Team 15

Easy C++

Software Design Specification (SDS)

Version: 2

CM Identifier: SDS_easy

Revision History

Sl. No.	Prepared/ Modified by	E-mail	Version	Date	Approved by	Descriptions/ Remarks
1.	Mohamed Talaat	Mohamedtalaat0111790@gmail.com	1	20/3/2018		5 System Deployment
2.	Khaled sabry	Khaledsab1997@gmail.com	2	22/3/2018		3.1 Design Patterns Description
3.	Khaled sabry	Khaledsab1997@gmail.com	3	1/4/2018		3.3 Interaction Diagram
4	Mohamed Talaat	Mohamedtalaat0111790@gmail.com	4	2/4/2018		3.2 class diagrams
5	Khaled sabry	Khaledsab1997@gmail.com	5	4/4/2018		4 Data Models (ER)
6	Khaled sabry	Khaledsab1997@gmail.com	6	5/4/2018		6 Traceability to Requirements
7	Mohamed Talaat	Mohamedtalaat0111790@gmail.com	7	5/4/2018		2 System Architecture
8	Ahmed Adel	-	8	7/4/2018		1 Introduction
9	Kareem	kareemomar96@gmail.com	9	8/4/2018		Revision

Distribution list

Name	E-mail	Notes
Eng. Ali Elseddeek	ALIELSEDDEEK@GMAIL.COM	

Table of Contents

List of Figures

1. Introduction

This SDS will cover the software easy c++.

Easy c++ will be a tool to help students to understand the concepts of the c++ programming language and provide a community to the students and their instructors.

1.1 Purpose of this Document

This Software Design Specification (SDS) exists to establish a baseline for the technical design of easy c++ app.

This document is primarily a blueprint for Team 15 to use in implementing the project.

1.2 Scope

Easy C++ is an educational android app used for explaining programming concepts in C++ language. In it you can take lessons at your own pace, test your knowledge in its quizzes, interact with fellow learners and ask instructors for help, there is discussion room where you could put your question or see other's questions and see the replies or you can put your reply if you know the answer.

After you get to the last level you can apply to become an instructor.

1.3 Table of Acronyms and Definitions

Term	*Definition*
SRS	software requirements specification

1.4 References

[1] "Software Engineering", 10th Edition, Ian Sommerville.
[2] www.SmartDraw.com
[3] "Mastering UML with Rational Rose 2002", Wendy Boggs, Michael Boggs, SYBEX, 2002.
[4] "Design Patterns Explained Simply", Alexander shvits.

1.5 Overview of Document

This document consists of several sections all to demonstrate the design and structure of The app.

- Architectural Design

  Architectural design describes the hardware, software, and their interplay within the easy c++ application.

  The architectural design of easy c++ will include the communication between the data models, logic controller and interface.

- Class Diagrams

  class diagrams describe how the classes connected to each other.

- Interaction Diagrams

  Describe how the system will work according to user interaction.

- Entity relationship diagram

  illustrate the logical structure of databases.

2. System Architecture

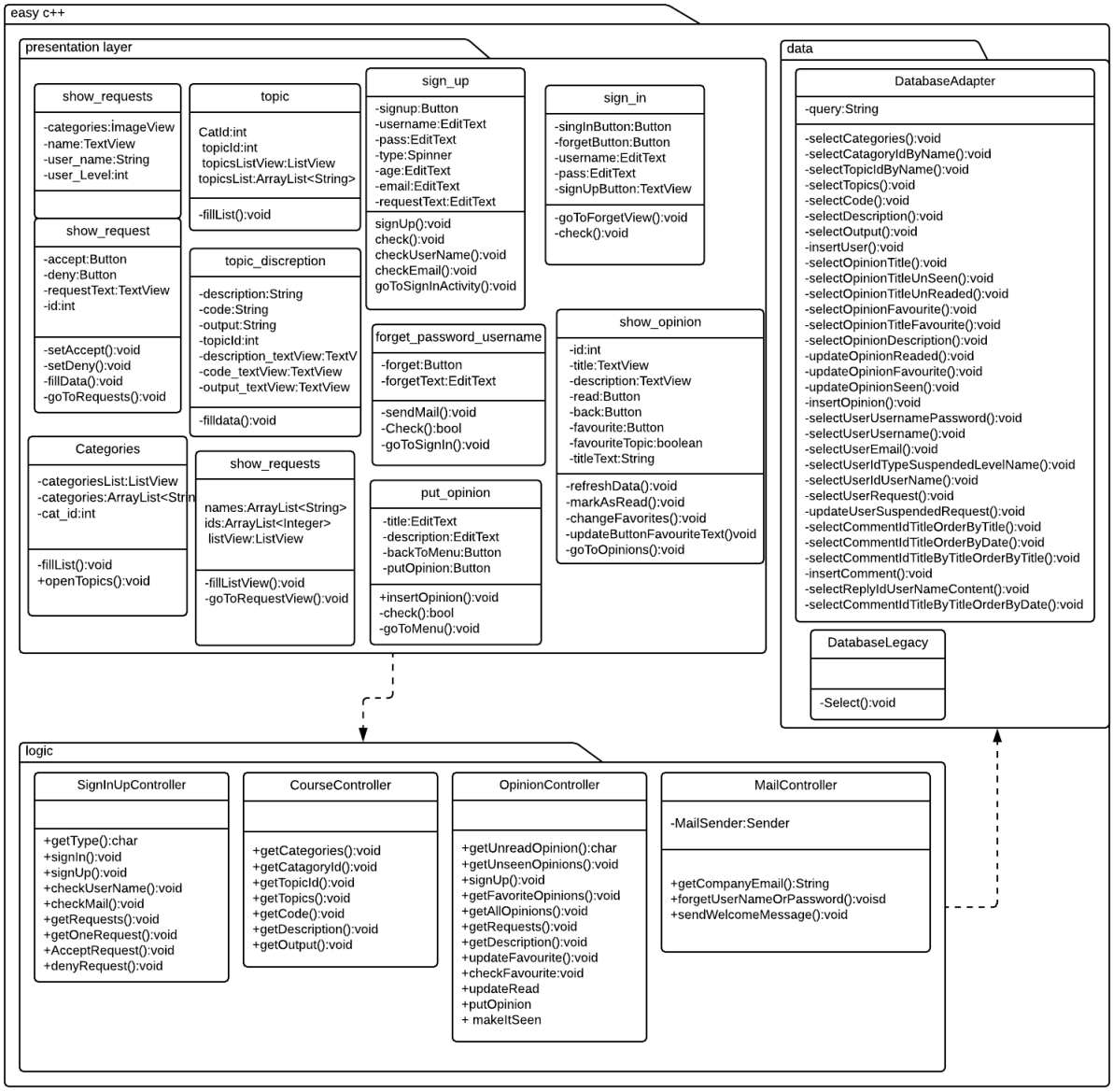


Figure 1 – System Architecture

3. Design Models

3.1 Design Patterns Description

3.1.1 Design Pattern 1

Name: Singleton.

Location : ConnectionDb.

Reason: to have only a single connection with the database for the whole application.

Name: Singleton.

Location: User-Data.

Reason: to have only one user date while the app is running.

3.1.2 Design Pattern 2

Name: Adapter.

Location: Database-Adapter.

Reason: the database Legacy can do (insert, update, delete, select) with the database but what if we want to insert into specific table so in Database-Adapter we handle the query and then send it to the Database-Legacy to handle the query.

