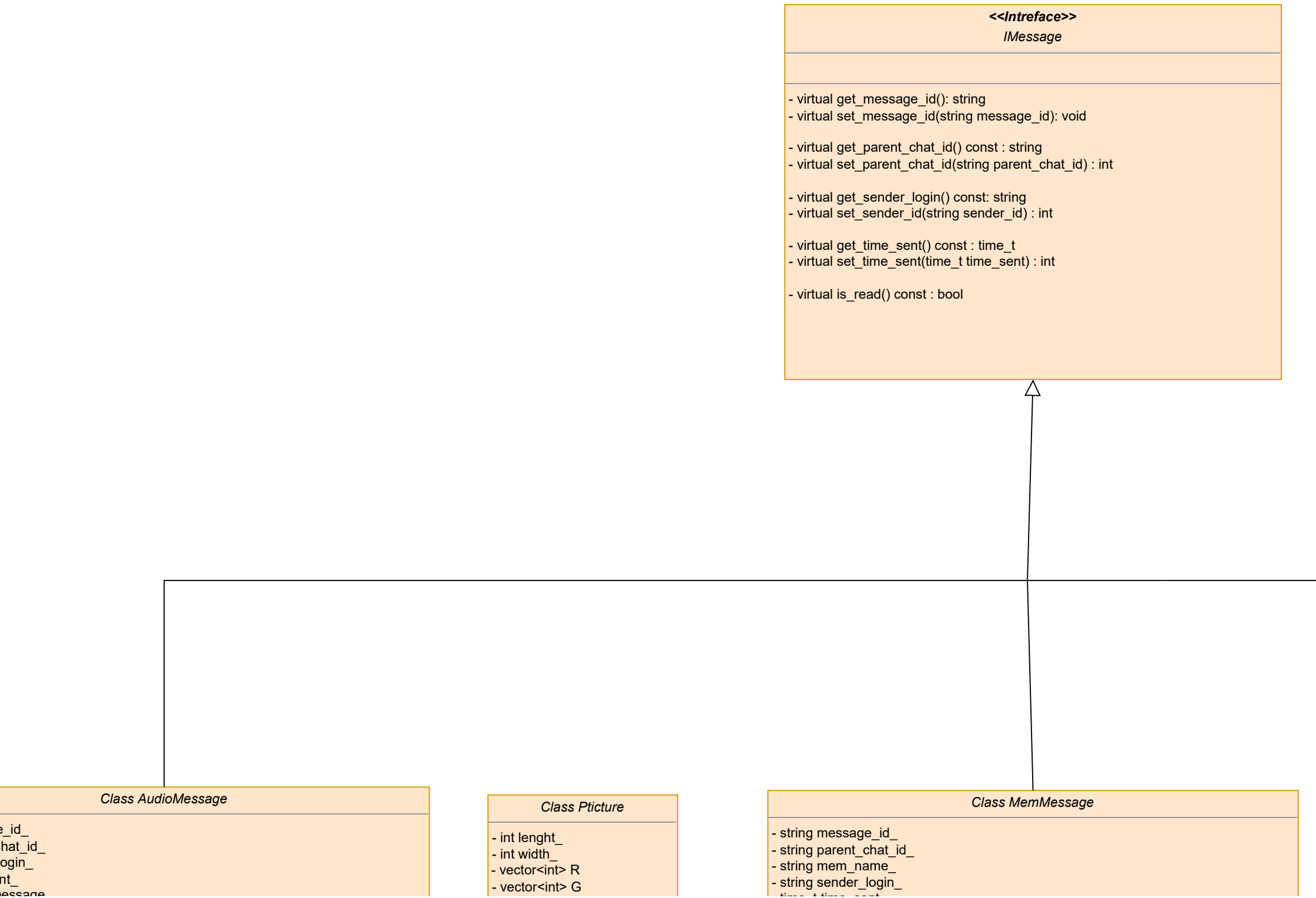




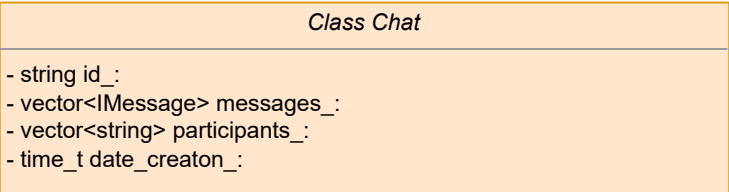


- string message  
- string parent\_c  
- string sender\_I  
- time\_t time\_se  
- Audio audio m



