

MainUser
- <u>mainUser</u> : User
+ <u>setMainUser</u> (User): void + <u>getMainUser</u> (): User

HashingUtil
+ generateUserHashedPassword(String): string

+ validateUserInput(email: Stri + validateUserEmail(email: Str

ConnectHub
+ Main(String): void

FriendshipManagement
+ sendFriendRequest(User,User): boolean + acceptFriendRequest(User,FriendRequest): boolean + declineFriendRequest(User,FriendRequest): boolean + blockUser(User,User): void + removeFriend(User,User): void + suggestFriends(User): Set<User>

- userInstance: UserSignup;
- UserSignupSingleton() + getUser(): User + setUser(user: User): void + getInstance(): UserSignup + userSignup(email: String,

FriendRequest
- requestSender: User - requestReceiver: User - requestStatus: Status
<div><<enumeration>></div> <div>Status</div> <div>+ Accepted + Declined + Pending</div>
+ FriendRequest(User, User, Status): + getRequestSender(): User + getRequestReciever(): User + getRequestStatus(): Status + setRequestSender(User): void + setrequestReciever(User): void + setRequestStatus(Status): void

User
- userId: String - email: String - username: String - password: String - dateOfBirth: String - status: String - friends: Set<User> - blockedUsers: Set<User> - sentFriendRequests: Set<FriendRequest> - receivedFriendRequests: Set<FriendRequest>
+ getUserId(): String + getUserEmail(): String + getUsername(): String + getUserDateOfBirth(): String + getUserStatus(): String + getUserPassword(): String + getUserFriends(): Set<User> + getUserBlockedUsers(): Set<User> + getUserSentFriendRequests(): Set<FriendRequest> + getUserReceivedFriendRequests(): Set<FriendRequest> + setUserPassword(unHashedPassword: String): void + setUserDateOfBirth(date: LocalDate): void + setUsername(username: String): void + setUserStatus(status: String): void + setUserEmail(email: String): boolean + userLogout(): void + toJSON(): JSONObject + sendFriendRequest(receiver: User): void + acceptFriendRequest(request: FriendRequest): boolean + declineFriendRequest(request: FriendRequest): boolean + blockUser(user: User): void + removeFriend(user: User): void + isUserBlocked(user: User): boolean

- user_ - users - unique
- UserL + getUi + getIn + getU: + getU: + addU + getU: + userL + saveL + loadL + getU:

ValidationUtil
ing, username: String, password: String, dateOfBirth: LocalDate): boolean ing): boolean

UserSignupSingleton
Singleton - user: User
Singleton username: String, password: String, dateOfBirth: LocalDate): boolean

UserDatabase
_database: UserDatabase : ArrayList<User> ieCounter: int
Database() niqueCounter(): int istance(): UserDatabase serIds(): ArrayList<String> sers(): ArrayList<User> Jser(user: User): boolean ser(userId: String): User Login(userId: String, password: String): boolean UsersToFile(filePath: String): void UsersFromFile(filePath: String): void serFromId(userId: String): User

AContent
- contentId: int - author: User - text: String - imagePath: String - timeOfUpload: LocalDateTime - timeStampFormat: DateTimeForm
+ getTimestamp(): String + getText(): String + getImagePath(): String + getContentId(): int + getAuthor(): User + getTimeOfUpload(): LocalDateTim + setText(text: String): void + setImagePath(imagePath: String): + setTimeOfUpload(timeOfUpload: l + setTimeOfUpload(): void + getTimeStampFormat(): DateTime + AContent(author: User) + toJSON(): JSONObject + abstract uplope(): void

Post
+ Post(authorId: User) + uplope(): void + fromJSON(jsonObject: JSONObject): Post

atter
ne
c: void
LocalDateTime): void
eFormatter

ContentDataBase
- posts: ArrayList<Post> - stories: ArrayList<Story> - dataBase: ContentDataBase - id: int
+ getInstance(): ContentDataBase + addContent(cont: Post): void + addContent(cont: Story): void - removeStory(): void # getPosts(): ArrayList<Post> # getStories(): ArrayList<Story> + getFriendsPosts(user: User): ArrayList<Post> + getFriendsStories(user: User): ArrayList<Story> + getUniqueld(): int - load(): void + save(): void + close(): void

story
+ Story(authorId: User) + uplade(): void + fromJSON(jsonObject: JSONObject): Story