3.2 Class Diagrams

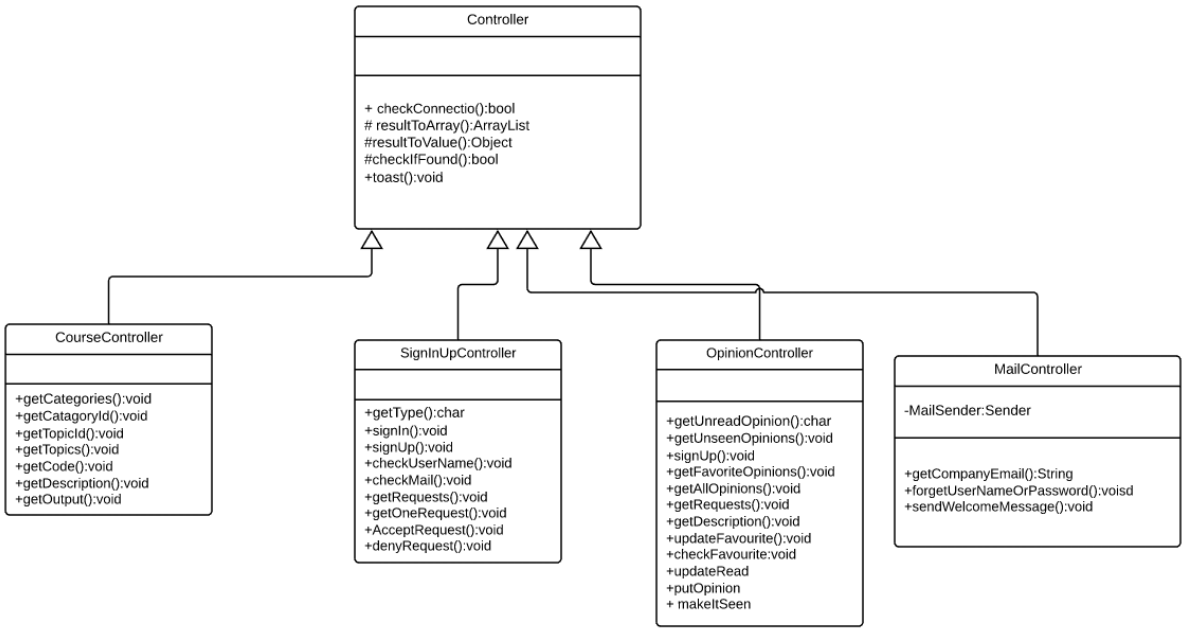


Figure 2 – Controllers

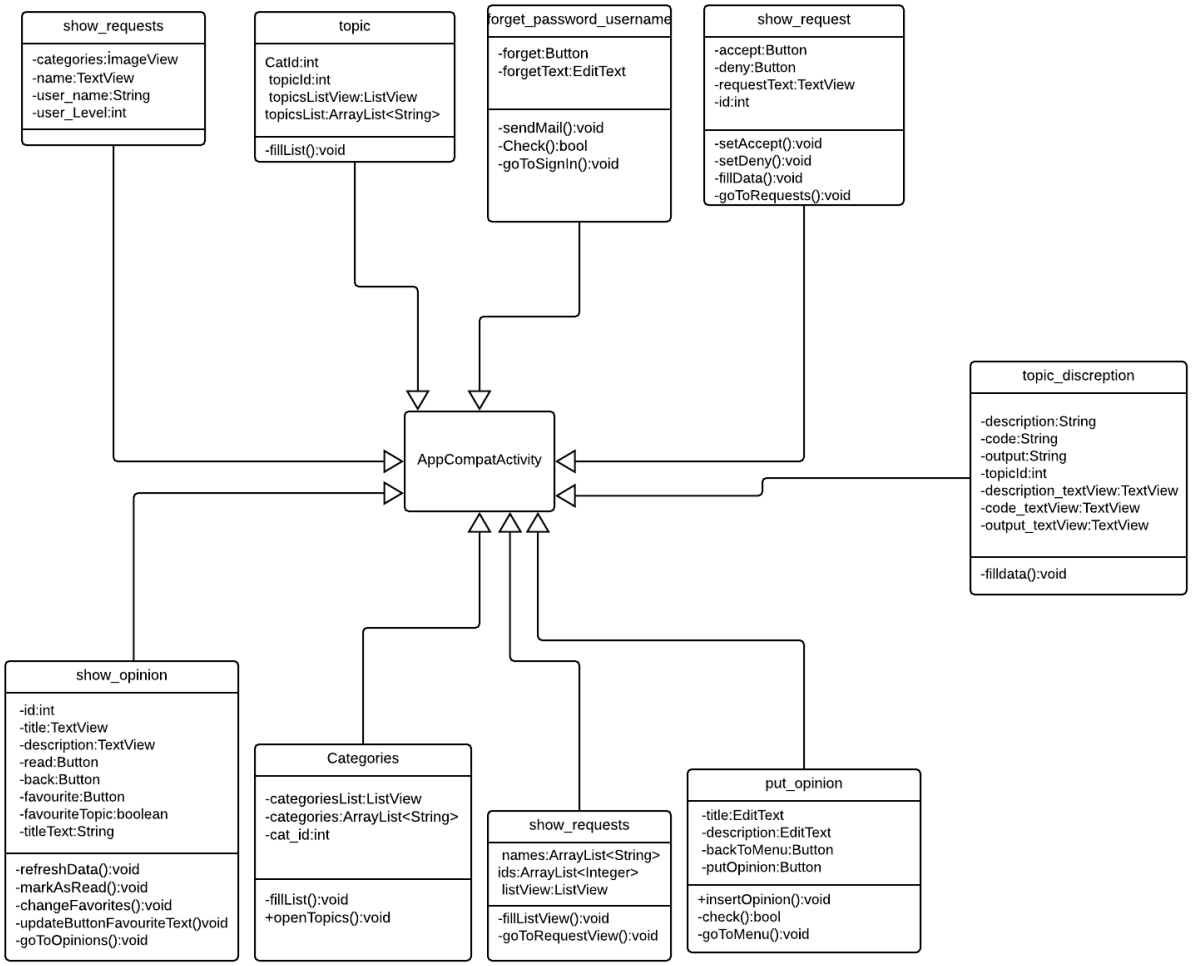


Figure 3 – Views

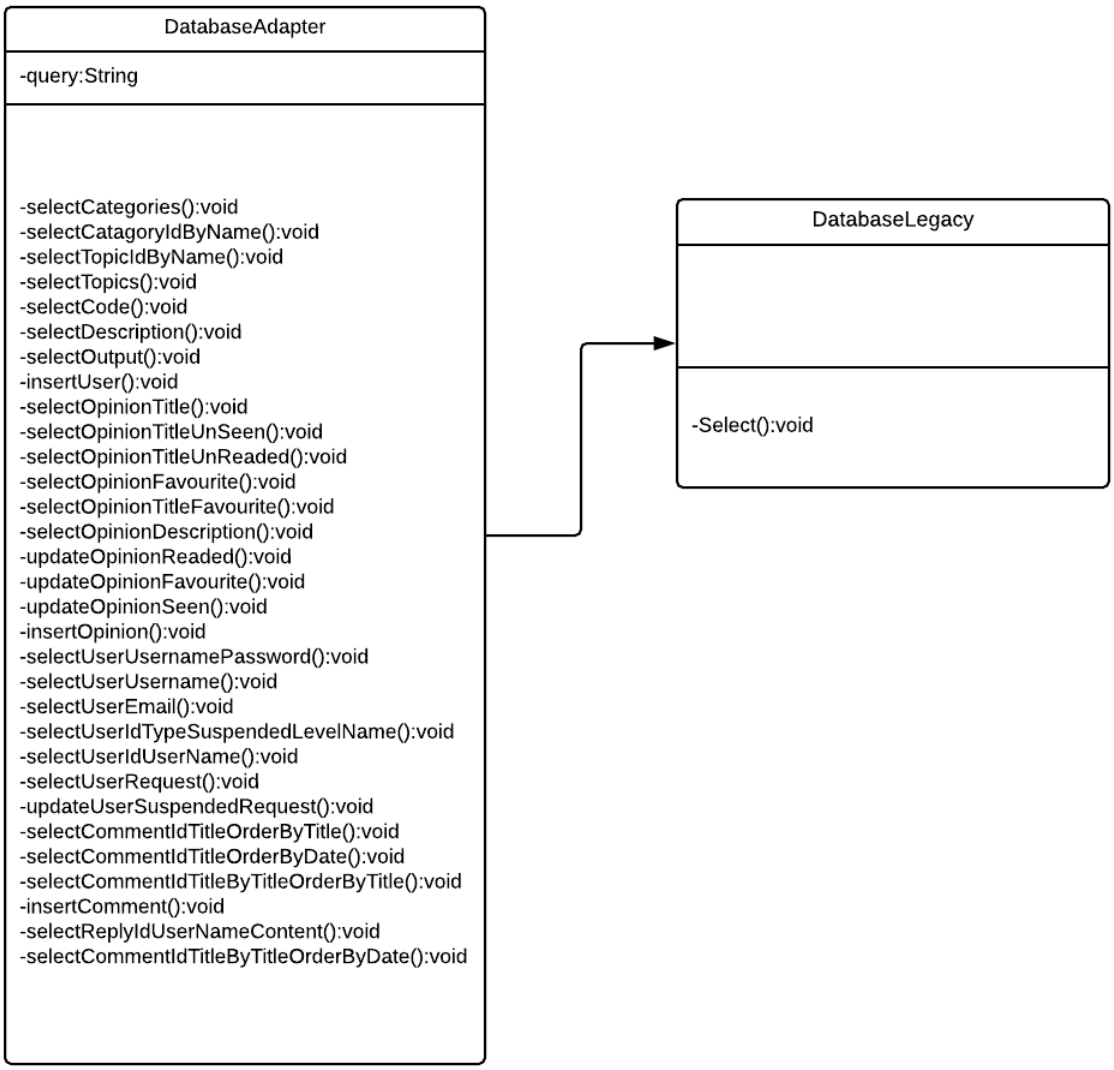


Figure 4 – Models