Class TextMessage
<div><div>- string message_id_ - string parent_chat_id_ - string sender_id_ - time_t time_sent - string text_message</div></div>

Class User
<div><div>- string login - string password - string token - vector&lt;string&gt; chat_list - int active_status</div></div>

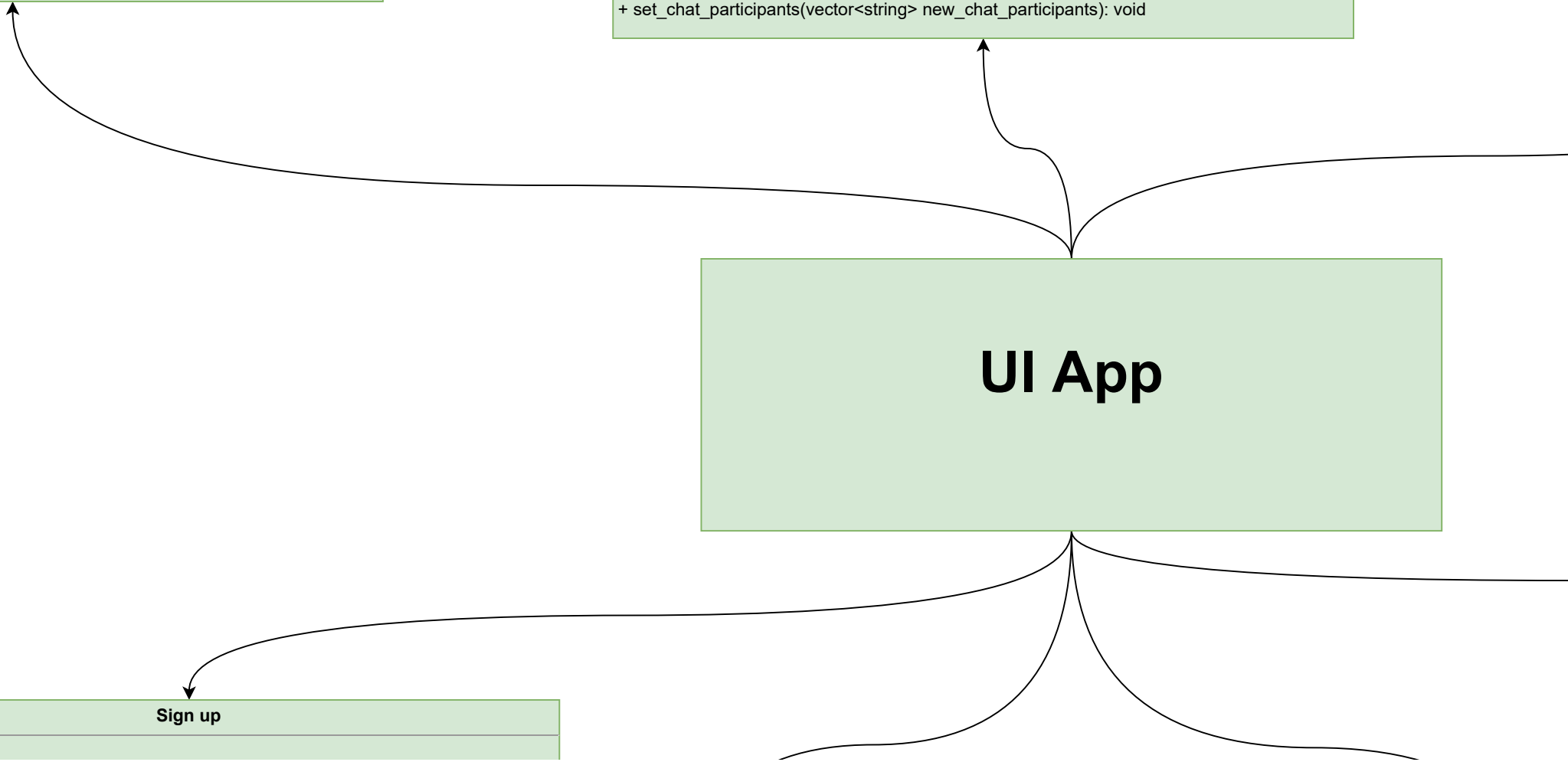
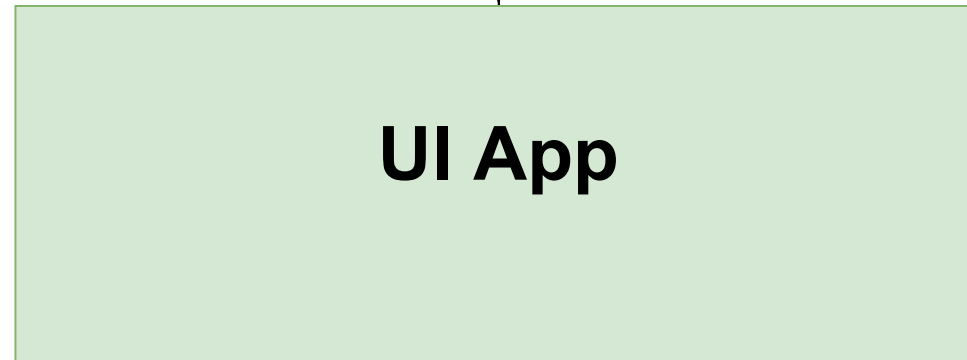
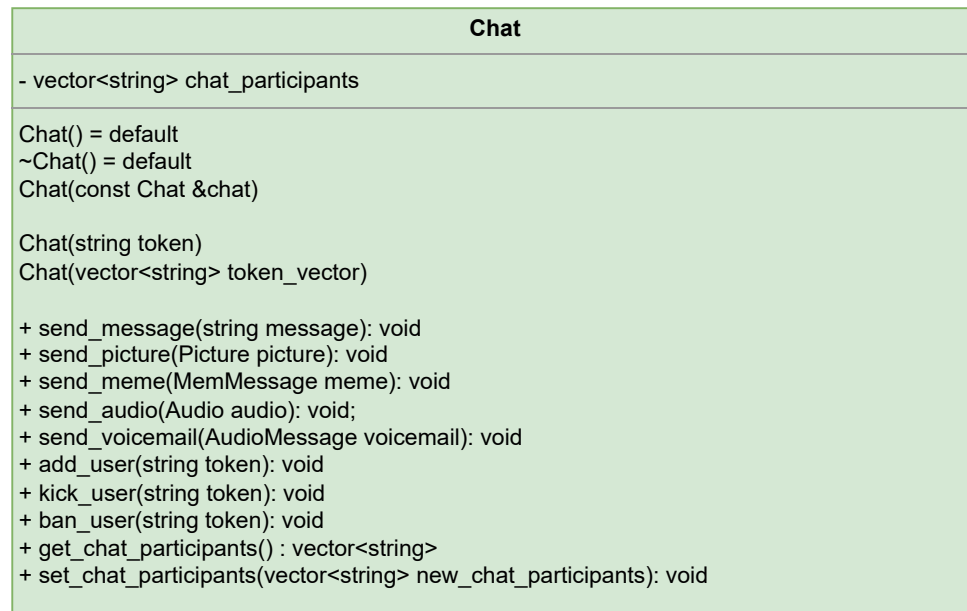
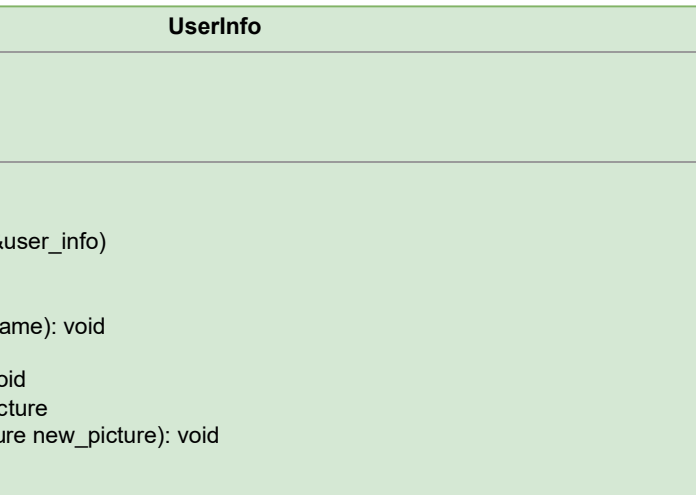


