#### SystemAdmin.h void clearData(string) systemAdmin() systemAdmin(string, string, s systemAdmin(string, string, st void createOfficial(string, string, void delOfficial(Official&) void setActiveOfficial(Official& void setInactiveOfficial(Official void createSystemAdmin(strii void delSystemAdmin(string) string returnUserID(string) void resetPassword(string, str

#### Member.h

string memberAddress

string memberPhone Member() Member(string, string, string, string, Member(string, string, string, string void setAddress(string) string getAddress() void setPhone(string) string getPhone() void displayData() void getAccountHistory(vector-void saveData() void constructMember(string)

#### Timer.h

int year int month int day

Time(int, int, int) Time(string) void setYear(int) int getYear() void setMonth(int) int getMonth() void setDay(int) int getDay() string toString(int) int toInt(string) string timeNow()

#### AVL.h

Node<T> \*root

int itemCount int Height(Node<T>\*&)

int Balance(Node<T>\*) int leafs(Node<T>\*)

void midpoints(vector<Node<T>\*>&, int, int)

void delSubRoot(Node<T>\*&)
Node<T>\*insertAssist(Node<T>\*&, Node<T>\*&)

Node<T>\*insertItemAssist(Node<T>\*&, Node<T>\*&) Node<T> \*delAssist(Node<T>\*&, T) Node<T> \*delItemAssist(Node<T>\*&, T, T)

Node<T> \*searchAssist(T, Node<T>\*) void displaySystemAdmins(Node<T>\*)

void displayOfficials(Node<T>\*) void displayMembers(Node<T>\*)

void displayItems(Node<T>\*)

void rotate(Node<T>\*&) Node<T> \*rotateRight(Node<T>\*&)

Node<T> \*rotateLeft(Node<T>\*&) Node<T> \*rightMost(Node<T>\*&) void saveAVL(Node<T>\*, ofstream&)

AVLtree() ~AVLtree()

int getHeight() bool getBalance() int getLeafCount int getCount()

void insert(T) void insertItem(T, T)

void insertList(T, vector<T>) void del(T) void delltem(T, T)

vector<T> returnItems(T)

void search(T)

void display(int) void saveData(string)

void constructTree(string)

## Node.h

T value vector<T> list Node\* left Node \*right int height

Node() Node(T val) Node(T val, T item) Node(T val, vector<T> items)

#### string officialStatus

Official()

Official.h

Official(string, string, string, string, Official(string, string, string, string, void setOfficialStatus(string) string getStatus() void addNewMember(Member&) void searchmemberAccountID(stri void searchMemberID(string) void searchMemberFirstName(stri void searchMemberLastName(stri

void searchMemberPhone(string) void searchMemberAddress(string

#### void addNewAccount(Member&, A void saveData() void constructOfficial(string)

Controller.h

Users.h

User(string, string, string, string, string, string, string, string, string, string)

string name

string password

string history

User()

string type string loginReport

void setName(string)

string getID() void setPass(string)

void setType(string) string getType()

string getPass()

string getLogin() void setHistory(string) string getHistory()

virtual void storeUserInfo()

virtual void constructUser(string)

string getName() void setID(string)

string ID

static memberData allFiles static vector<AccountVersion> list

Controller()

static void initialAccounts()

static void createType(string, double, double, double, double) static void alterType(string, double, double, double, double)

static void displayTypes() static void saveTypes()

static void constructTypes(string)

static string validLogin(string, string) static bool availableID(string)

static void recordNewOfficial(string) static void recordNewSystemAdmin(string)

static vector<string> getMemberLogin(string) static void recordNewMember(Member&, Account&) static void updateMemberData(string, string)

static void displayAllAccount(string) static void displayAccountData(string) static void changeFirstName(string, string, string)

static void changeLastName(string, string, string) static void changeAddress(string, string, string)

static void changePhone(string, string, string) static void newAccount(string, string)

#### Encryption.h

static string locker static vector<int> scrambler

Encryption() static size\_t position

static long long FNV1a(string) static string encryption(string)

static string decryption(string)

static string hasher(string) static string hexer(long long)

# static int index(string, int)

#### Dictionary.h

AllFiles

AVLTree<string>FirstNames AVLTree<string>LastNames

AVLTree<string>Phones AVLTree<string>Addresses

AVLTree<string>Users AVLTree<string>Accounts

dictionary: LinkedList<dictEntry>\* int maximum int usedLocals int totalKeys double factor float threshold int collisionCount

void updateFactor() void resize() sccountDict() ~accountDict() void insert(dictEntry void del(string) void update(string, string) void display() bool isEmpty() string search(string) bool inDict(string) void saveEntry(string) void constructDict(string) void refreshDict()

### DictEntry

string key

dictEntry() dictEntry(string) dictEntry(string, string) bool operator == (const dictEntry& bool operator != (const dictEntry&)

### Queue.h

LinkedList<string> list

Queue(int) void enqueue(string) void dequeue(string&) bool isEmpty() bool isFull() void peek(string&) void clearQueue() void displayQueue() void saveQueue() void constructQueue()

### LinkedList.h

node<T> \*head node<T> \*tail int maximum int currentCount

LinkedList(int) LinkedList(const LinkedList&) ~LinkedList() void push(T) void append(T) void del(T) void delHead() void delTail() bool search(T&) void peekHead(T&) void peekTail(T&) void display() bool full() bool empty() int getCount() void clearAl() void operator = (const LinkedList&)

# AccountVersion

Account.h

#### node T value

node<T> \*next node<T> \*prev