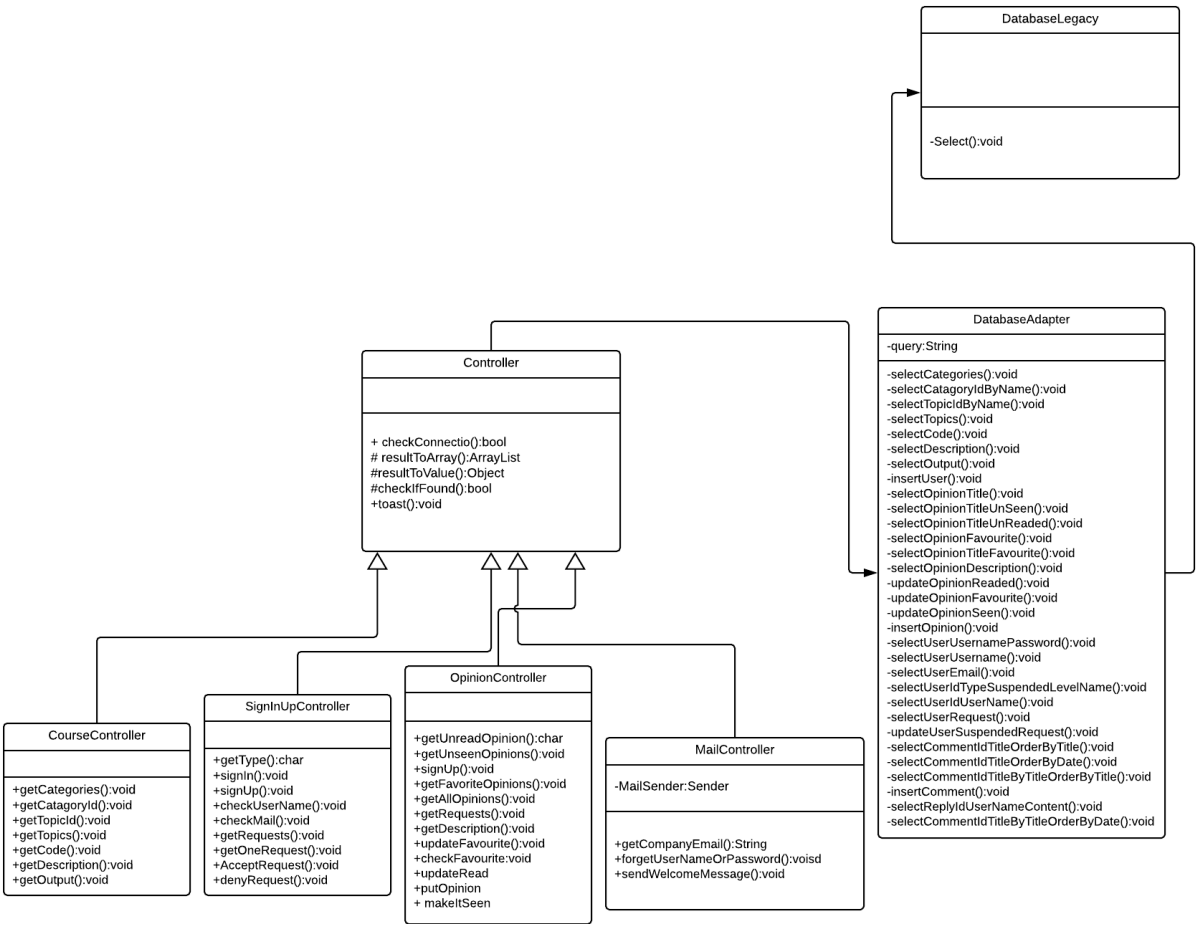


Figure 5– Model-Controller

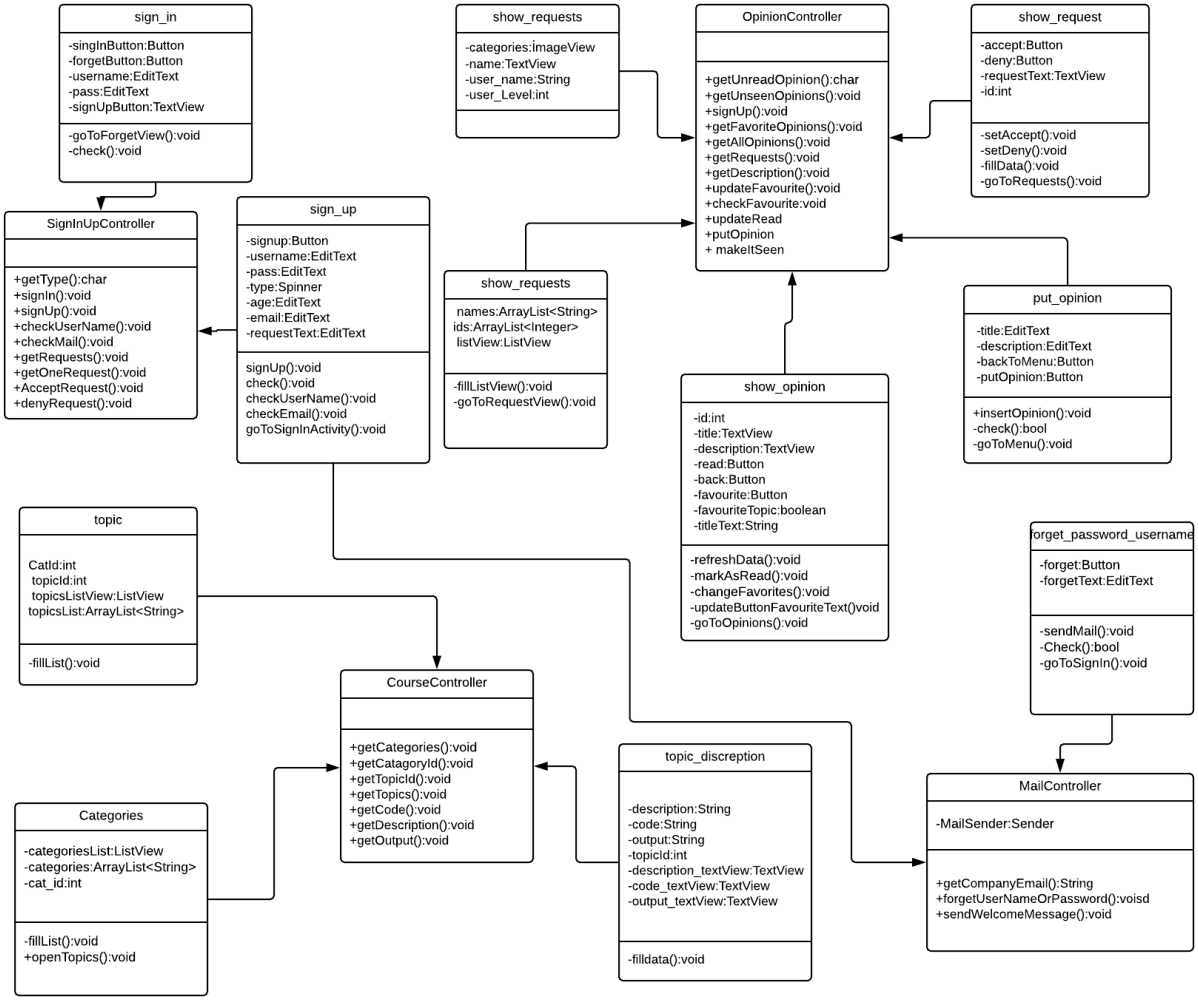
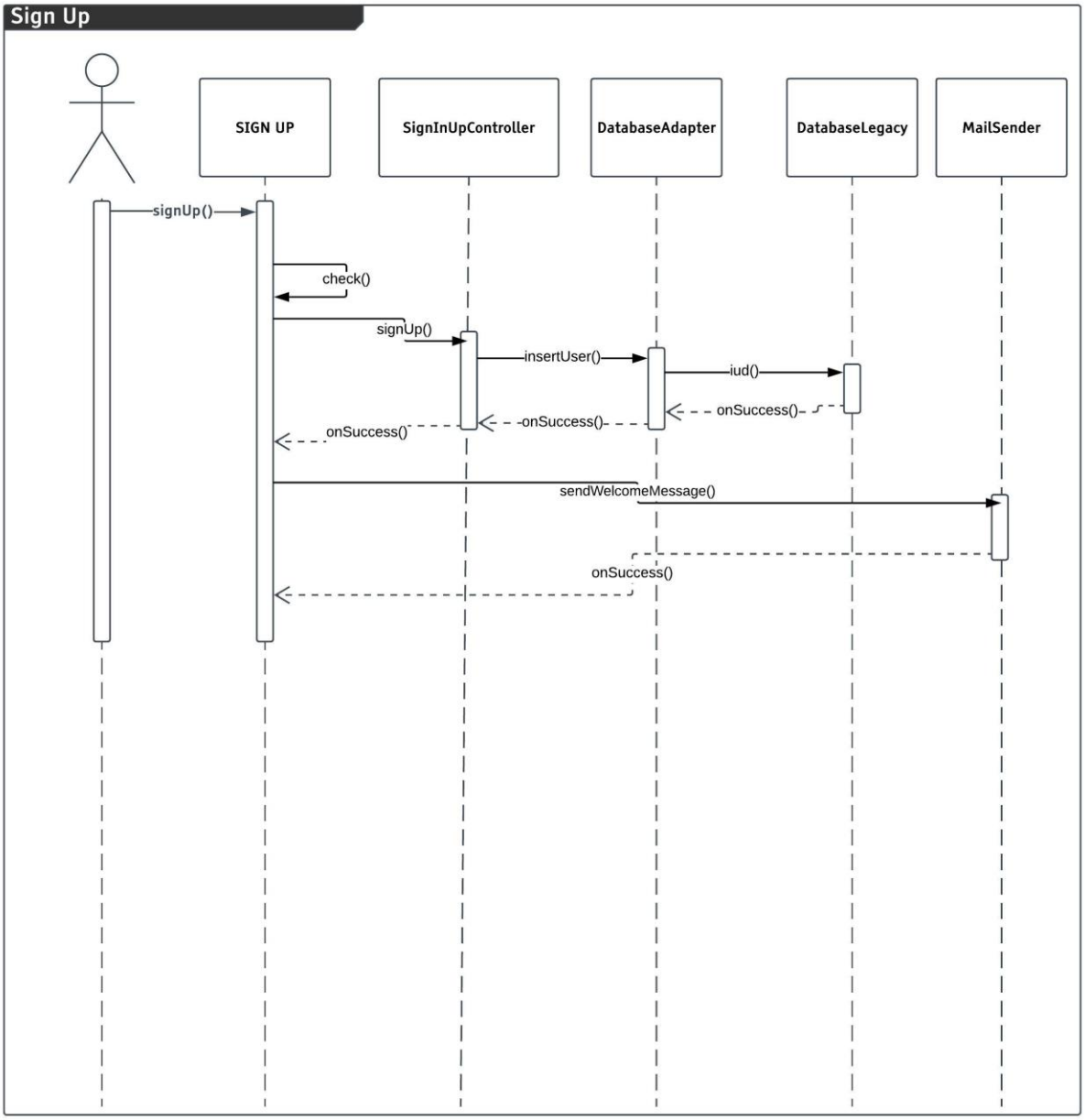
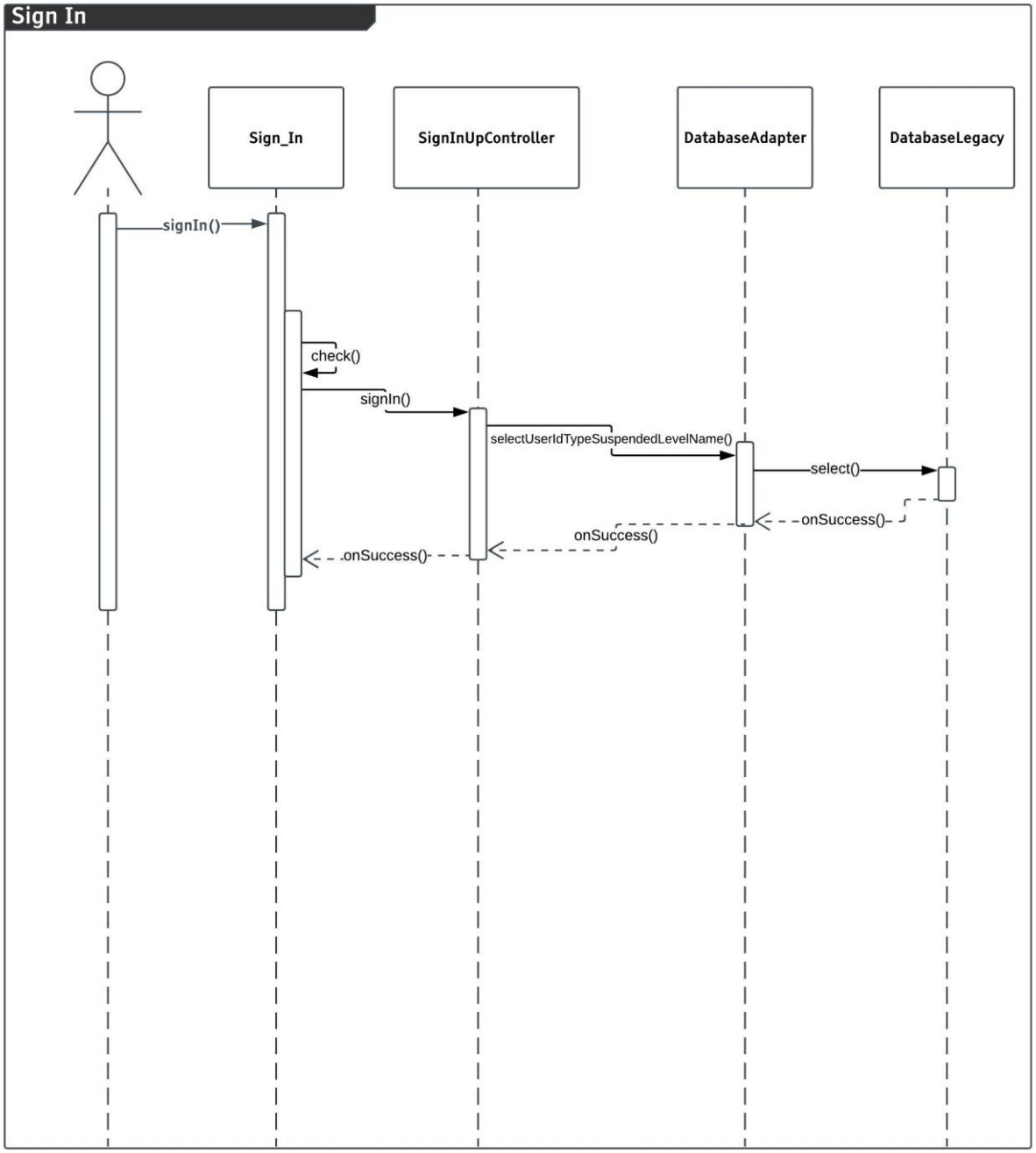


