|  |
| --- |
|  |
| Capstone Project Document |

**BOOKAHOLIC SOCIAL NETWORK**

Report #2 – Class Design

|  |  |  |
| --- | --- | --- |
| **Bookaholic Social Network** | | |
| **Group Members** | Phạm Thị Huyền | SE03416 |
| Thái Thị Cẩm Vân | SE03409 |
| Vũ Hải Đăng | SE03380 |
| Nguyễn Thị Hải Yến | SE03171 |
| Chu Minh Hải | SE03143 |
| **Supervisor** | Mr. Nguyễn Văn Sang | |
| **Project code** | BSN | |

**- Hanoi, 10/2016 -**

# SIGNATURE PAGE

AUTHOR: Phạm Thị Huyền --/--/2016

Project manager

REVIEWERS: --/--/2016

Team member

APPROVAL: Nguyễn Văn Sang --/--/2016

Supervisor

Record of change

\*A - Added M - Modified D – Deleted

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Effective Date | Changed Item | A,M,D | Change Description | Reason for Change | Rev. Number |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

TABLE CONTENTS

[SIGNATURE PAGE 0](#_Toc469237535)

[Introduction 2](#_Toc469237536)

[1.1 Purpose 2](#_Toc469237537)

[1.2 Definitions, Acronyms and Abbreviations 2](#_Toc469237538)

[2 Package 2](#_Toc469237539)

[2.1 Package Diagram 2](#_Toc469237540)

[2.2 Package Description 3](#_Toc469237541)

[2.2.1 Model 3](#_Toc469237542)

[2.2.2 0](#_Toc469237543)

[3 Function 0](#_Toc469237544)

[3.1 Guest Module 0](#_Toc469237545)

[3.1.1 Register 0](#_Toc469237546)

[3.2 Reader/Author Module 0](#_Toc469237547)

[3.2.1 Account 0](#_Toc469237548)

[3.2.2 Post 1](#_Toc469237549)

[3.2.3 Interaction 3](#_Toc469237550)

[3.2.4 Group 3](#_Toc469237551)

[3.2.5 Book 4](#_Toc469237552)

[3.3 Admin Module 4](#_Toc469237553)

[3.3.1 User Accounts Management 4](#_Toc469237554)

[3.3.2 Groups Management 4](#_Toc469237555)

[3.3.3 Books Management 5](#_Toc469237556)

[3.3.4 Sliders Management 0](#_Toc469237557)

[3.3.5 Publishers Management 4](#_Toc469237558)

[3.3.6 Statistics 7](#_Toc469237559)

# Introduction

## Purpose

BSN class design document contains descriptions of all classes of the BSN. It includes: Class diagrams that describe the static relation of all classes in the system.

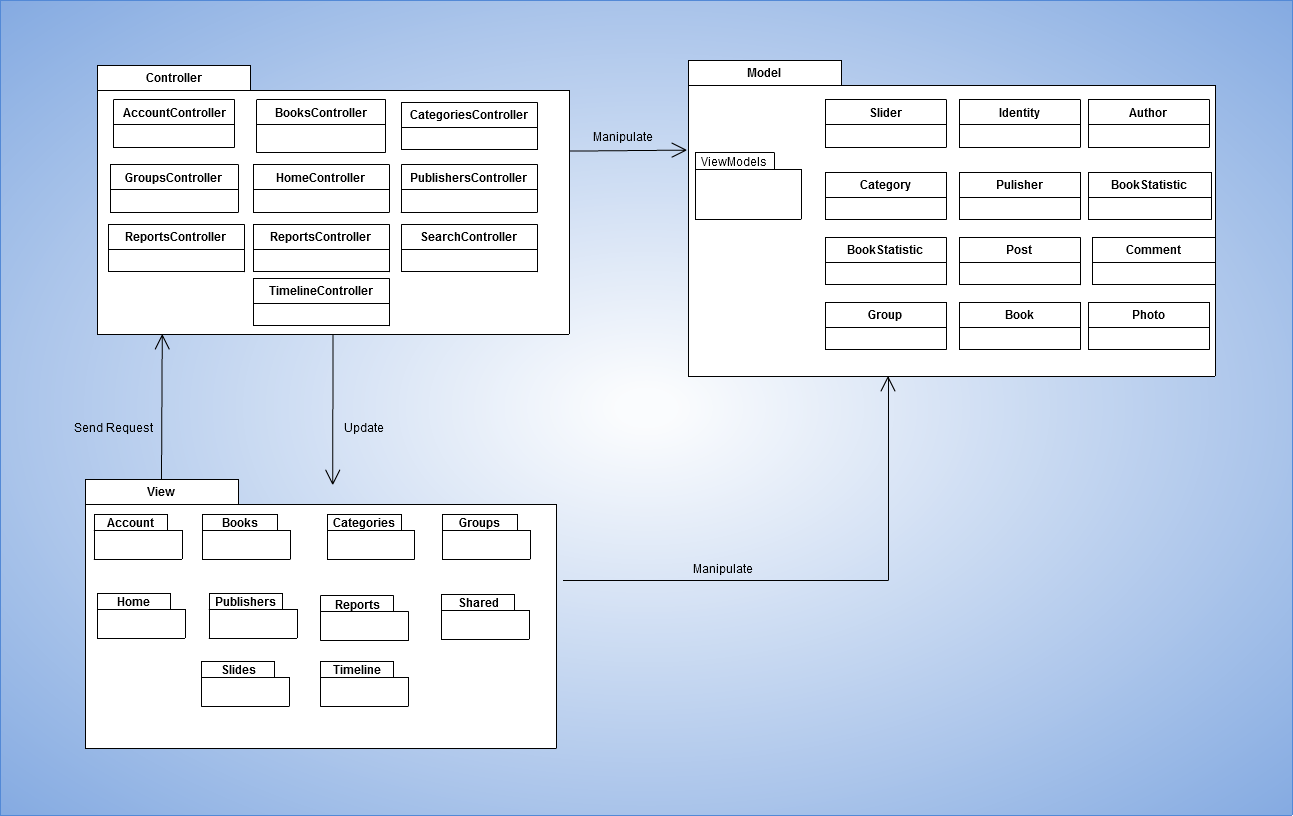
## Definitions, Acronyms and Abbreviations

|  |  |  |
| --- | --- | --- |
| Abbreviations | Description | Note |
| BSN | Bookaholic Social Network |  |
|  |  |  |