- string name  
- string id  
- Picture profile\_picture

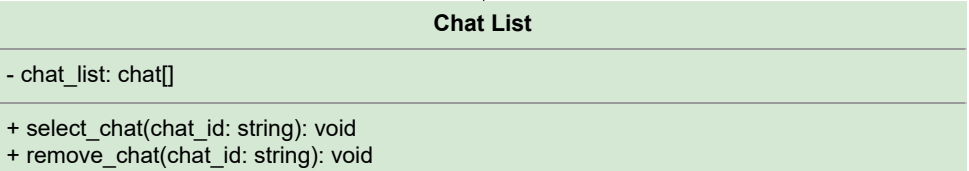
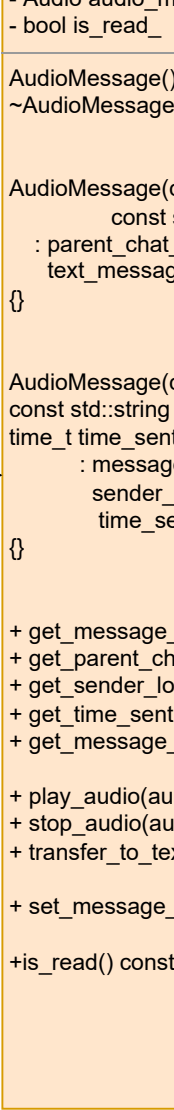
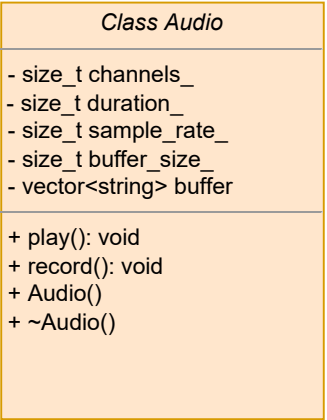
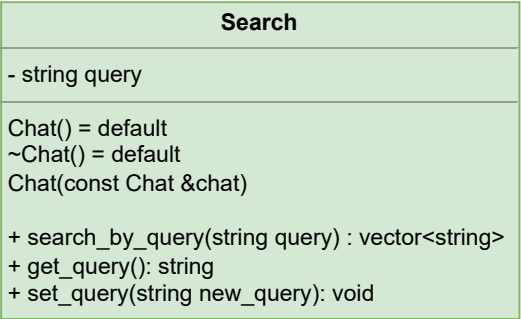
UserInfo() = default  
~UserInfo() = default  
UserInfo(const UserInfo &

