

AskMeSystem
<ul style="list-style-type: none"> userManager questionManager
<ul style="list-style-type: none"> load_data_base(bool) reset_currentUser_questions() run()

HelperFunction
<ul style="list-style-type: none"> read_line_from_file(string): vector<string> write_line_on_file(string, vector<string>, int) const split_string(string, char): vector<string> to_int(string): int

Question
<ul style="list-style-type: none"> question_id parent_question_id from_user_id to_user_id is_anonymous_questions question_text answer_text
<ul style="list-style-type: none"> Question() Question(string) to_string() const: string print_to_question() const print_from_question() const print_feed_question() const get_question_id() const: int set_question_id(int&) get_parent_question_id() const: int set_parent_question_id(int&) get_from_user_id() const: int set_from_user_id(int&) get_to_user_id() const: int set_to_user_id(int&) get_is_anonymous_questions() const: int set_is_anonymous_questions(int&) get_question_text() const: string set_question_text(string&) get_answer_text() const: string set_answer_text(string&)

QuestionManager
<ul style="list-style-type: none"> question_id_to_question_ids_thread: map<int, vector<int>> question_id_to_QuestionObject: map<int, Question> last_id: int
<ul style="list-style-type: none"> QuestionManager() load_data_base() get_user_to_questions(const User&) const: vector<int> get_user_from_questions(const User&) const: vector<pair<int, int>> print_user_to_question(User&) print_user_from_question(User&) ask_question(User&, pair<int, int>&) answer_question(User&, QuestionManager&) delete_question(User&, QuestionManager&) list_feed() const update_data_base() get_question_id_to_question_ids_thread() const: map<int, vector<int>> set_question_id_to_question_ids_thread(map<int, vector<int>>&) get_question_id_to_QuestionObject() const: map<int, Question> set_question_id_to_QuestionObject(map<int, Question>&) get_last_id() const: int set_last_id(int&)

User
<ul style="list-style-type: none"> user_id user_name password name email allow_anonymous_questions question_id_from_me: vector<int> question_id_to_question_ids_thread: map<int, vector<int>>
<ul style="list-style-type: none"> User() User(string) to_string() const: string print() const reset_to_questions(const vector<int>&) reset_from_questions(const vector<pair<int, int>>&) get_user_id() const: int set_user_id(int&) get_user_name() const: string set_user_name(string&) get_password() const: string set_password(string&) get_name() const: string set_name(string&) get_email() const: string set_email(string&) get_allow_anonymous_questions() const: int set_allow_anonymous_questions(int&) get_question_id_from_me(): vector<int>& set_question_id_from_me(vector<int>&) get_question_id_to_question_ids_thread(): map<int, vector<int>>& set_question_id_to_question_ids_thread(map<int, vector<int>>&)

UserManager
<ul style="list-style-type: none"> user_name_to_UserObject: map<string, User> current_user: User last_id: int
<ul style="list-style-type: none"> UserManager() load_data_base() log_in() sign_up() reset_from_questions(const vector<pair<int, int>>&) reset_to_questions(const vector<int>&) list_user_names_ids() read_user_id(): pair<int, int>& update_data_base(User&) get_user_name_to_UserObject() const: map<string, User> set_user_name_to_UserObject(map<string, User>&) get_current_user(): User& set_current_user(User&) get_last_id() const: int set_last_id(int&)

View
<ul style="list-style-type: none"> show_read_menu(vector<string>): int read_int(int, int): int read_question_id(User&, QuestionManager&): int read_question_id_thread(User&): int access_system(UserManager&) menu() const: vector<string>