1. Definitions, Acronyms and Abbreviations

# Package

## Package Diagram



1. Package Diagram

## Package Description

### Model

#### BSN\_User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
|  | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
|  | UserName | String |  | Public | User’s name |
|  | Email | String |  | Public | User’s Email |
|  | EmailConfirmed | Boolean |  | Public | Status of registration confirmation via email |
|  | PasswordHash | String |  | Public | Password encrypt by SHA 256 |
|  | Role | String |  | Public | User’s role |
|  | Gender | String |  | Public | User’s gender |
|  | Avatar | String |  | Public | Public id of user’s avatar in cloudinary |
|  | Cover | String |  | Public | Public id of user’s cover in cloudinary |
|  | DOB | DateTime |  | Public | User’s date of birth |
|  | Address | String |  | Public | User’s address |
|  | SSNImgId | String |  | Public | User’s SSN image public id in cloudinary |

#### Post

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public |  |
| 2 | Content | String |  | Public |  |
| 3 | PostedBy | String |  | Public |  |
| 4 | PostedDate | DateTime |  | Public |  |
| 5 | BookTag | Array[String] |  | Public | List of book taged in user’s posst |

#### Book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | BookName | String |  | Public | Book’s name |
| 3 | ImgPublicId | String |  | Public | Book’s image publicid in cloudinary |
| 4 | ReleaseDay | DateTime |  | Public | Date of book release |
| 5 | Url | String |  | Public | Link detail of book |
| 6 | Description | String |  | Public | Content of book |
| 7 | RateTime | Int |  | Public | Sum rating point of each rate |
| 8 | RatingPoint | Int |  | Public | Sum rating time |
| 9 | AvarageRating | Double |  | Public | RatingPoint/RateTime |

#### Category

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | public | Mongodb objectid  Unique for each document |
| 2 | Name | String |  | public | Category’s name |

#### Author

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | UserId | ObjectId |  | Public | User’s id |
| 3 | AuthorName | String |  | Public | Name of Author |
| 4 | RatingPoint | Double |  | Public | Author’s rating point |

#### Publisher

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | Name | String |  | Public | Publisher’s name |
| 3 | PhoneNumber | String |  | Public | Publisher’s phone number |
| 4 | ImagePublicId | String |  | Public | Publisher’s image publicid in cloudinary |
| 5 | Address | String |  | Public | Publisher’s address |

#### Reports

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | UserId | ObjectId |  | Public | User’s id |
| 3 | GroupId | ObjectId |  | Public | Group’s id |
| 4 | Reporteduserid | ObjectId |  | Public | Id of user reported |
| 5 | Reportedgroupid | ObjectId |  | Public | Id of group reported |
| 6 | Content | String |  | Public | Content of report |
| 7 | Status | Boolean |  | Public | Status of report if admin accepted or decline report |

#### Group

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | GroupName | String |  | Public | Book’s name |
| 3 | Locked | Boolean |  | Public | Status of group (lock or not) |
| 4 | GroupType | String |  | Public | Group’s type |
| 5 | CreateDate | DateTime |  | Public | Date time group created |
| 6 | Tag | String |  | Public | Group’s tag(author’s name or book’s name) |
| 7 | CreatorId | String |  | Public | Id of user that created group |
| 8 | Description | String |  | Public | Content of group |
| 9 | CoverImg | String |  | Public | Group cover’s image public id in cloudinary |
| 10 | AvaImg | String |  | Public | Group avatar’s image public id in cloudinary |

#### Slider

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | ImagePublicId | String |  | Public | Slide’s image public id in cloudinary |
| 3 | Description | String |  | Public | Content of slide |
| 4 | Status | Boolean | True | Public | Slide’s status (show or not show) |

#### Book Statistic

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Type** | **Default** | **Note** | **Description** |
| 1 | \_id | ObjectId |  | Public | Mongodb objectid  Unique for each document |
| 2 | BookId | String |  | Public | Book’s id |
| 3 | EachDate | DateTime |  | Public | Each date of book statistic |
| 4 | Count | int |  | Public | Number of access time in book page each date |

### View

Contain many .cshtml files to display user interface.

### Controller

#### AccountController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
|  | AccountController | + userManager : ApplicationUserManager |  | Get user management |
|  | Login | + returnUrl : string | ActionResult | Show login form for user |
|  | Login | + model : AccountViewModel  + returnUrl : string | async | Handle login request from user |
|  | Register |  | ActionResult | Show normal user register form for user |
|  | AuthorRegister |  | ActionResult | Show author register request form for user |
|  | Register | model : AccountViewModel | async | Handle register for normal account |
|  | AuthorRegister | model : AccountViewModel | async | Handle register request for author account |
|  | ConfirmEmail | + userId : string  + code : string | async | Send email confirm for activate account to user |
|  | ForgotPassword |  | ActionResult | Show forgot password form for user |
|  | ForgotPassword | model : AccountViewModel | async | Handle input data from user then send email to reset password to user |
|  | ForgotPasswordConfirmation |  | ActionResult | Show confirm message when user confirm |
|  | ResetPassword | code : string | [AllowAnonymous] | Show reset password form for user |
|  | ResetPassword | model : ResetPasswordViewModel | async | Handle reset password |
|  | ResetPasswordConfirmation |  | ActionResult | Show confirm message when user confirm |
|  | LogOff |  | ActionResult | Handle log out |
|  | Users | + searchString : string  + currentFilter : string, page : int | ActionResult Authors | Get all users to show for admin manage |
|  | Authors | + searchString : string  + currentFilter : string, page : int | ActionResult | Get all authors to show for admin manage |
|  | AuthorConfirm | id : string | async | Admin confirm request when guest register as author account |