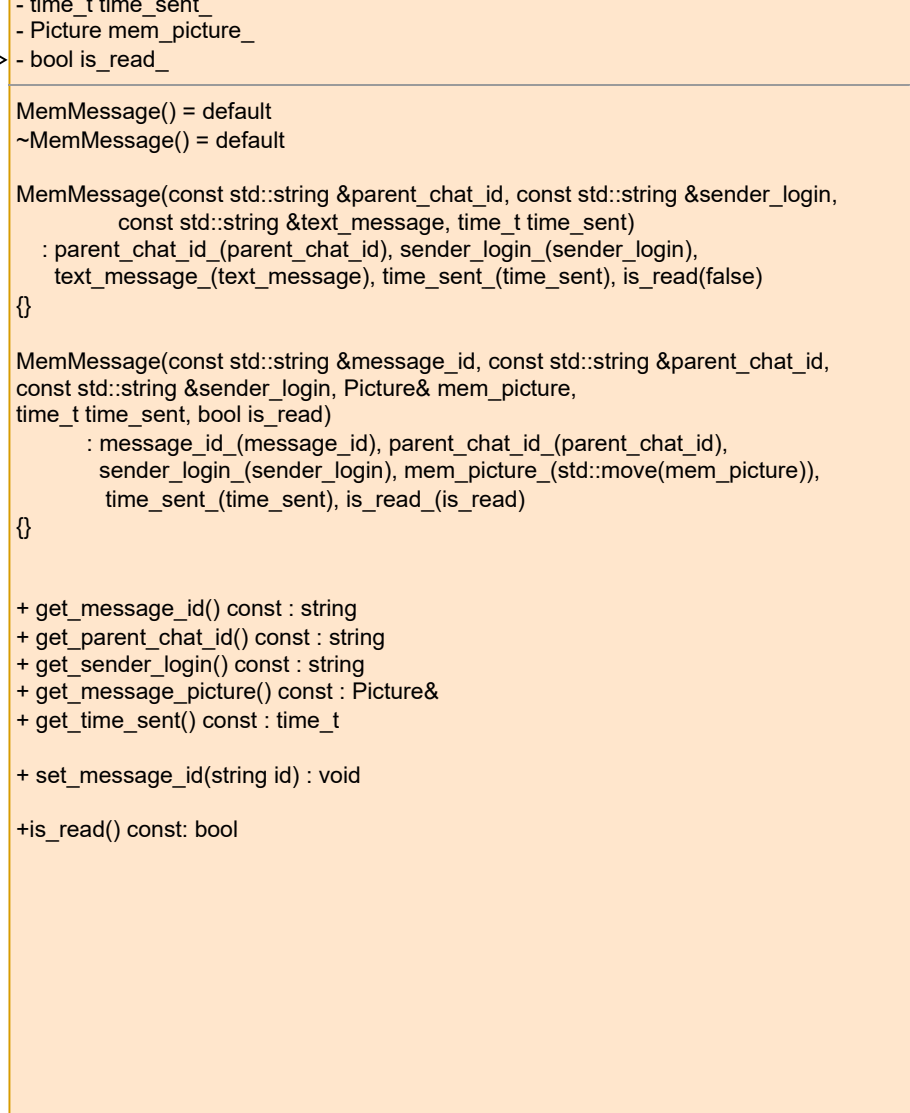
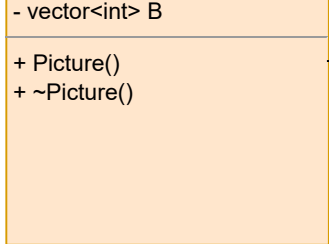
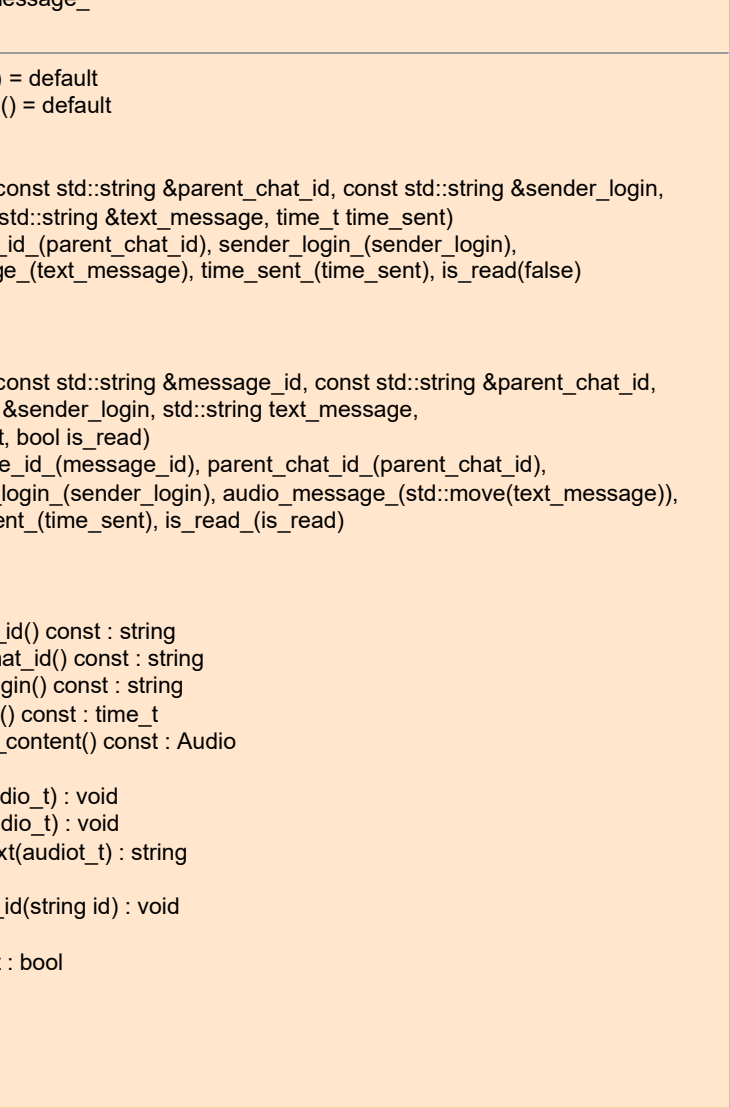
+ get\_name(): string  
+ set\_name(string new\_n  
+ get\_id(): string  
+ set\_id(string new\_id): v  
+ get\_profile\_picture(): Pic  
+ set\_profile\_picture(Pictu

