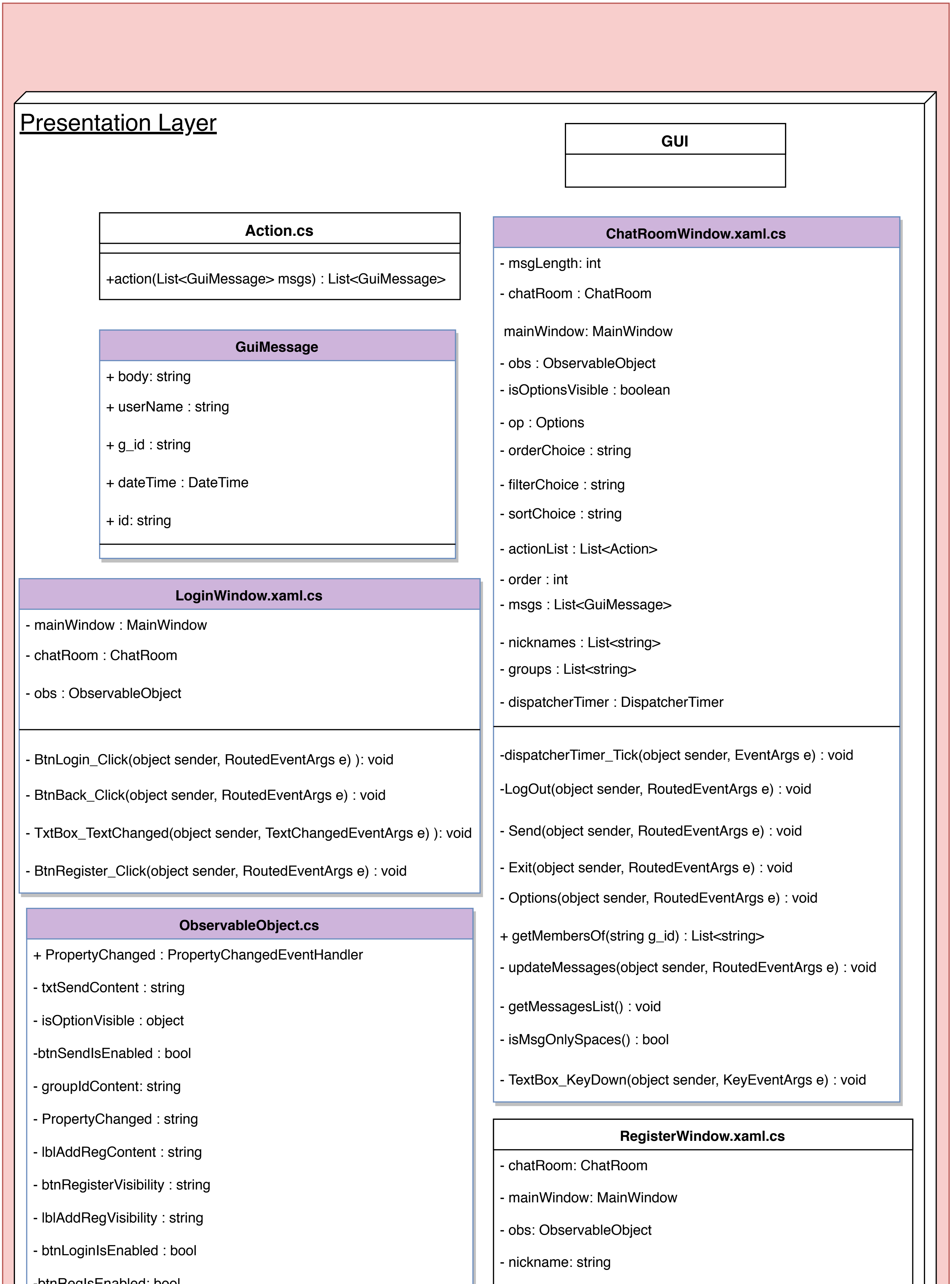


Presentation Layer

GUI

Action.cs

+action(List<GuiMessage> msgs) : List<GuiMessage>

GuiMessage

+ body: string

+ userName : string

+ g_id : string

+ dateTime : DateTime

+ id: string

LoginWindow.xaml.cs

- mainWindow : MainWindow
- chatRoom : ChatRoom
- obs : ObservableObject

- BtnLogin_Click(object sender, RoutedEventArgs e)): void
- BtnBack_Click(object sender, RoutedEventArgs e) : void
- TxtBox_TextChanged(object sender, TextChangedEventArgs e)): void
- BtnRegister_Click(object sender, RoutedEventArgs e) : void