#### Book Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Index | + searchString : string  + currentFilter : string  + page : int | ActionResult | Get all books for admin manage |
| 2 | Details | id : string | ActionResult | Get book information that have book id equal with requested book |
| 3 | Create |  | ActionResult | Show book create form for admin to create new book |
| 4 | Create | book : Book | ActionResult | Accept input data and handle data then send back respond |
| 5 | Edit | id : string | ActionResult | Show book edit form with all information for admin to edit |
| 6 | Edit | book : Book | ActionResult | Accept input data and handle data then send back respond |
| 7 | DeleteConfirm | id : string | ActionResult | Handle delete request a book from admin |
| 8 | RatingBook | + id : string  + rating : int | ActionResult | Auto plus 1 to RatingPoint and RateTime each time user rate a book |
| 9 | CommentBook | + commentedUser : string  + bookId : string  + commentContent : string | ActionResult | Save user comment in book page |
| 10 | BookRequest | searchString : string | ActionResult | Handle data when user request a book |
| 11 | RequestAddBook | + bookName : string  + author : string  + categories : string | ActionResult | Handle user book request |

#### Category Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Index |  | ActionResult | Get all categories |
| 2 | Create | categoryName : string | ActionResult | Create new category |

#### Group Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | GetUserForAdd | + selectedValues : string [][]  + groupId : string | MultiSelectList | Get all user that following group admin for add new member function |
| 2 | GroupReportContent |  | List<string> | Get all group report content |
| 3 | Index | + searchString : string  + currentFilter : string  + page : int | ActionResult | Get all groups for admin manage |
| 5 | MainPage | id : string | ActionResult | Get information for group main page |
| 6 | Members | id : string | ActionResult | Get all member in group for member page |
| 7 | GroupRequestHandle | + groupId : string  + userId : string  + option : int | ActionResult | Handle all request to group (Join request, leave group, accept join request, set to admin, delete member…) |
| 8 | Setting | id : string | ActionResult | Get information of group for setting page |
| 9 | Setting | groupSetting : GroupSettingViewModel | ActionResult | Handle input data to update group information |
| 10 | Create | + groupName : string  + groupTag : string  + groupDesc : string  + groupType : string | ActionResult | Handle input data to create new group |
| 11 | ChangeImage | + id : string  + file : HttpPostedFileBase  + option : int | ActionResult | Handle change group avatar and group cover image and upload to cloudinary |
| 12 | AddNewMember | + addUser : string  + groupId : string | ActionResult | Add new member into group |

#### Home Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Index |  | ActionResult | Get all information, post of user and suggest books, authors for user |
| 2 | GetListBookTag | input : string | ActionResult | Get all books for book tag |
| 3 | LockedPage |  | ActionResult | Show locked page if user or group has been locked |

#### Timeline Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Details | id : string | ActionResult | Show user time line with all information |
| 2 | EditProfile | id : string | ActionResult | Show edit form for user change information |
| 3 | EditProfile | user : UserProfile | async | Edit user information |
| 4 | ChangeImage | + id : string  + file : HttpPostedFileBase, option : int | ActionResult | Handle change user avatar and user cover image |
| 5 | FollowHandle | + followerId : string  + followedUserId : string  + option : int | ActionResult | Handle follow or unfollow user |

#### Report Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Index | page : int | ActionResult | Get all user report for admin to manage |
| 2 | GetReport | + reportContent : string  + userId : string  + beReportedId : string  + option : int | ActionResult | Get report from user and save into database |
| 3 | Lock | + id : string  + option : int  + reportId : string | ActionResult | Handle lock or unlock user and group |
| 4 | Reject | id : ObjectId | ActionResult | Reject report user or group |

#### Publisher Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Get | page : int | ActionResult | Get list publishers |
| 2 | Upload | publisher : Publisher | ActionResult | Upload new publisher |
| 3 | Delete | id : string | ActionResult | Delete a publisher |
| 4 | Edit | publisher : Publisher | ActionResult | Edit a publisher’s information |

#### Slider Controller

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Method | Parameter | Return | Description |
| 1 | Get | page : int | ActionResult | Get list sliders |
| 2 | Upload | desc : string | ActionResult | Upload new slider |
| 3 | Delete | id : string | ActionResult | Delete a slider |
| 4 | Show | id : string | ActionResult | Show sliders on register page |