- In









```

- string text_message_
- bool is_read_

TextMessage() = default
~TextMessage() = default

TextMessage (const std::string &parent_chat_id, const std::string &sender_login,
              const std::string &text_message, time_t time_sent)
    : parent_chat_id_(parent_chat_id), sender_login_(sender_login),
      text_message_(text_message), time_sent_(time_sent), is_read(false)
{}

TextMessage (const std::string &message_id, const std::string &parent_chat_id,
              const std::string &sender_login, std::string text_message,
              time_t time_sent, bool is_read)
    : message_id_(message_id), parent_chat_id_(parent_chat_id),
      sender_login_(sender_login), text_message_(std::move(text_message)),
        time_sent_(time_sent), is_read_(is_read)
{}

+ get_message_id() const : string
+ get_parent_chat_id() const : string
+ get_sender_login() const : string
+ get_message_content() const : string
+ get_time_sent() const : time_t

+ set_message_id(string id) : void

+is_read() const: bool

```

```

- int active_status

User() = default
~User() = default

User(const std::string &user_login)
    : login(user_login), status(0)
{}

User(const std::string &user_login, const std::string &user_password)
    : login(user_login), password(user_password), status(0)
{}

// constructor-move
User(std::string &user_login, std::string &user_password,
      std::string user_token, int user_status)
    : login(std::move(user_login)), password(std::move(user_password)),
      token(std::move(user_token)), status(user_status)
{}

+ get_login() const : string
+ get_password() const : string
+ get_token() const : string
+ get_active_status() const : int
+ get_chat_list() const : vector<string>

+ set_login(const string &login) : void
+ set_password(const string &login) : void
+ set_token(const string &token) : void
+ set_status(int active_status) : void

+ add_chat_to_list()

+ operator==(const User &user) const : bool
+ operator!=(const User &user) const : bool

```

Chat() = default  
~Chat() = default

Chat(vector<string> participants, time\_t date\_creation)  
: participants\_(std::move(participants)),  
date\_creation\_(date\_creation)  
{ }

Chat(string id,  
vector<IMessage> messages,  
vector<string> participants, time\_t date\_creation)  
:  
id\_(std::move(id)),  
messages\_(std::move(messages)),  
participants\_(std::move(participants)),  
date\_creation\_(date\_creation) {  
}  
}

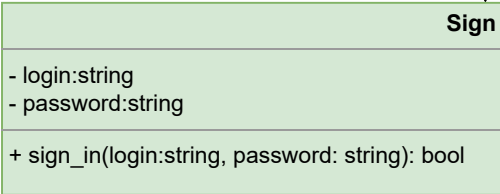
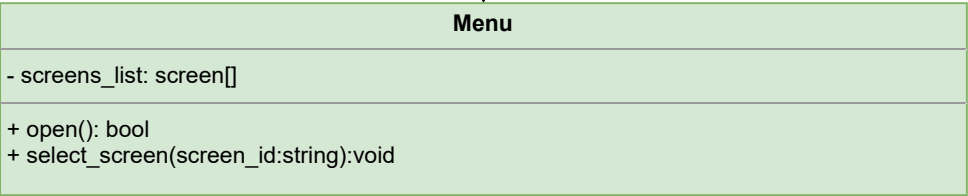
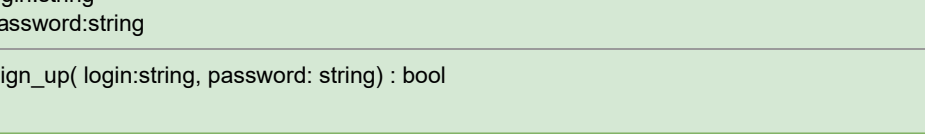
+ get\_last\_message() const : IMessage&  
+ get\_participants() const : vector<string>  
+ get\_messages() const : vector<IMessage>  
+ get\_chat\_date\_creation() const : time\_t  
+ get\_id() const : string