CreateContentP
- selectedFile: File
+ CreateContentP() + initComponents(): void + jLabel2MouseClicked(evt: java.awt.event.MouseEvent): void + uploadPostEvent(evt: java.awt.event.ActionEvent): void + uploadStoryEvent(evt: java.awt.event.ActionEvent): void <<Swing Components>> - jLabel2: JLabel - jScrollPane1: JScrollPane - jTextArea1: JTextArea - uploadPostButton: JButton - uploadStoryButton: JButton

ImageResizer
- ImageResizer() + resizeImage(imagePath: String, width: ir

LoginFrame
+ LoginFrame() - initComponents(): void + main(args: String[]): void - passwordActionPerformed(evt: java.awt.
<<Swing Components>> - jLabel1: JLabel - jLabel2: JLabel - login: JButton - password: JTextField - userid: JTextField

FriendsGui
- currentUser: User - friendRequestsModel: DefaultListModel<String> - friendListModel: DefaultListModel<String> - friendSuggestionsModel: DefaultListModel<String> - changeFriendStatusModel: DefaultListModel<String>
+ FriendsGui() + initComponents(): void - acceptActionPerformed(evt: java.awt.event.ActionEvent): void - declineActionPerformed(evt: java.awt.event.ActionEvent): void - addFriendActionPerformed(evt: java.awt.event.ActionEvent): void - blockActionPerformed(evt: java.awt.event.ActionEvent): void - removeActionPerformed(evt: java.awt.event.ActionEvent): void <<Swing Components>> - accept: JButton - decline: JButton - addFriend: JButton - block: JButton - remove: JButton - friendRequests: JList<String> - friendList: JList<String> - friendSuggestions: JList<String> - changeFriendStatus: JList<String> - jTabbedPane2: JTabbedPane - jPanel1: JPanel - jPanel2: JPanel - jPanel3: JPanel - jPanel4: JPanel - jScrollPane1: JScrollPane - jScrollPane2: JScrollPane - jScrollPane3: JScrollPane - jScrollPane4: JScrollPane - jScrollPane5: JScrollPane

SignupFrame
+ SignupFrame() - initComponents(): void - signupActionPerformed(evt: java.awt.even + main(args: String[]): void <<Swing Components>> - email: JTextField - jCalendar1: JCalendar - jLabel1: JLabel - jLabel2: JLabel - jLabel3: JLabel - jLabel4: JLabel - password: JPasswordField - signup: JButton - username: JTextField

WelcomeFrame
+ WelcomeFrame() - initComponents(): void - loginActionPerformed(evt: java.awt.ever + main(args: String[]): void <<Swing Components>> - jLabel1: JLabel - login: JButton - signup: JButton

nt, height: int): ImageIcon

.event.ActionEvent): void

it.ActionEvent): void

nt.ActionEvent): void

news
<div> <div>- panel1: JPanel</div> <div>- panel2: JPanel</div> <div>- panel3: JPanel</div> <div>- panel4: JPanel</div> <div>- panel5: JPane</div> </div>
<div> <div>+ news() throws IOException</div> <div>+ postComp(post: Post): JPanel</div> <div>+ sroryComp(story: Story): JPanel</div> <div>+ friendComp(friend: User): JPanel</div> <div>+ refreshContent(): void</div> <div>+ createStyledButton(text: String): JButton</div> <div>+ main(args: String[]): void</div> </div> <div> <div><<Swing Components>></div> <div>- button1: JButton</div> <div>- refreshButton: JButton</div> <div>- button3: JButton</div> <div>- tabbedPane: JTabbedPane</div> <div>- buttonPanel: JPanel</div> </div>