# Function

## Reader/Author Module

### Post

#### Class diagram

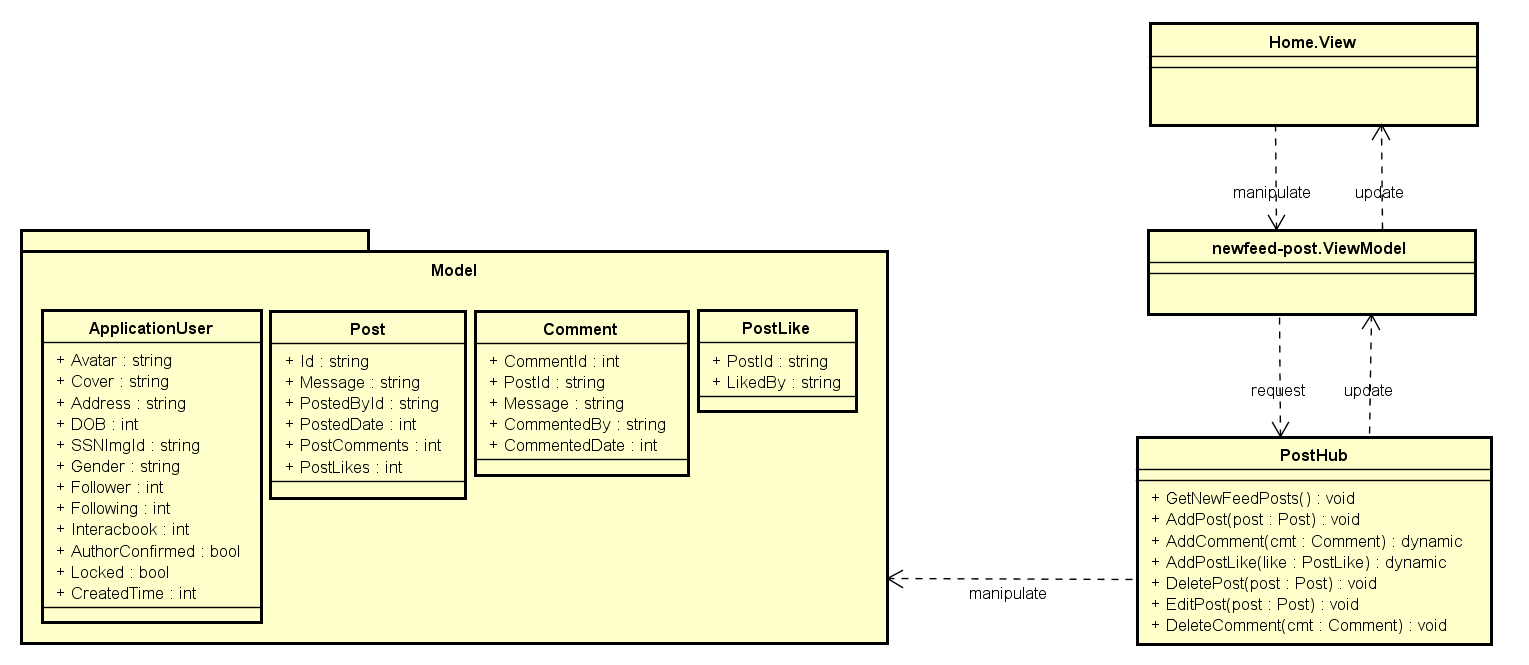


Figure 3‑1: Post Class Design

#### Sequence diagram

##### Create a post

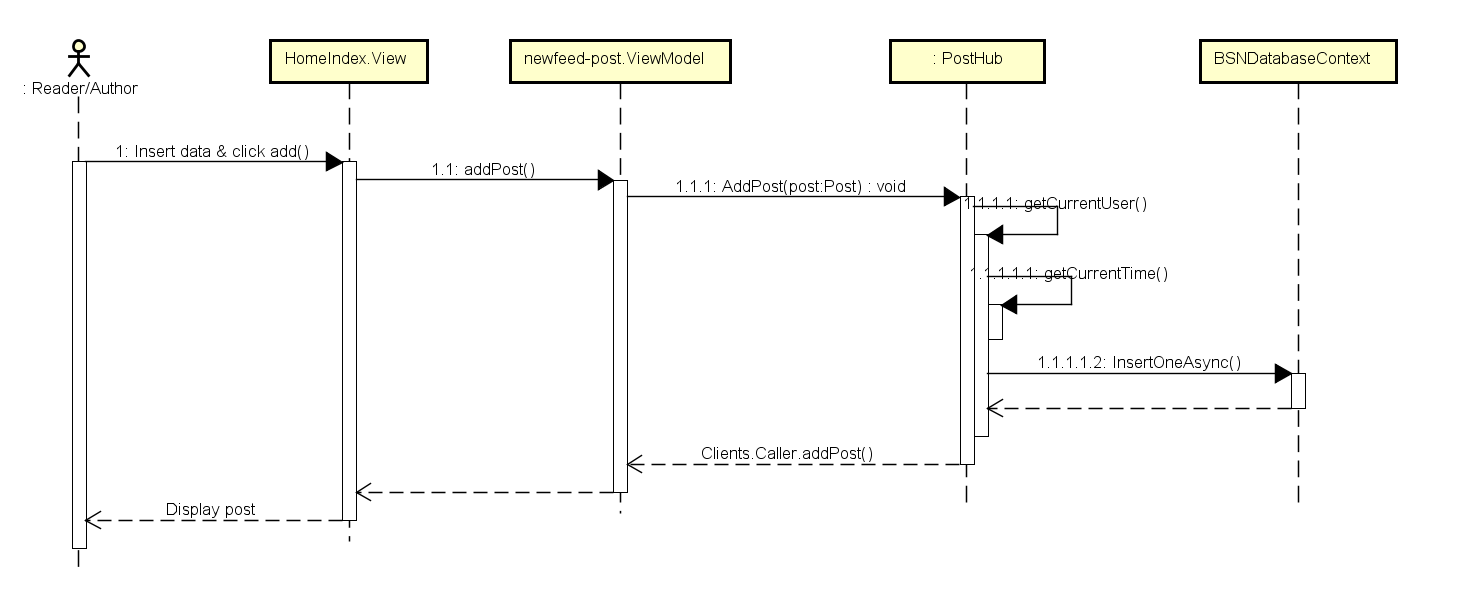


Figure 3‑2: Create Post Sequence Diagram