+ push\_new\_message(const IMessage& new\_message) : void  
+ add\_new\_participant(string& new\_participant\_id) : void

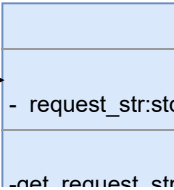
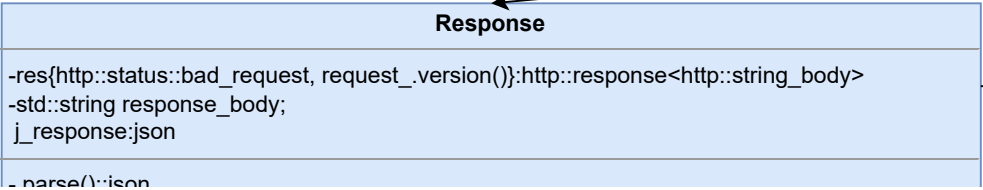
+ is\_empty() const : bool  
+ is\_dialogue() const : bool  
+ is\_monologue() const : bool  
+ is\_polilogue() const : bool

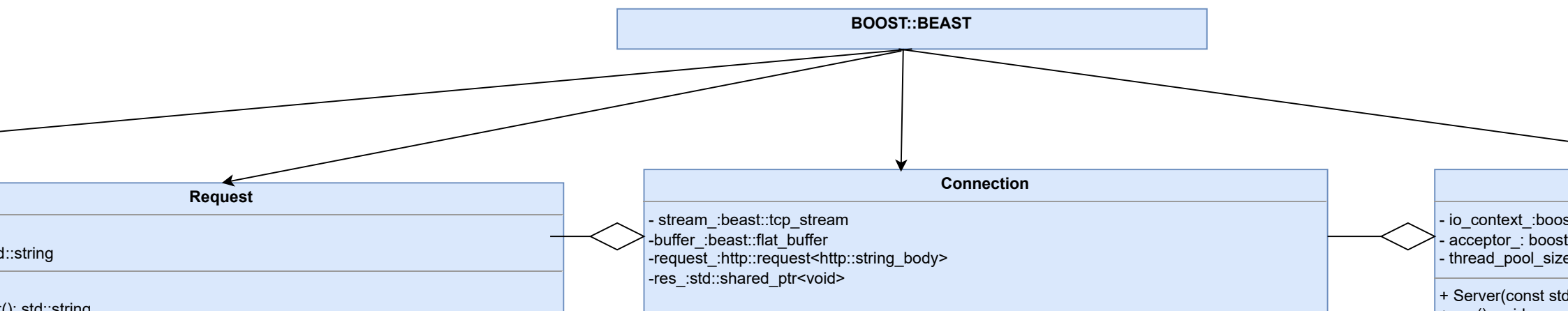
+ operator==(const Chat &chat) const : bool  
+ operator!=(const Chat &chat) const : bool