ObservableObject.cs

+ PropertyChanged : PropertyChangedEventHandler

- txtSendContent : string
- isOptionVisible : object

-btnSendIsEnabled : bool

- groupIdContent: string
- PropertyChanged : string
- lblAddRegContent : string
- btnRegisterVisibility : string
- lblAddRegVisibility : string
- btnLoginIsEnabled : bool

-btnRegIsEnabled: bool

ChatRoomWindow.xaml.cs

- msgLength: int
- chatRoom : ChatRoom

mainWindow: MainWindow

- obs : ObservableObject
- isOptionsVisible : boolean
- op : Options
- orderChoice : string
- filterChoice : string
- sortChoice : string
- actionList : List<Action>
- order : int
- msgs : List<GuiMessage>
- nicknames : List<string>
- groups : List<string>
- dispatcherTimer : DispatcherTimer

-dispatcherTimer_Tick(object sender, EventArgs e) : void

-LogOut(object sender, RoutedEventArgs e) : void

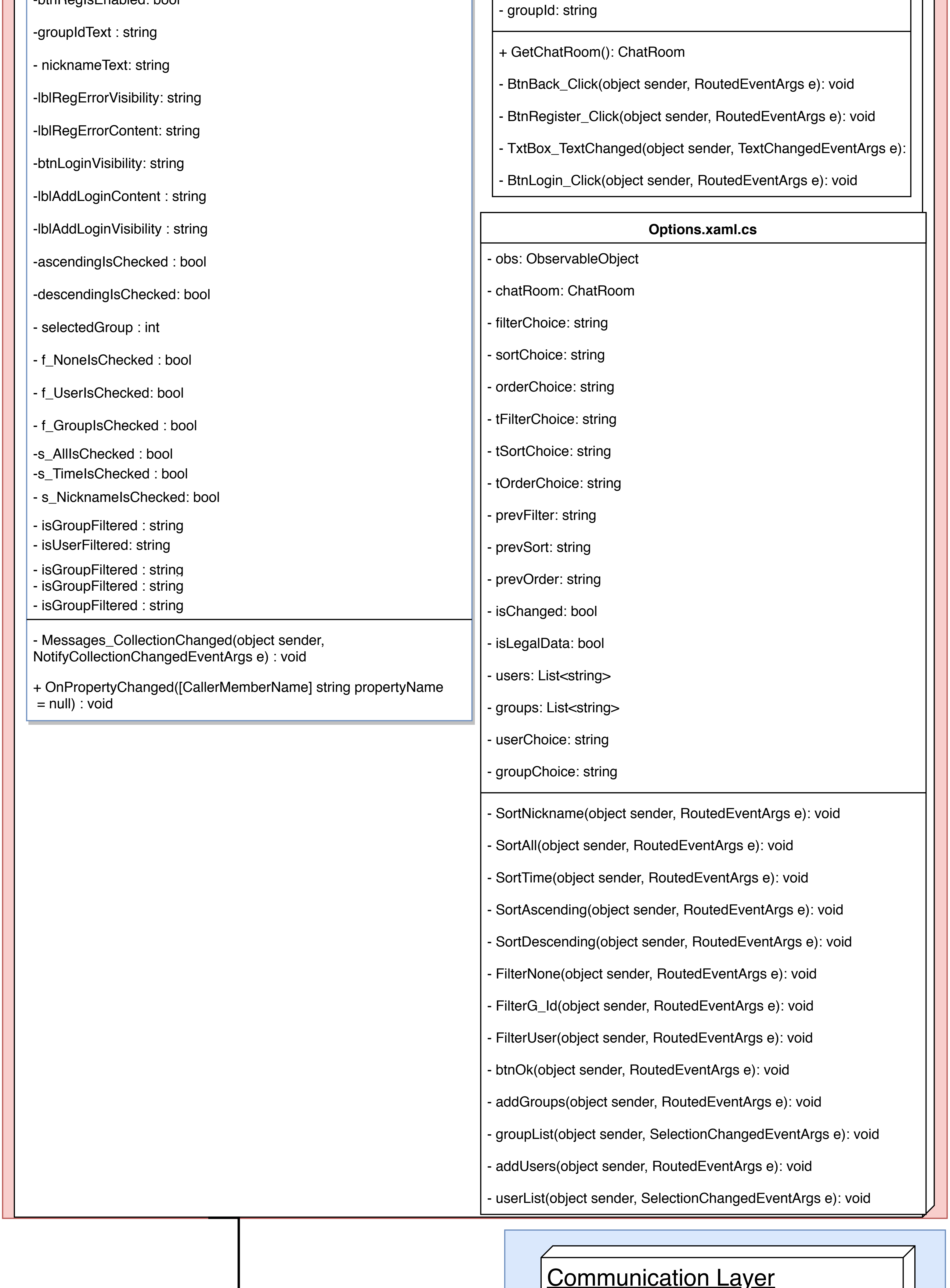
- Send(object sender, RoutedEventArgs e) : void
- Exit(object sender, RoutedEventArgs e) : void
- Options(object sender, RoutedEventArgs e) : void