##### Edit a post

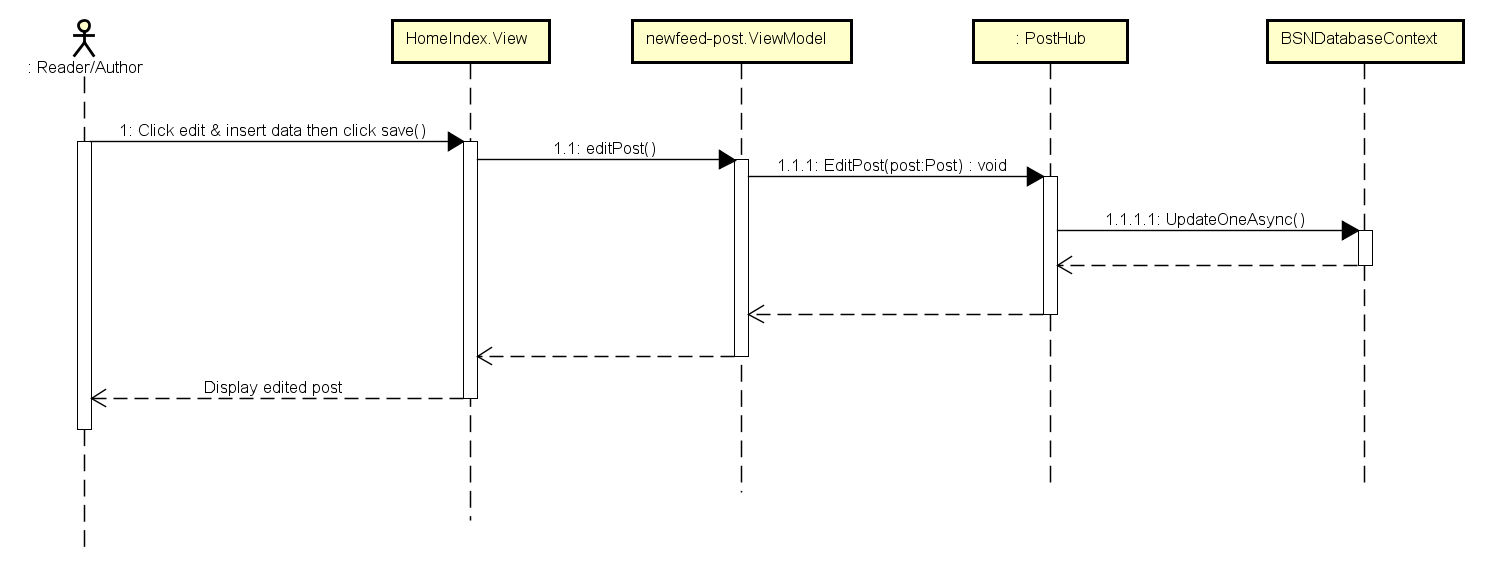


Figure 3‑3: Edit Post Sequence Diagram

##### Delete a post

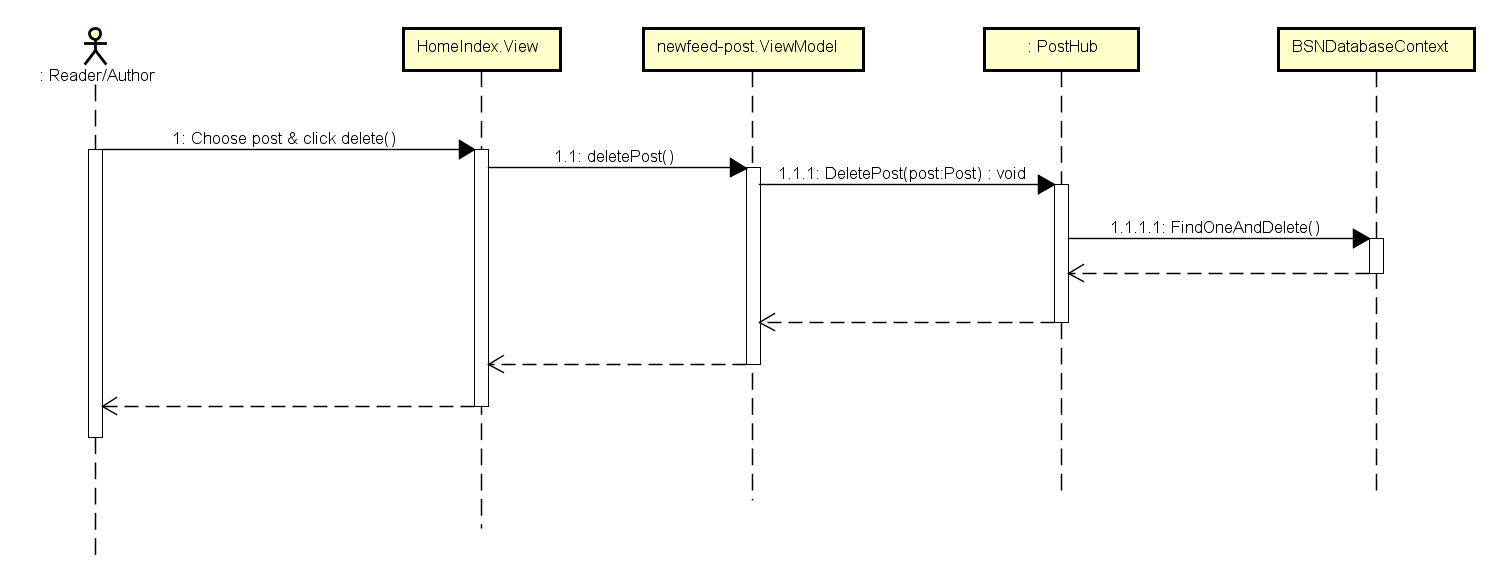


Figure 3‑4: Delete Post Sequence Diagram

##### Like a post

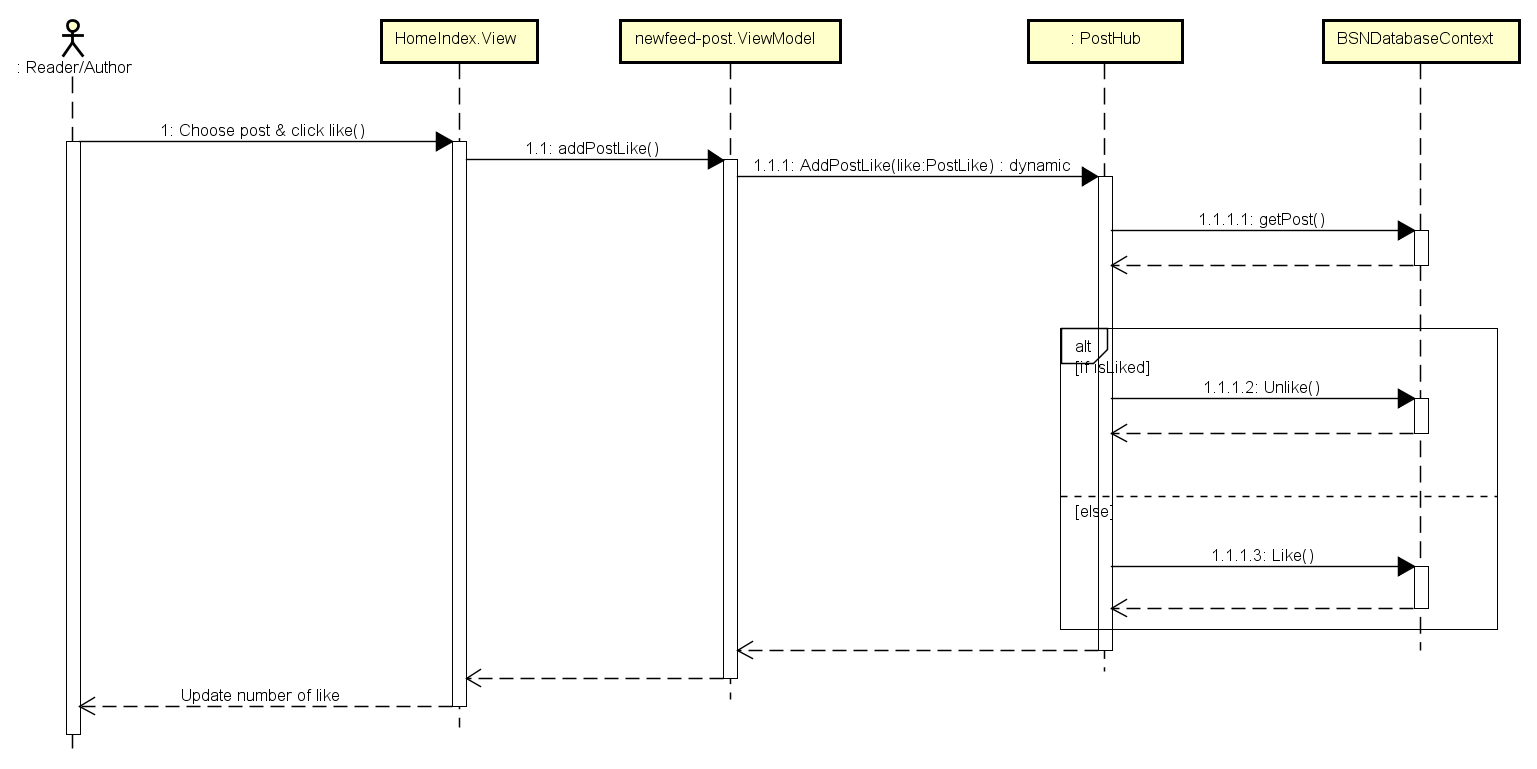