Figure 6 – Views-Controller

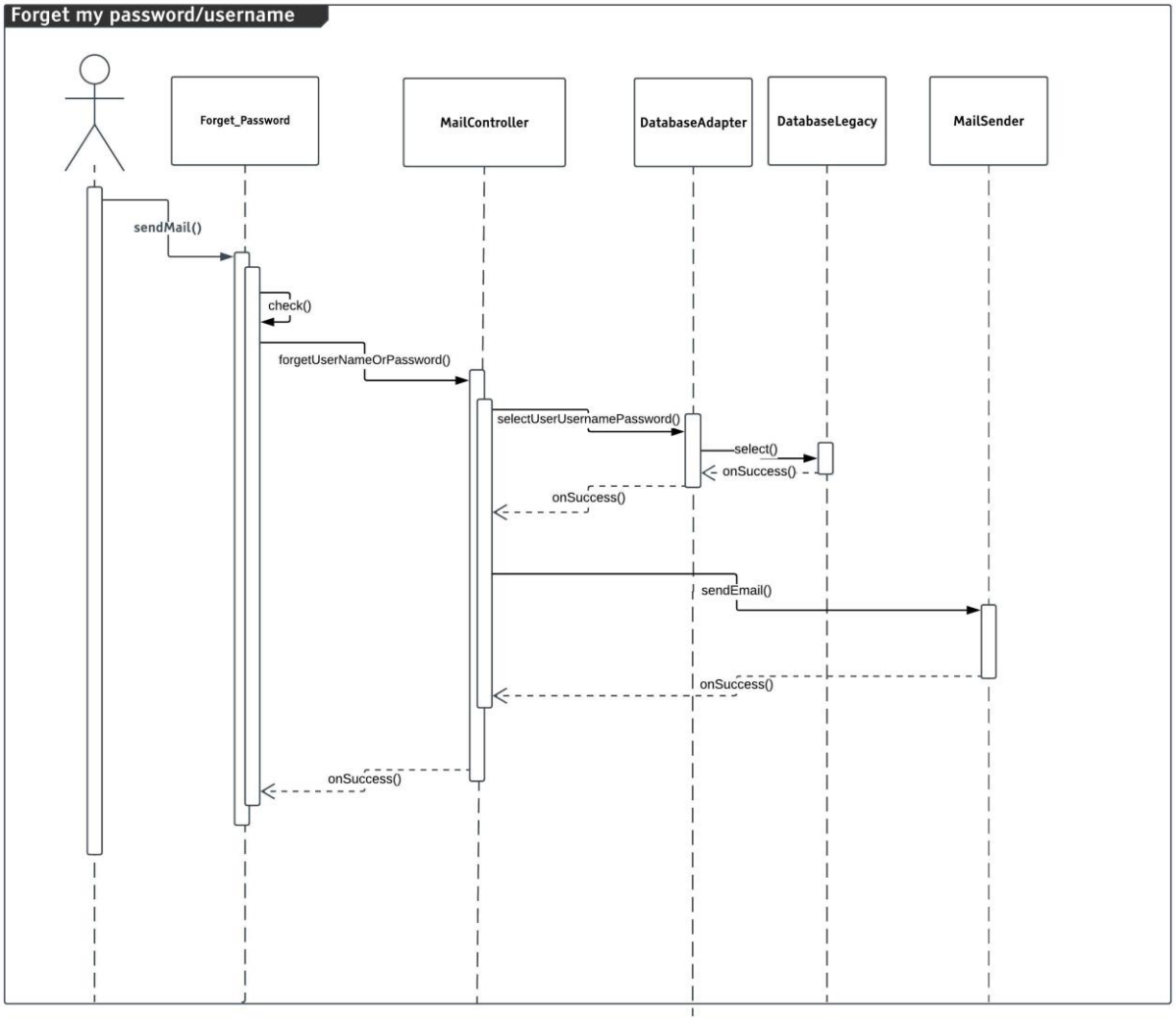
3.3 Interaction Diagrams



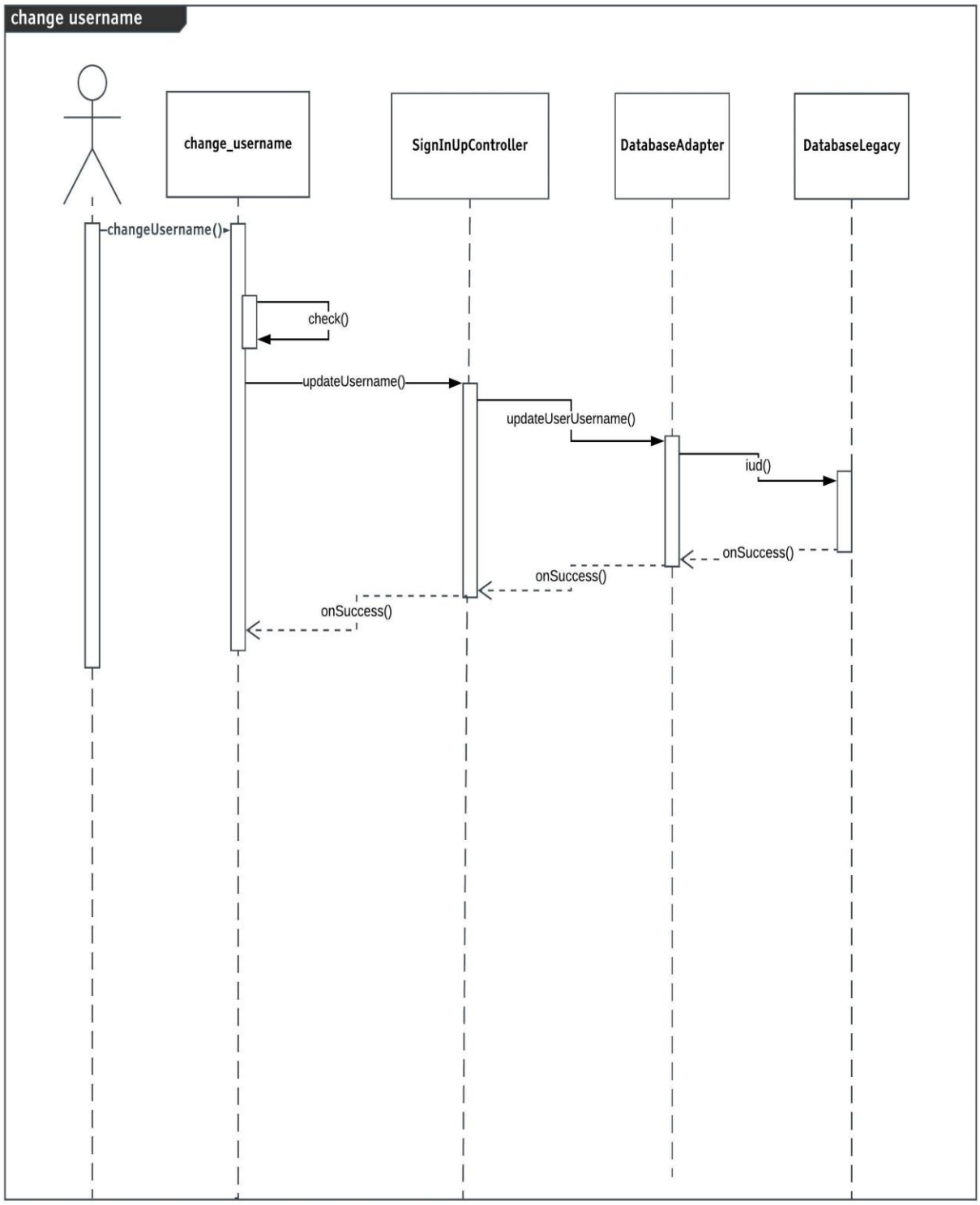
3.3.1 Sign Up


Sign In
Sign_In
SignInUpController
DatabaseAdapter
DatabaseLegacy
signIn()
check()
signIn()
selectUserIdTypeSuspendedLevelName()
select()
onSuccess()
onSuccess()
onSuccess()