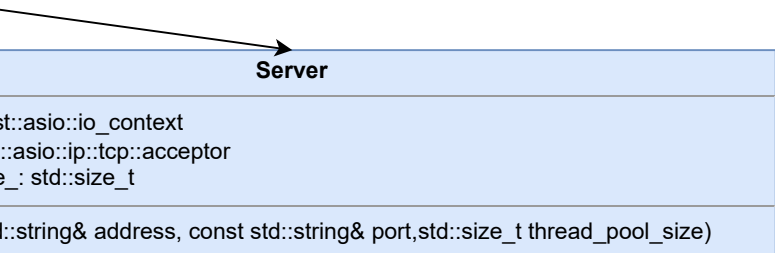




socket con

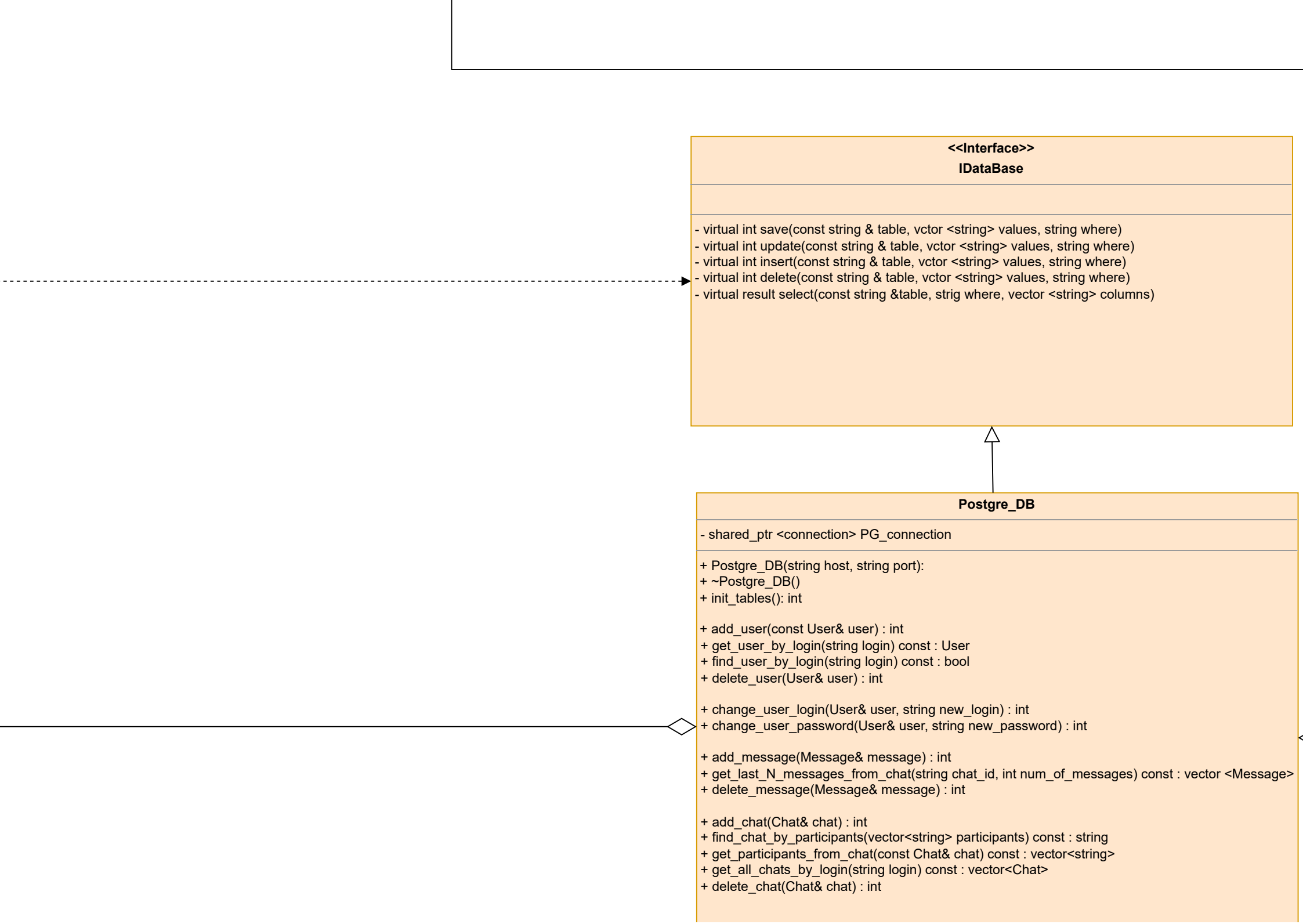






data flow from server









```
- parse():.json  
+get_user_id(std::string):int  
+change_user_login():int  
+change_user_password():int  
+get_all_chats_by_login(std::string):vector  
+add_chat():int  
+del_chat():int
```

```
-get_request_str  
+get_user_id(std  
+get_user_by_lo  
+get_all_chats_  
+add_chat():std  
+del_chat():std  
-what_type_requ
```

```
(): std::string  
d::string):int  
login(std::string):std::string  
py_login(std::string):vector  
:string  
string  
uest(std::string):bool
```

```
+ Connection(boost::asio::ip::tcp socket&& socket )  
+ start(): void  
- do_read():void  
-handle_read(boost::beast::error_code e,std::size_t bytes_transferred):void  
- handle_write(bool close, boost::beast::error_code e,std::size_t bytes_transferred):void  
-do_close():void
```

```
+ run(): void  
- start_accept():vo  
-handle_accept(b  
- handle_stop():vo
```

id  
post::beast::error\_code e, tcp::socket socket):void  
id