Figure 3‑5: Like Post Sequence Diagram

##### Unlike a post

Refer figure 3.2.2.2.4

##### Comment on a post

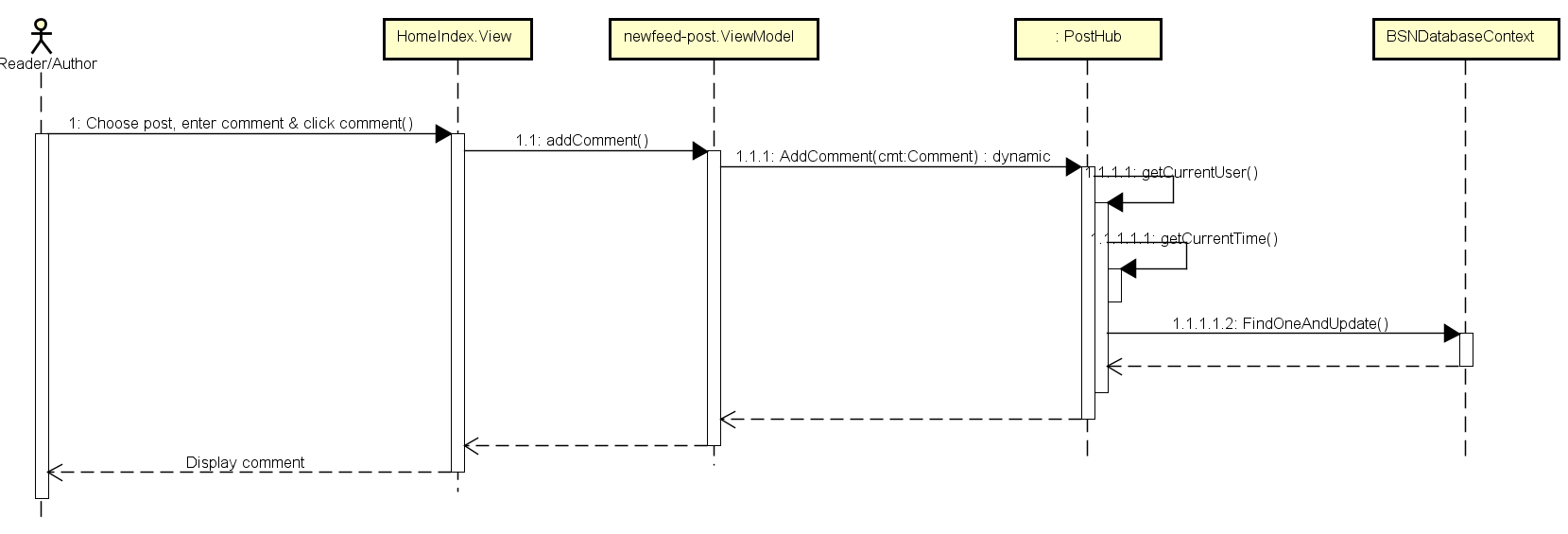


Figure 3‑6: Comment on Post Sequence Diagram

##### Delete a comment

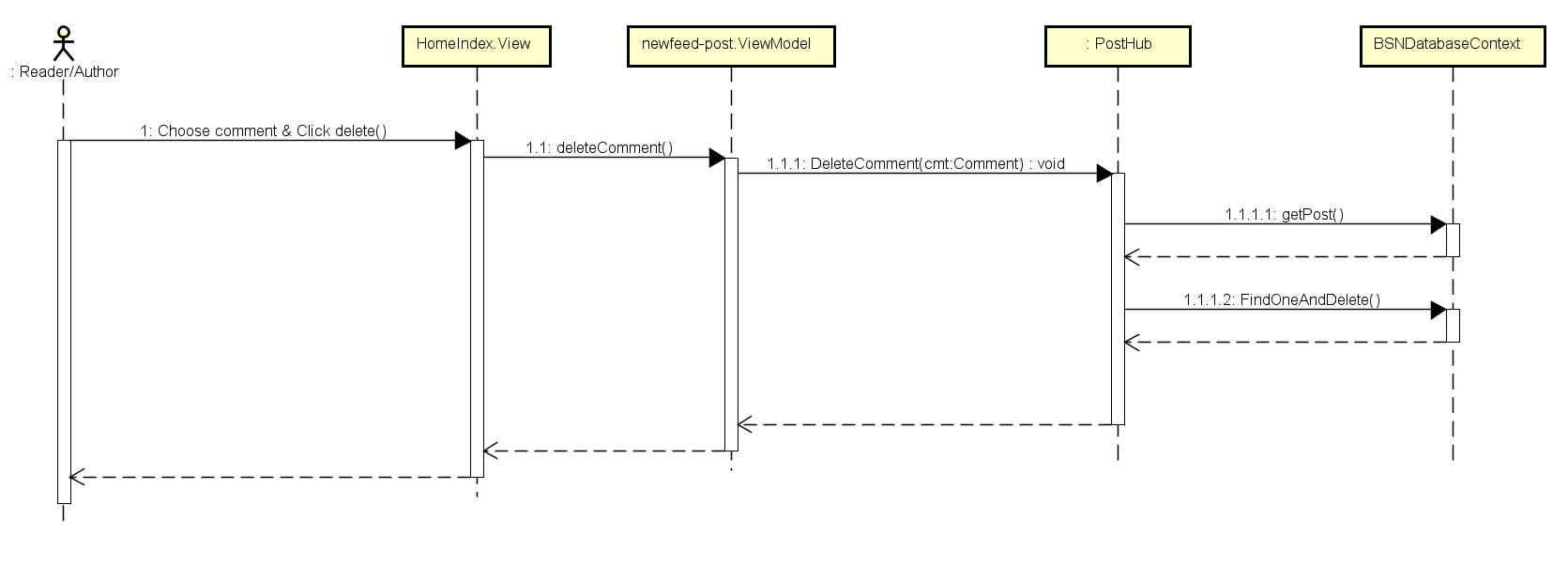