+ getMembersOf(string g_id) : List<string>

- updateMessages(object sender, RoutedEventArgs e) : void
- getMessagesList() : void
- isMsgOnlySpaces() : bool
- TextBox_KeyDown(object sender, KeyEventArgs e) : void

RegisterWindow.xaml.cs

- chatRoom: ChatRoom
- mainWindow: MainWindow
- obs: ObservableObject
- nickname: string



-btnRegIsEnabled: bool

-groupIdText : string

- nicknameText: string

-lblRegErrorVisibility: string

-lblRegErrorContent: string

-btnLoginVisibility: string

-lblAddLoginContent : string

-lblAddLoginVisibility : string

-ascendingIsChecked : bool

-descendingIsChecked: bool

- selectedGroup : int

- f_NoneIsChecked : bool

- f_UserIsChecked: bool

- f_GroupIsChecked : bool

-s_AllIsChecked : bool

-s_TimeIsChecked : bool

- s_NicknameIsChecked: bool

- isGroupFiltered : string

- isUserFiltered: string

- isGroupFiltered : string

- isGroupFiltered : string

- isGroupFiltered : string

- Messages_CollectionChanged(object sender, NotifyCollectionChangedEventArgs e) : void

+ OnPropertyChanged([CallerMemberName] string propertyName = null) : void

- groupId: string

+ GetChatRoom(): ChatRoom

- BtnBack_Click(object sender, RoutedEventArgs e): void

- BtnRegister_Click(object sender, RoutedEventArgs e): void

- TxtBox_TextChanged(object sender, TextChangedEventArgs e):