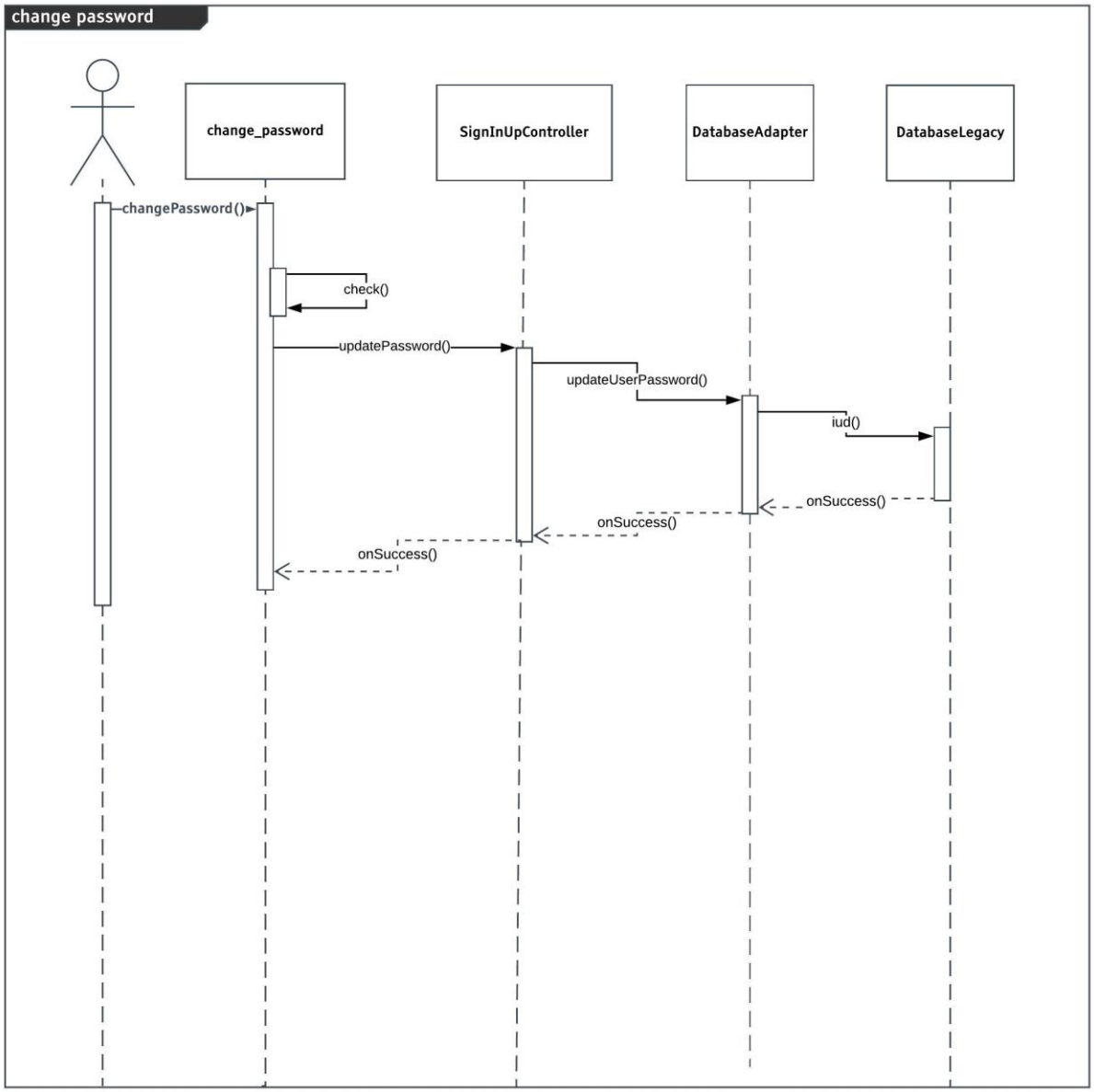
3.3.2 Sign In



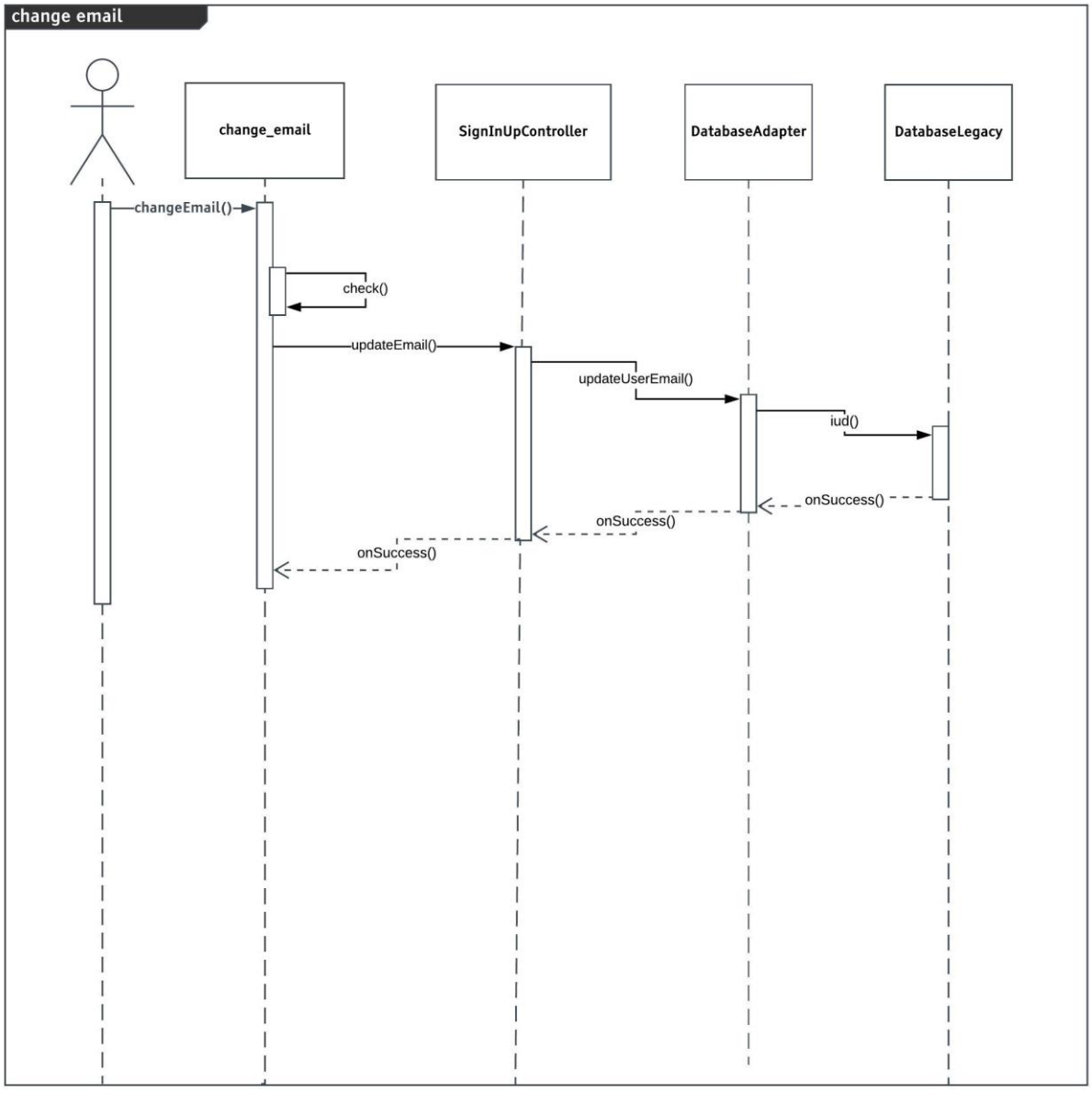
3.3.3 Forget my password / username



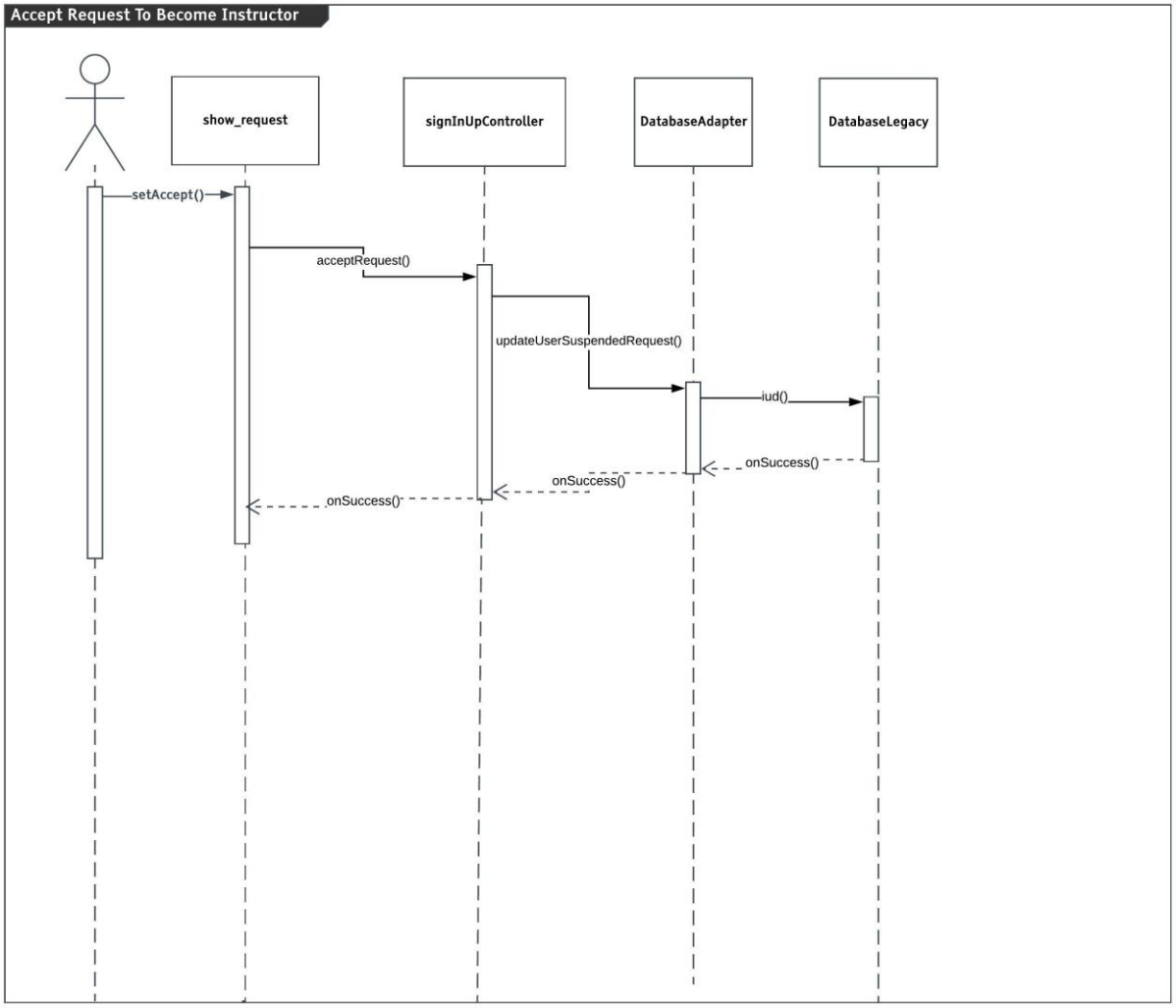
3.3.4 Change Username