Figure 3‑7: Delete Comment Sequence Diagram

### Books Management

#### Class diagram

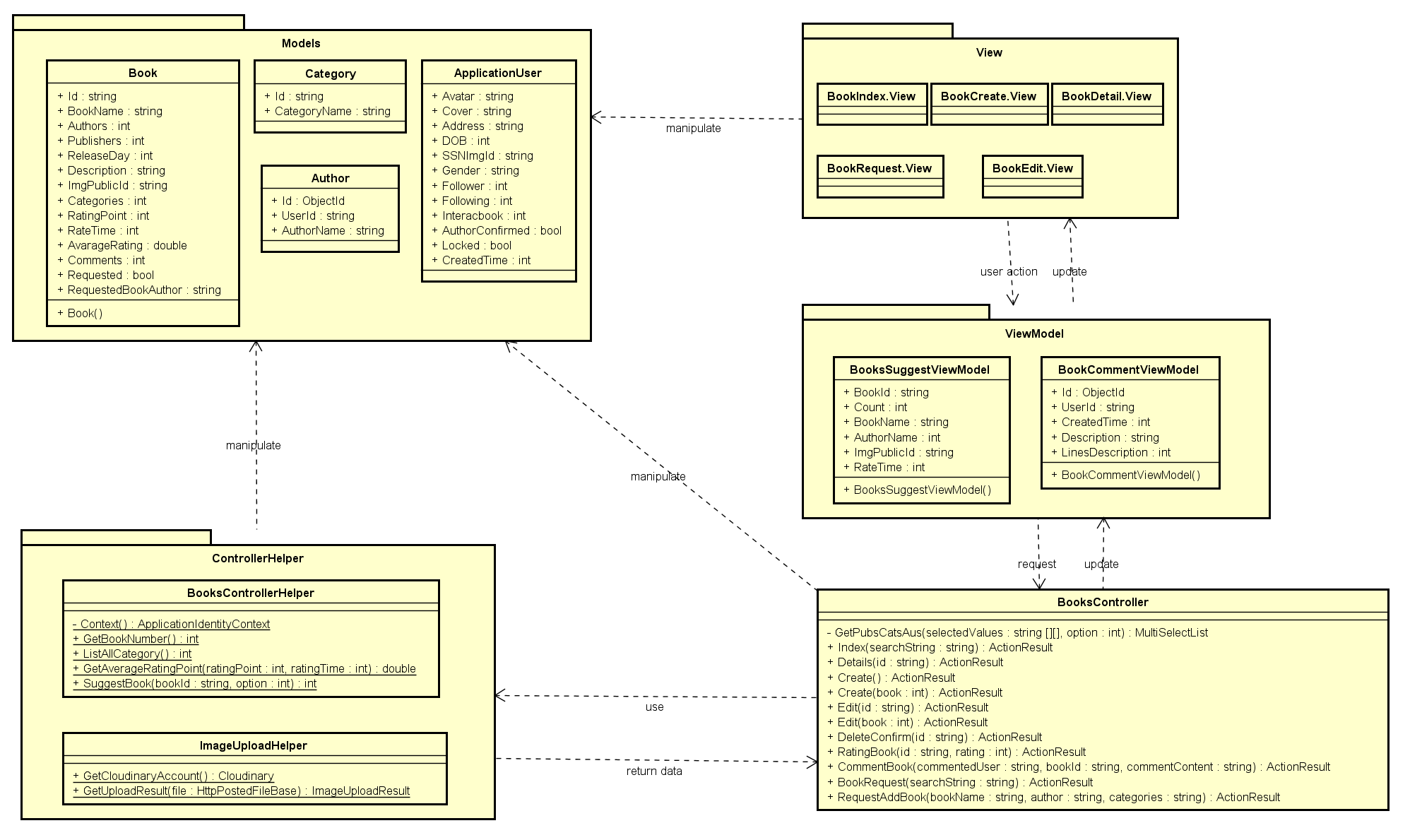


Figure 3‑8: Book Management Class Diagram

#### Sequence diagram

##### Add a new book

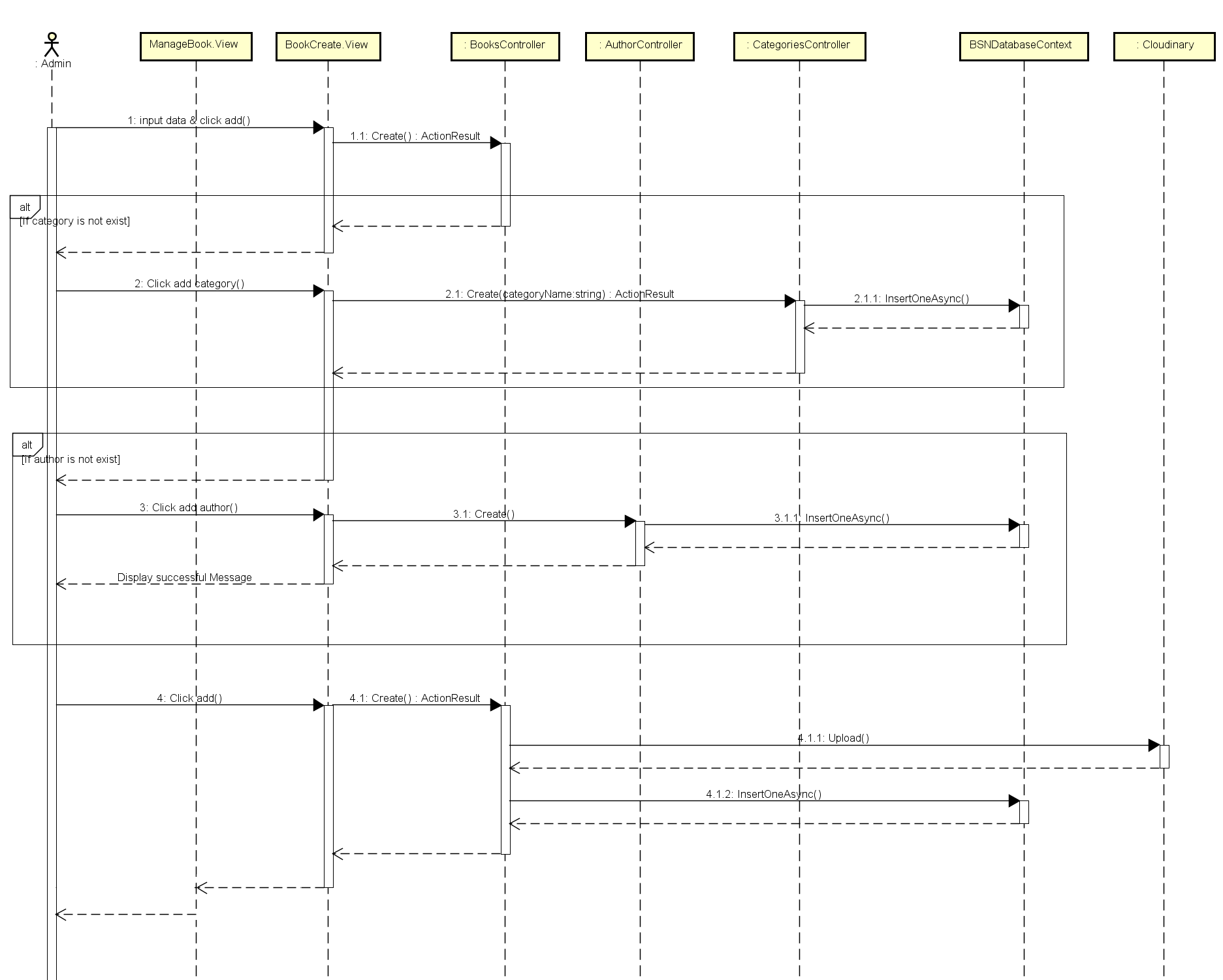