- BtnLogin_Click(object sender, RoutedEventArgs e): void

Options.xaml.cs

- obs: ObservableObject

- chatRoom: ChatRoom

- filterChoice: string

- sortChoice: string

- orderChoice: string

- tFilterChoice: string

- tSortChoice: string

- tOrderChoice: string

- prevFilter: string

- prevSort: string

- prevOrder: string

- isChanged: bool

- isLegalData: bool

- users: List<string>

- groups: List<string>

- userChoice: string

- groupChoice: string

- SortNickname(object sender, RoutedEventArgs e): void

- SortAll(object sender, RoutedEventArgs e): void

- SortTime(object sender, RoutedEventArgs e): void

- SortAscending(object sender, RoutedEventArgs e): void

- SortDescending(object sender, RoutedEventArgs e): void

- FilterNone(object sender, RoutedEventArgs e): void

- FilterG_Id(object sender, RoutedEventArgs e): void

- FilterUser(object sender, RoutedEventArgs e): void

- btnOk(object sender, RoutedEventArgs e): void

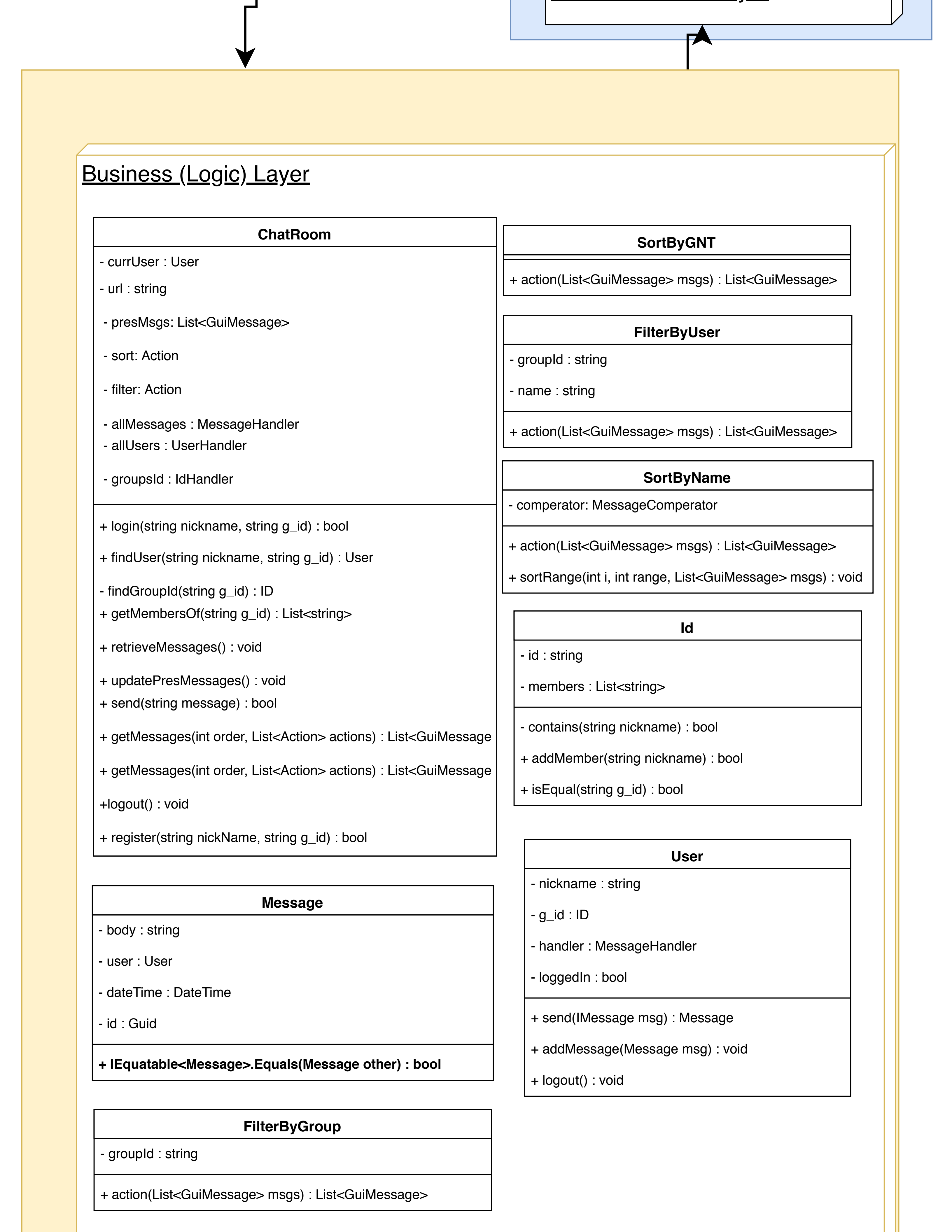
- addGroups(object sender, RoutedEventArgs e): void

- groupList(object sender, SelectionChangedEventArgs e): void

- addUsers(object sender, RoutedEventArgs e): void

- userList(object sender, SelectionChangedEventArgs e): void

Communication Layer



Business (Logic) Layer

ChatRoom

- currUser : User
- url : string
- presMsgs: List<GuiMessage>
- sort: Action
- filter: Action
- allMessages : MessageHandler
- allUsers : UserHandler
- groupsId : IdHandler

+ login(string nickname, string g_id) : bool
+ findUser(string nickname, string g_id) : User
- findGroupId(string g_id) : ID
+ getMembersOf(string g_id) : List<string>
+ retrieveMessages() : void
+ updatePresMessages() : void
+ send(string message) : bool
+ getMessages(int order, List<Action> actions) : List<GuiMessage
+ getMessages(int order, List<Action> actions) : List<GuiMessage
+logout() : void
+ register(string nickName, string g_id) : bool

SortByGNT

+ action(List<GuiMessage> msgs) : List<GuiMessage>

FilterByUser

- groupId : string
- name : string

+ action(List<GuiMessage> msgs) : List<GuiMessage>

SortByName

- comperator: MessageComperator

+ action(List<GuiMessage> msgs) : List<GuiMessage>
+ sortRange(int i, int range, List<GuiMessage> msgs) : void

Id

- id : string
- members : List<string>

- contains(string nickname) : bool
+ addMember(string nickname) : bool
+ isEqual(string g_id) : bool

