UNIVERSITY OF ENGINEERING AND TECNOLOGY LAHORE NEW CAMPUS**

Contents

[Classes 3](#_Toc90923470)

[Personal Tab 4](#_Toc90923471)

[Data Structure: 4](#_Toc90923472)

[Functionality: 4](#_Toc90923473)

[Description: 4](#_Toc90923474)

[Diagram: 4](#_Toc90923475)

[Home Tab 5](#_Toc90923476)

[Data Structure: 5](#_Toc90923477)

[Functionality: 5](#_Toc90923478)

[Description: 5](#_Toc90923479)

[Diagram: 5](#_Toc90923480)

[Friends Tab 7](#_Toc90923481)

[Data Structure: 7](#_Toc90923482)

[Functionality: 7](#_Toc90923483)

[Description: 7](#_Toc90923484)

[Diagram: 7](#_Toc90923485)

[Message Tab 8](#_Toc90923486)

[Data Structure: 8](#_Toc90923487)

[Functionality: 8](#_Toc90923488)

[Description: 8](#_Toc90923489)

[Diagram: 8](#_Toc90923490)

# **Classes**

1. Str Class (I / O for Sign Up and Log In)
2. Post Class (I / O for for Post Class)
3. PNode Class (Post Node (next and pre))
4. PostLinking Class (add, update and delete Post)
5. FNode Class (Friend Node (name, next and pre))
6. FriendLinking Class (add, un-follow and View friend)
7. Node Class (str, post and friend Object)
8. Sign-Up Class (Insert User, Log-In and Search function)
9. Comment Class (I / O for Comment)
10. CNode Class (Comment Object, next and pre)
11. CommentLinking Class (Add and View Comment)
12. Message Class (I / O for Message)
13. MNode Class (Message Object, next and pre)
14. MessageLinking Class (Send, View and Delete Message)

# **Personal Tab**

## **Data Structure:**

Doubly linked list

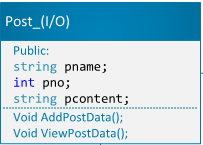
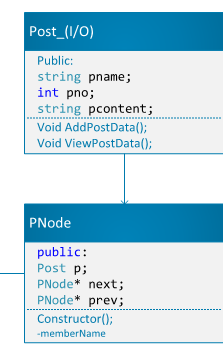
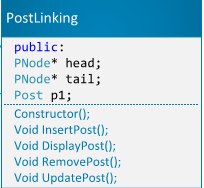
## **Functionality:**

1. ADD POST
2. VIEW POST
3. REMOVE POST
4. UPDATE POST

## **Description:**

In this Tab we are using Doubly linked list (Data Structure). In add Post function the Log-In User added some posts and in the View Tab user can view his post. In Remove post Function user can delete and update his post. This class is linking with Node and post (I / O) class.

## **Diagram:**

****

# **Home Tab**

## **Data Structure:**

Doubly linked list

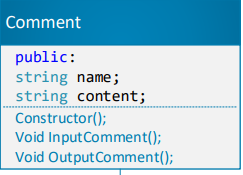
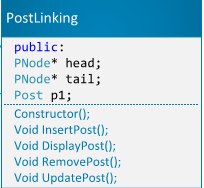
## **Functionality:**

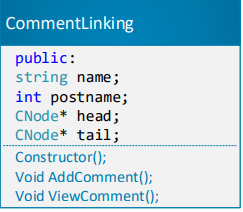
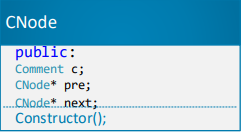
1. ADD POST
2. VIEW POST
3. REMOVE POST
4. UPDATE POST
5. ADD COMMENT
6. VIEW COMMENT

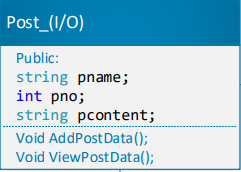
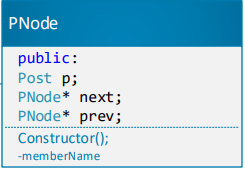
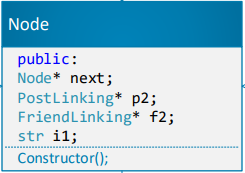
## **Description:**

In this Tab we are using Doubly linked list (Data Structure). In add Post function the Log-In User added some posts and in the View Tab user can view his post and his friend post. In Remove post Function user can delete and update his post. In this function we add some comment on his post and view comments on his post. This class is linking with Node and post (I / O) class.

## **Diagram:**







# **Friends Tab**

## **Data Structure:**

Doubly linked list

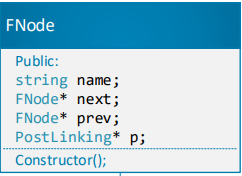
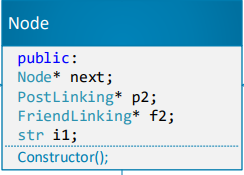
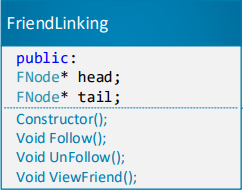
## **Functionality:**

1. ADD FRIEND
2. VIEW FRIENDLIST

## **Description:**

In this Tab we are using Doubly linked list (Data Structure). In add Friend function the Log-In User added some friend that are already sign-Up and in the View Friend List user can view his friend in the List.

## **Diagram:**



# **Message Tab**

## **Data Structure:**

Doubly linked list

## **Functionality:**

1. SELECT FRIEND AND SEND MESSAGE
2. VIEW MESSAGE
3. DELETE MESSAGE

## **Description:**

In this Tab we are using Doubly linked list (Data Structure). In send Message function the Log-In User send message to added friends by selecting one of them. User can delete message and view Message through their Proper Function

## **Diagram:**