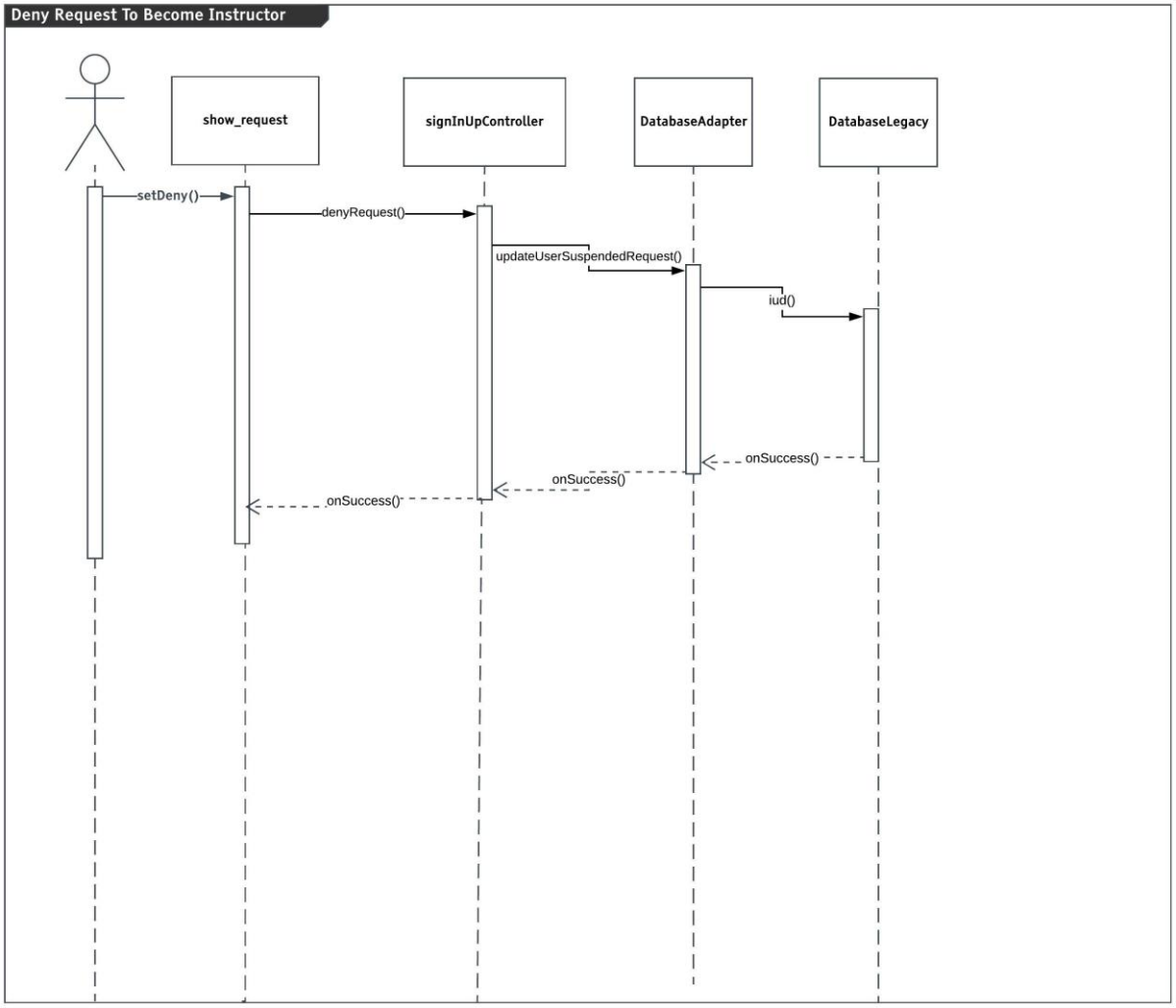
3.3.5 Change Password



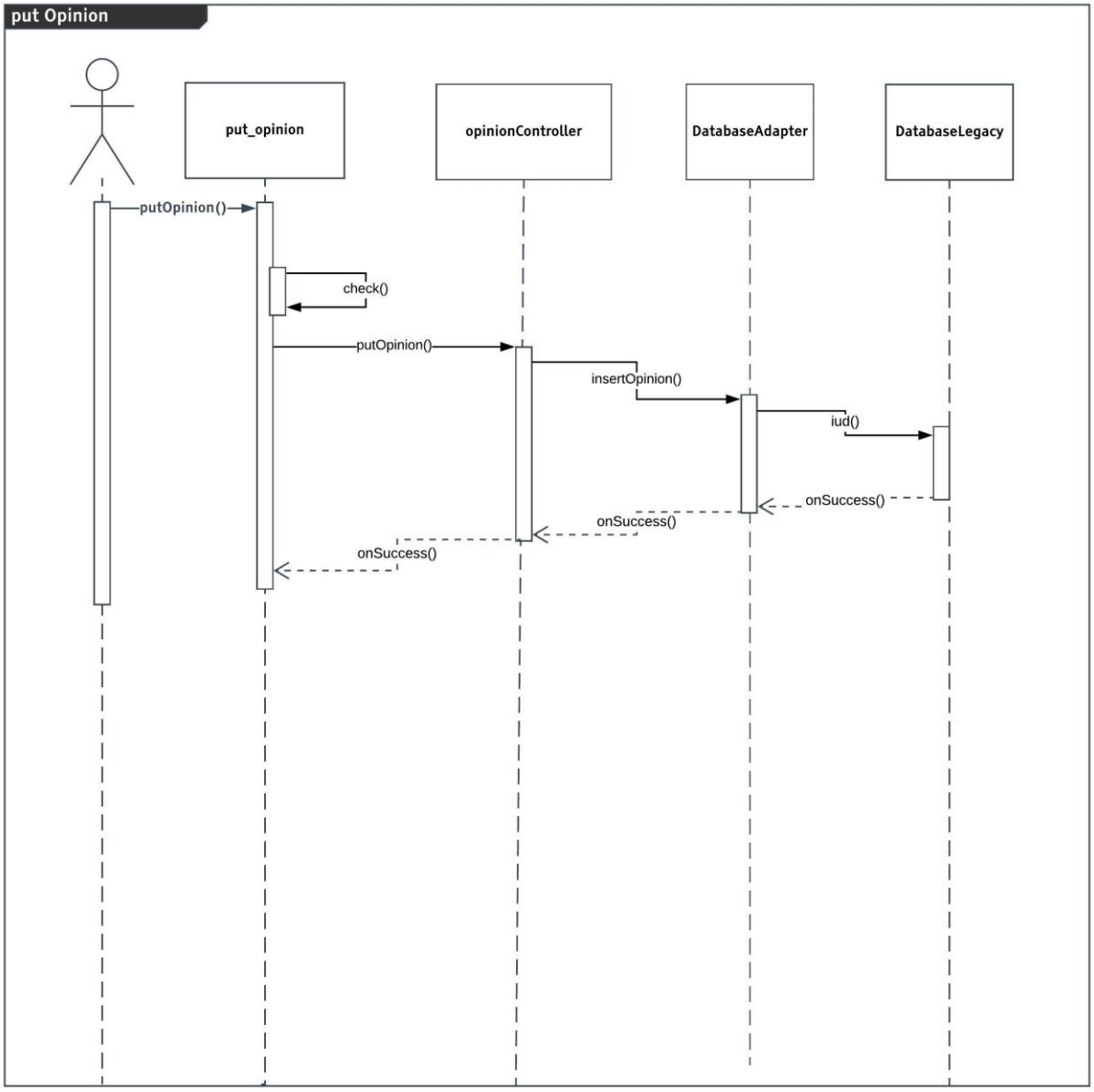
3.3.6 Change email



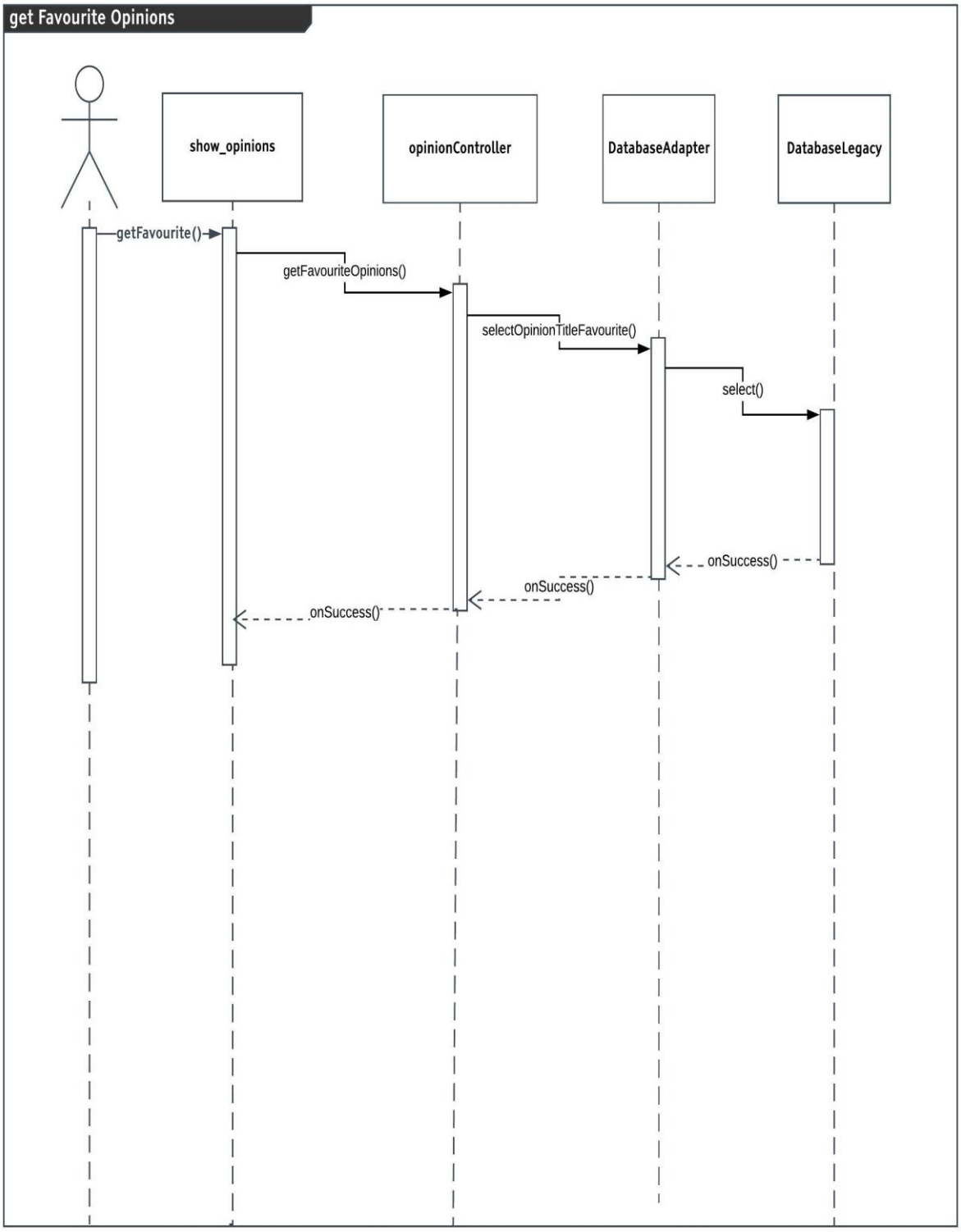
3.3.7 Accept request to become instructor