Message

- body : string
- user : User
- dateTime : DateTime
- id : Guid

+ IEquatable<Message>.Equals(Message other) : bool

User

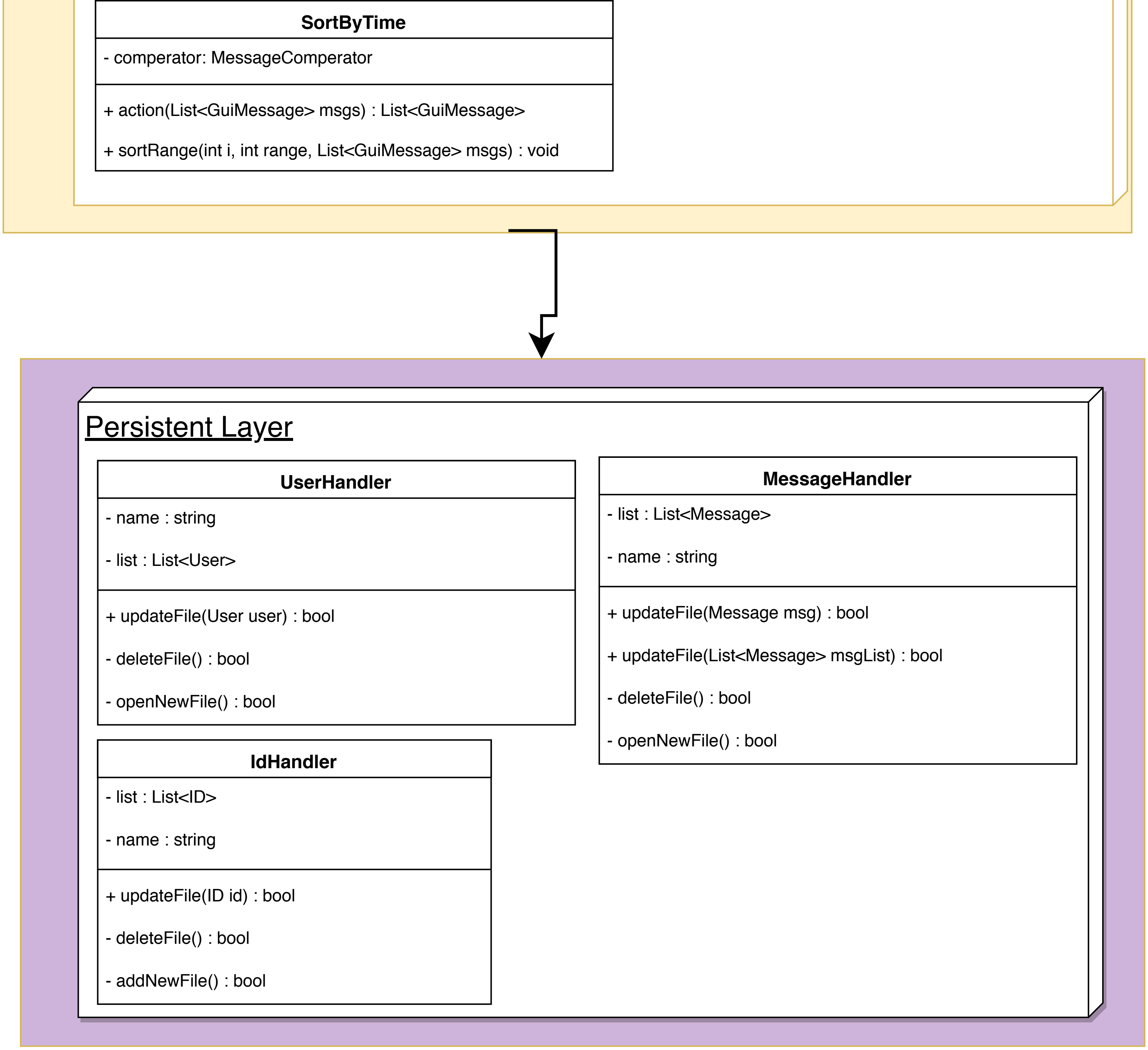
- nickname : string
- g_id : ID
- handler : MessageHandler
- loggedIn : bool

+ send(IMessage msg) : Message
+ addMessage(Message msg) : void
+ logout() : void

FilterByGroup

- groupId : string

+ action(List<GuiMessage> msgs) : List<GuiMessage>



SortByTime

- comperator: MessageComperator

+ action(List<GuiMessage> msgs) : List<GuiMessage>

+ sortRange(int i, int range, List<GuiMessage> msgs) : void

Persistent Layer

UserHandler

- name : string

- list : List<User>

+ updateFile(User user) : bool

- deleteFile() : bool

- openNewFile() : bool

MessageHandler

- list : List<Message>

- name : string

+ updateFile(Message msg) : bool

+ updateFile(List<Message> msgList) : bool

- deleteFile() : bool

- openNewFile() : bool

IdHandler

- list : List<ID>

- name : string

+ updateFile(ID id) : bool

- deleteFile() : bool

- addNewFile() : bool