SENT

+ void view\_sent()

+ void delete\_sent()

+ void forward()

MAILS

* Draft: int
* Draft\_content: string
* \*sent: SENT
* #\*password:Password\_safe

# From: string

# To: string

# Message\_body: string

+ void compose()

MOBILE

* Model\_no: int
* Model\_name: string
* \*mail: Mails
* \*contact: Contact List
* \*cal: Calendar
* \*password: Password\_safe

+ getdetails()

+ void add\_calendar()

+ void add\_mails()

+ void add\_contact\_list()

INBOX

+ void view()

+ void delete()

+ void forward()

CONTACT LIST

* Name: string
* Address: string
* Phone\_number: long int
* Password: Password\_safe

+ void add\_contact()

+ void remove\_contact()

+ void view\_contact()

CALENDAR

* Date: int
* Month: int
* Year: int
* Day: string
* \*password: Password\_safe

+ void viewdate()

+ void viewholidays()

+ void addevent()

PASSWORD SAFE

* Password: string
* \*cpass: Calendar
* \*clpass: Contact\_list
* \*mpass: Mails

+ void change\_password()

+ void view\_passwod()

+ string password\_generator(string)

*UML DIAGRAM FOR OUR PROJECT ‘BANANA PHONE’* ***PROJECT COMPONENTS AND FEATURES:***

The above UML Diagram gives a brief description of our project. The classes with their corresponding member variables and relationship are discussed below:

1. MOBILE: It is our MAIN CLASS. In this Basically we’ve Member Variables namely : Model\_no, Model\_name, and the pointers of other classes based on the relationship we’ve established between them. Its Public Functions are getDetails() which will show the Details of The Mobile Phone, and other functions used to add contact lists, Calendar Events etc. Note that we have established a ASSOCIATION Relationship between Mobile Phone class and Calendar, Mails, Contact List Classes etc. and Aggregation between Password Safe and Mobile Phone class Based on our Assumption that the things like Mails, Contact List and Calendar Events are synchronized to the online servers and we can access them using Laptops and other devices while on the other hand Password safe is not synchronized to the internet, therefore we’ve taken Aggregation relationship. We’ve not taken Composition Relationship as it is not necessary that whenever a mobile phone is made, a password safe is made automatically. It depends upon the user whether to ‘install’ password safe or not.

2) MAILS : This Class is used to store the Mails of the User. It is also a SuperClass/BaseClass to its SubClasses Inbox and Sent. It’s protected member variables(so that it can be inherited by it’s subclasses) are To, From and the message body..It’s Functions are compose() used to compose mails, and we’ve also taken a pointer of it’s subclass Sent as whenever a mail is composed and sent, it should automatically reflect back on SENT Class too.

2a.) SENT : It is the Subclass of MAILS class and inherits attributes from MAILS Super Class. It’s public functions are view\_sent() used to view sent mails, delete\_sent() used to delete saved sent mails and forward() used to forward sent mail to other recipients.

2b.) INBOX : It is also a Subclass of MAILS class. It’s public functions are same as that of SENT Class and stands the same real world meaning ,i.e. to forward, delete and view mails in our inbox.

3.) Calendar: This Class has private member variables as date, day, month, year etc. used to store date as the name suggests. It has Public Functions as viewHolidays() used to view holidays in that month/year etc., addEvent() used to add explicitly some event on a particular date, and viewDate() which simply displays the date.

4.) Contact List : This Class has private members as Name, Address and Phone No. use to store information about the contact to be added and public functions as viewContact() used to view information about the already stored contact, addContact() used to add Contact in the Contact List and removeContact() used to remove contact from the Contact List.

5.) Password Safe: This is a Class used as an installed app by the user to provide password to the other apps(classes) so that no one can access it without the password. It also acts as a password used to store password for different websites and apps within or outside the phone. To store the password we’ve implemented dual association relationship to other app (classes), so that whenever we create a password for any other apps, it will automatically be stored in the password safe. For this purpose we’ve created a pointer of password safe type into that class and a pointer of that class (app) type in password safe class (dual association). The Member variables of this class is password (string type) which will store the password for each class (app). It’s public Functions are changePassword() used to change password, viewPassword() used to display contents of the password safe and a special function PasswordGenerator() which will generate a non-hackable password for the User.

This gives a brief discussion about our project, in which we tried to implement a small part of a primitive Mobile Phone (kind of a prototype), by inserting some essential components of a mobile phone.

BANANA PHONE.

THANK YOU :D