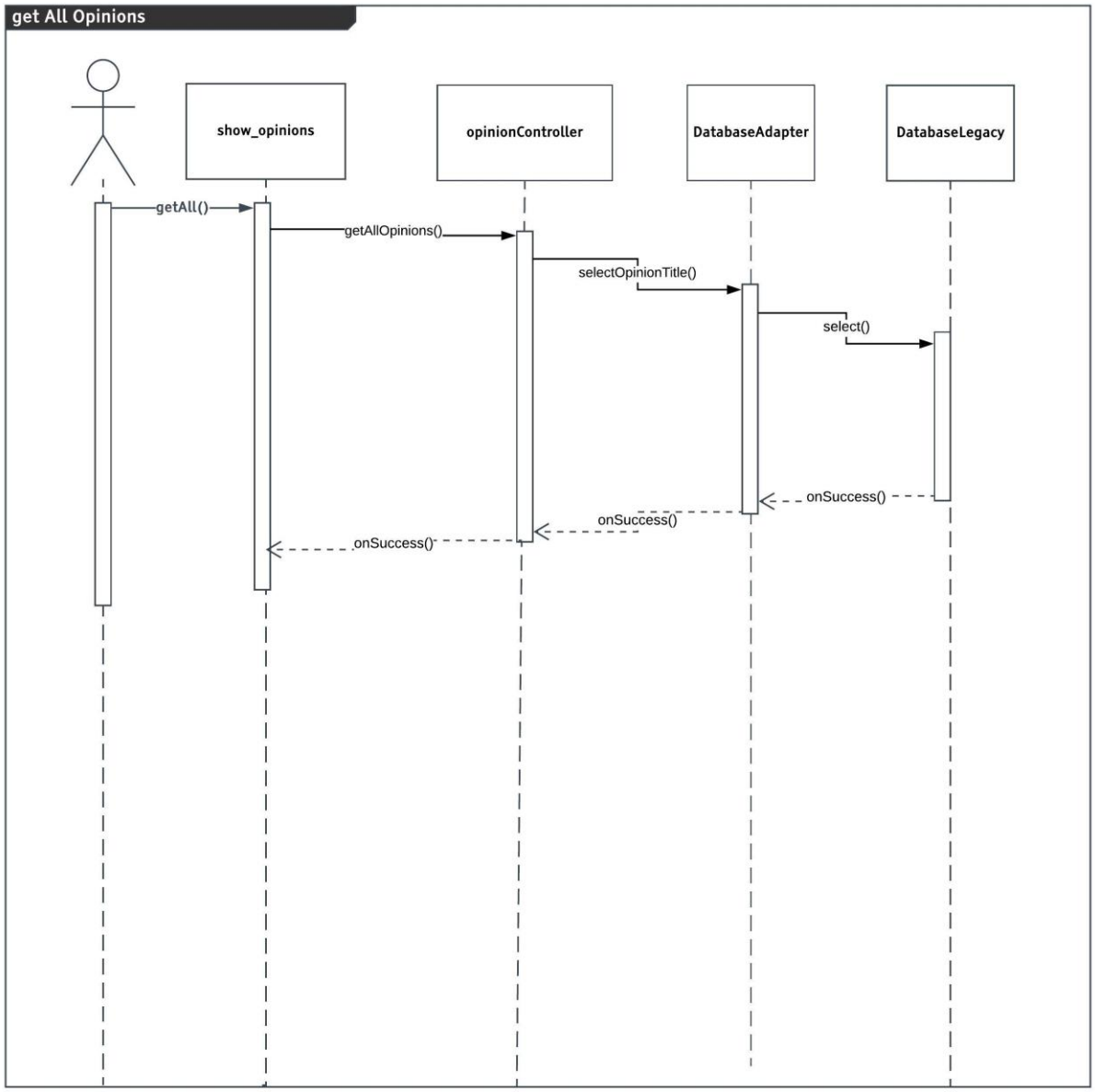
3.3.8 Deny Request to become instructor



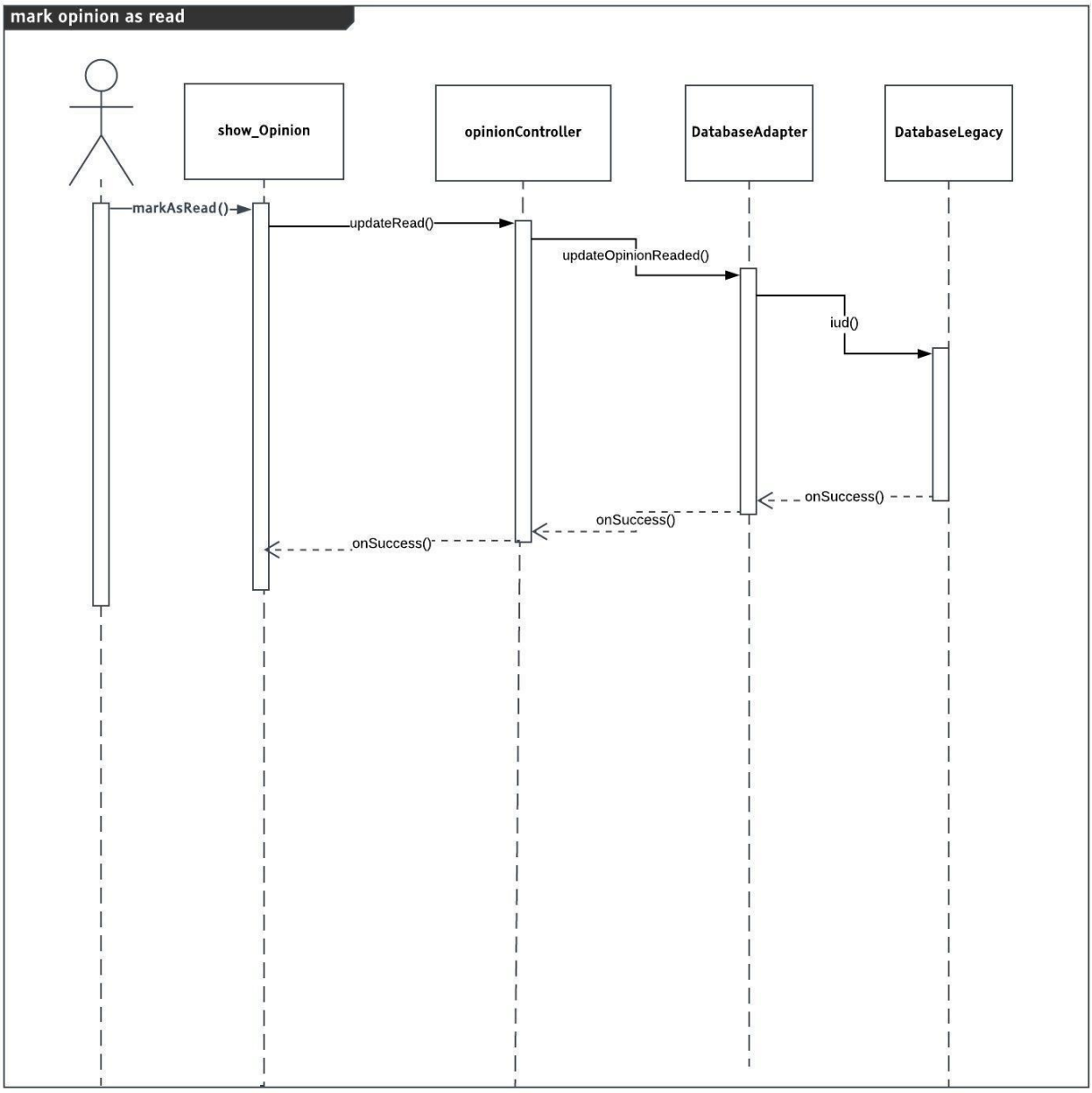
3.3.9 Put opinion



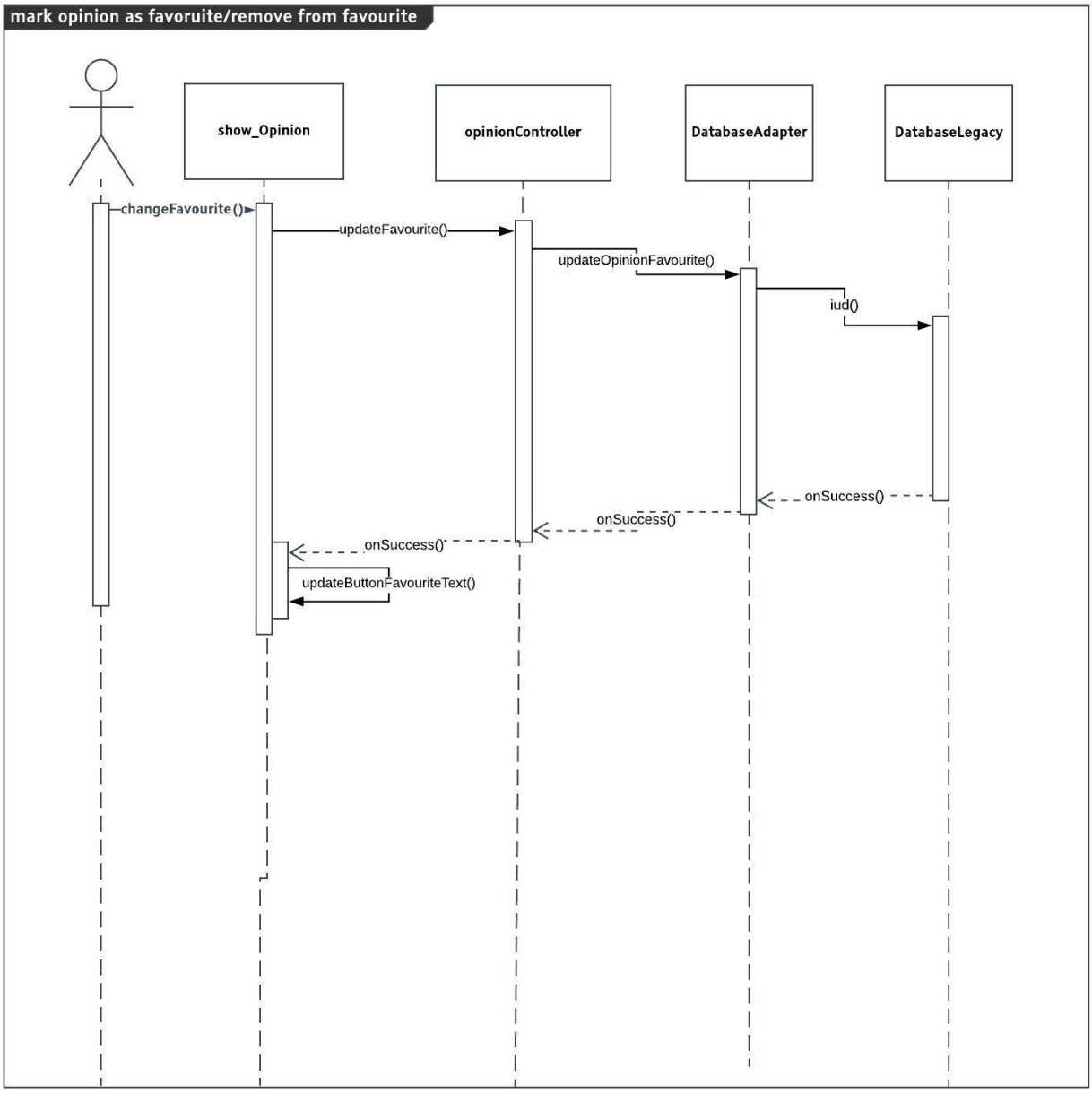
3.3.10 Get favorite opinions



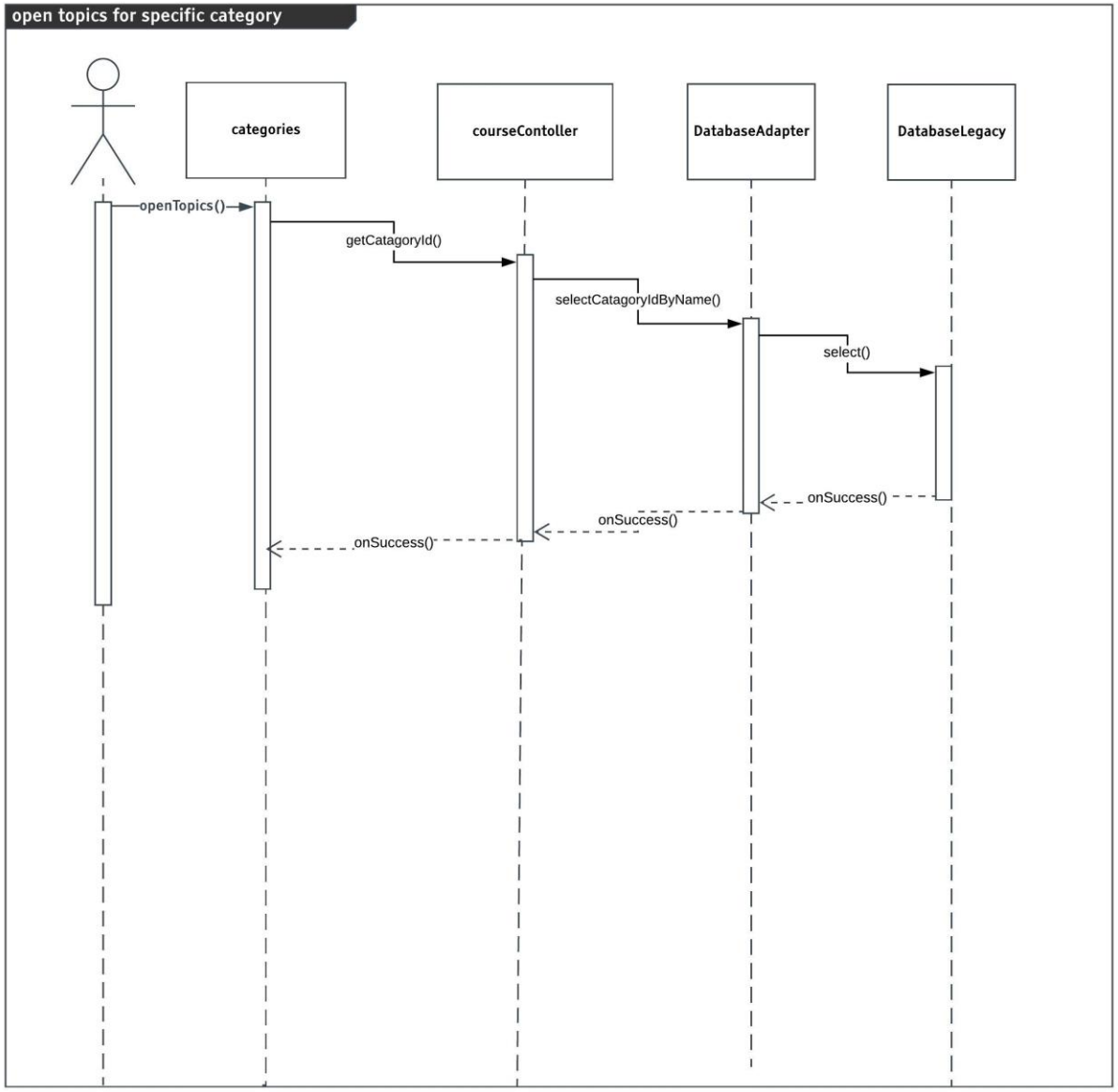
3.3.11 Get all opinions



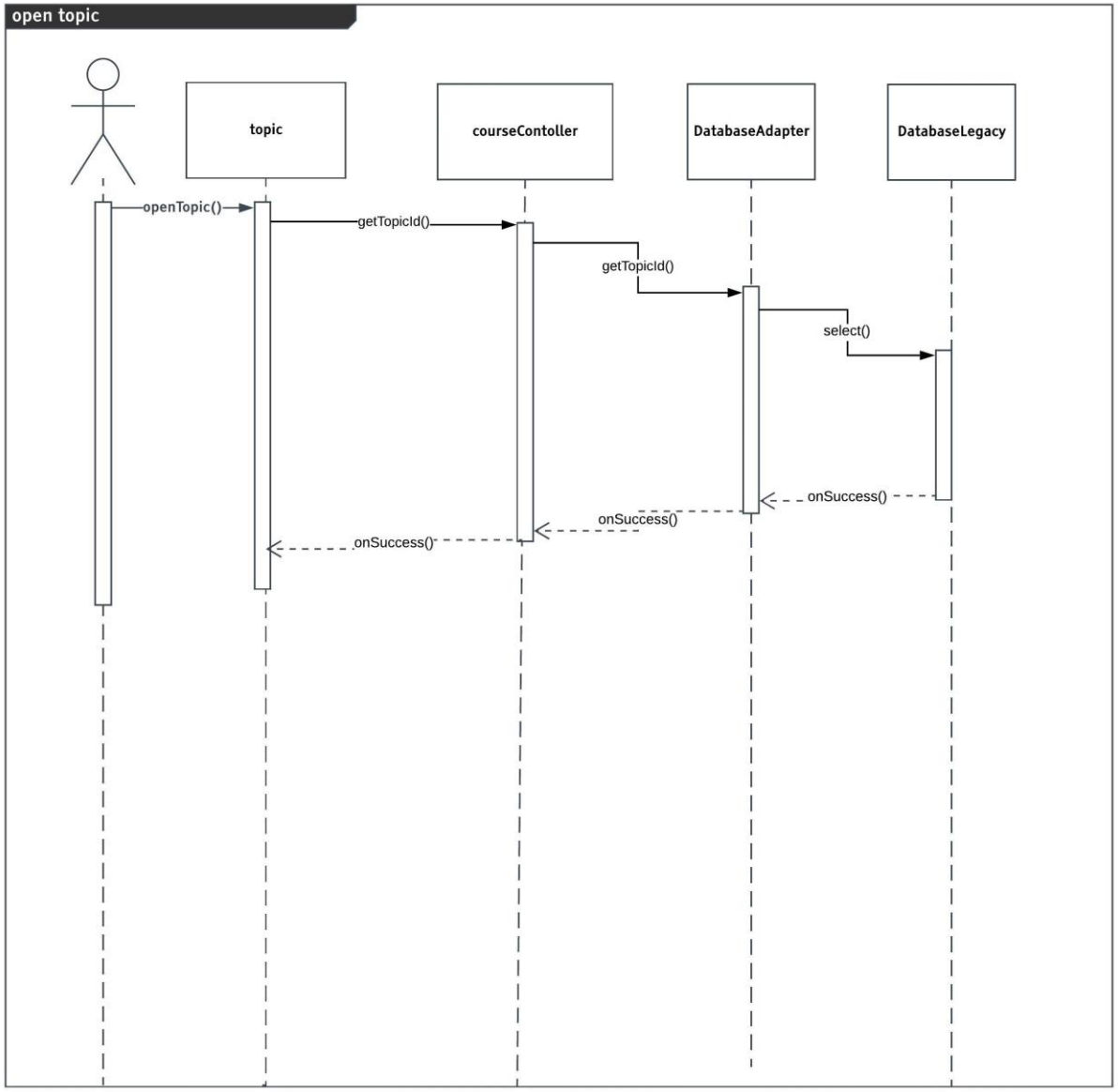
3.3.12 Make opinion as read



3.3.13 Put opinion as favorite or remove it from favorite opinions

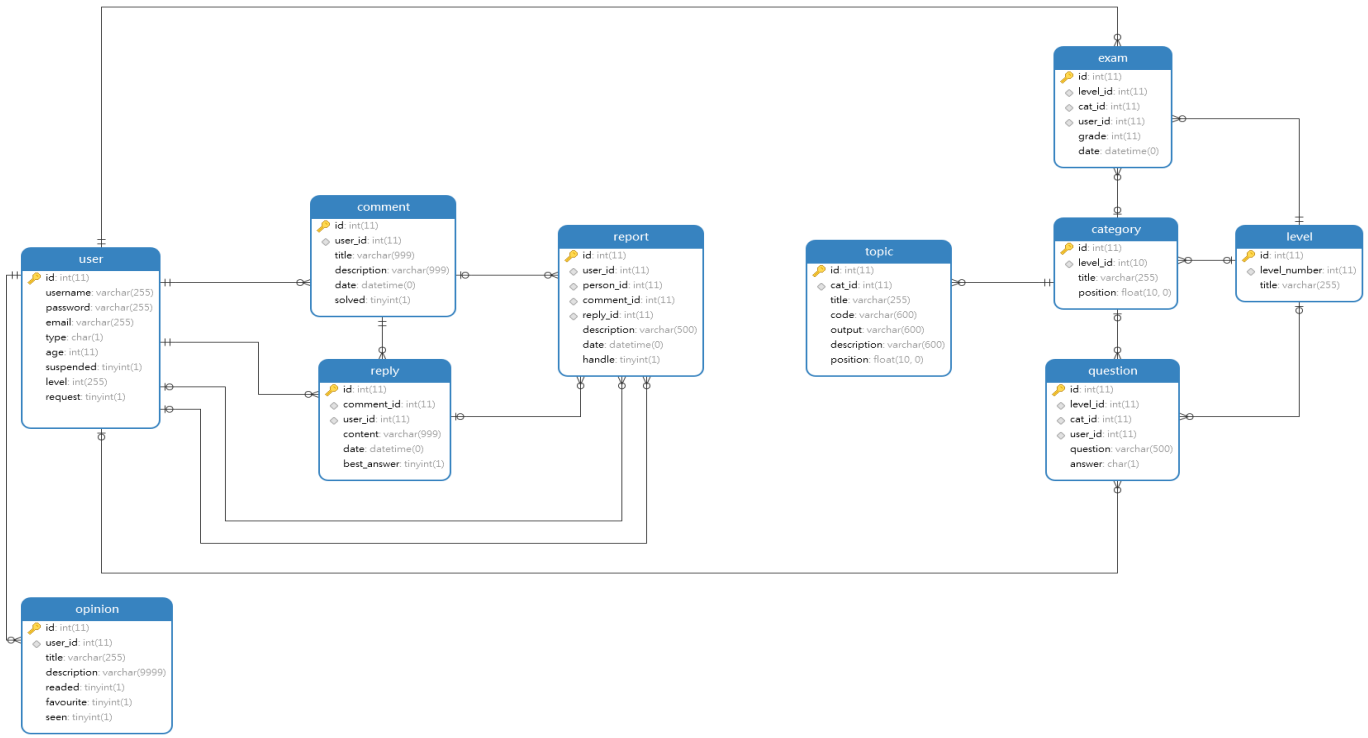


3.3.14 Open topics for specific category



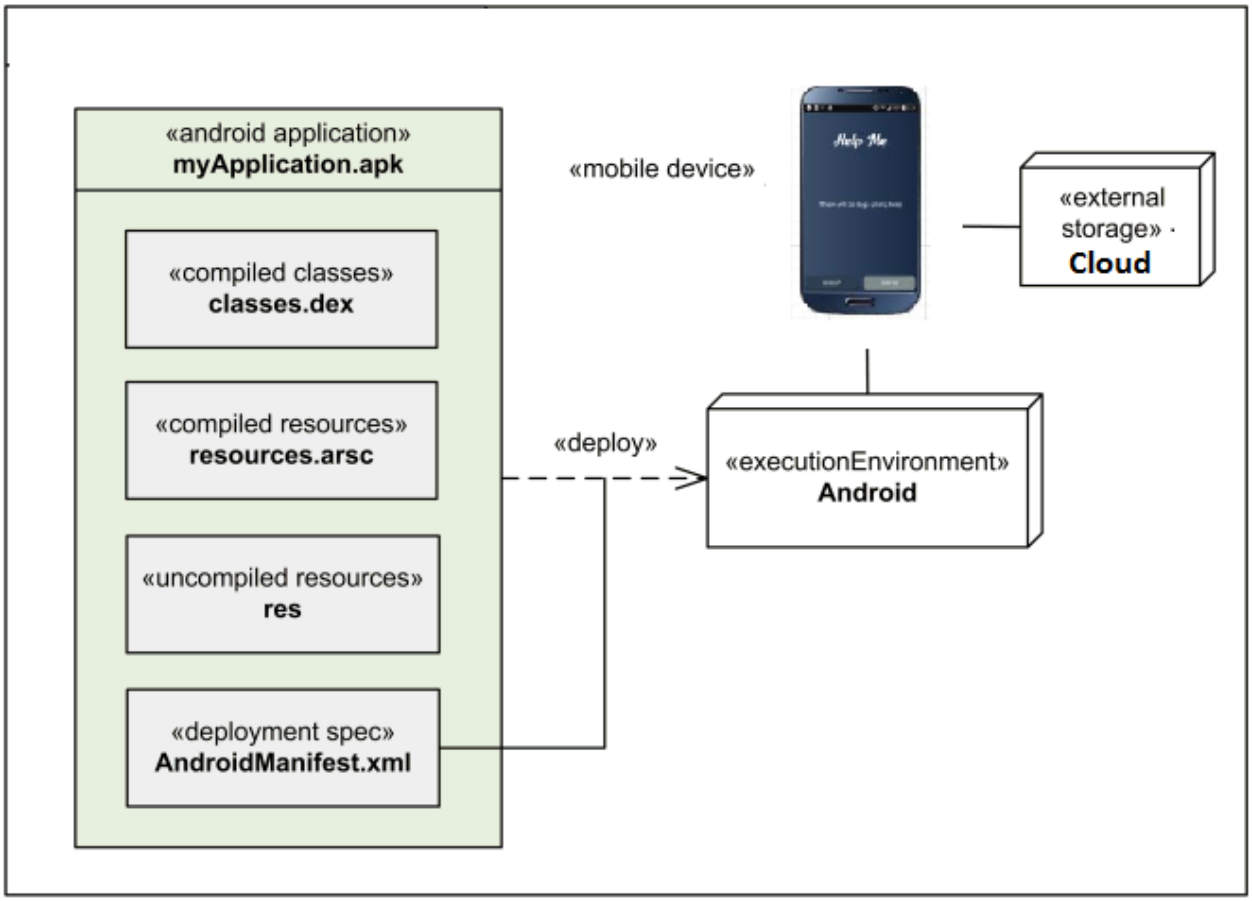
3.3.15 Open topic

4. Data Models



ER-Diagram

5. System Deployment



6. Traceability to Requirements

This table shows the relation among user stories in SRS document and Design Models diagrams

	Interaction diagram
US-01	**3.3.1**
US-02	**-**
US-03	**-**
US-04	**-**
US-05	**-**
US-06	**3.3.4 – 3.3.5 – 3.3.6**
US-07	**-**
US-08	**3.3.1**
US-09	**-**

US-10	**-**
US-11	**-**
US-12	**-**
US-13	**-**
US-14	**-**
US-15	**3.3.14 – 3.3.15**
US-16	**3.3.4 – 3.3.5 – 3.3.6**
US-17	**3.3.2**
US-18	**3.3.7 – 3.3.8**
US-19	**-**
US-20	**3.3.7 – 3.3.8**
US-21	**3.3.1 - 3.3.2**
US-22	**3.3.9**
US-23	**3.3.4 – 3.3.5 – 3.3.6**
US-24	**3.3.2**
US-25	**3.3.3**
US-26	**-**
US-27	**-**