Figure 3‑9: Add Book Sequence Diagram

##### Add a new Category

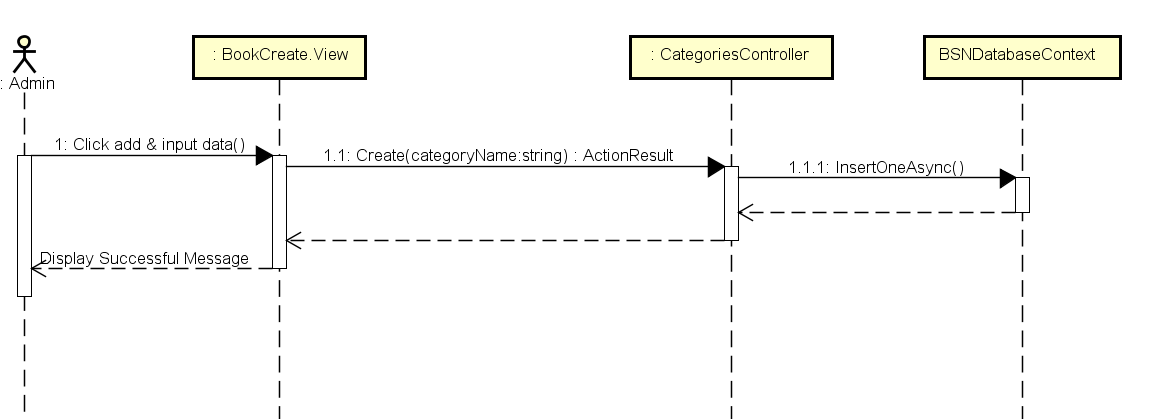


Figure 3‑10: Add Category Sequence Diagram

##### Add an author

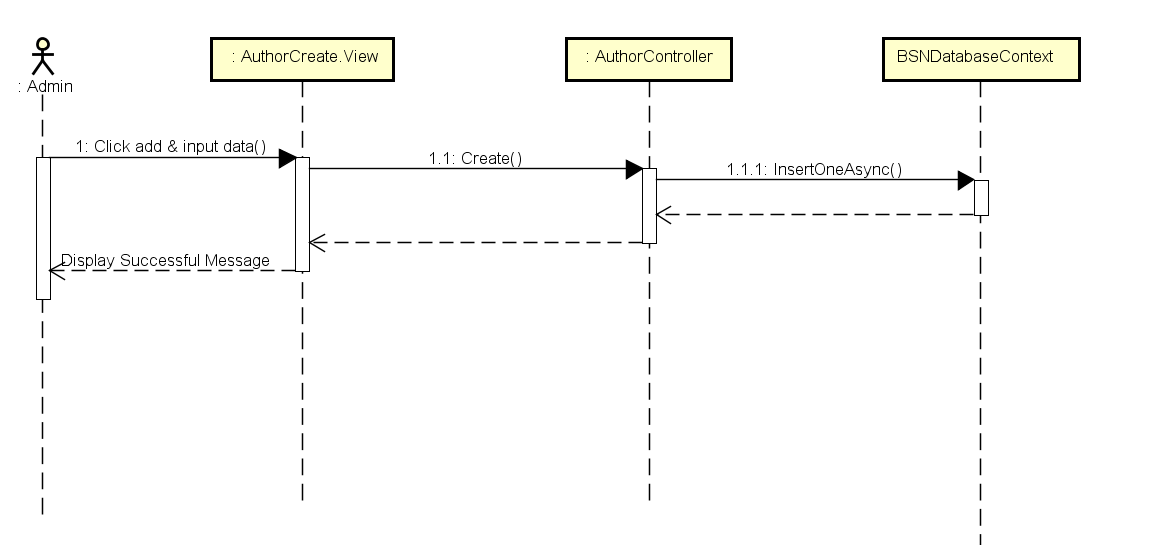


Figure 3‑11: Add Author Sequence Diagram

##### Edit book's information

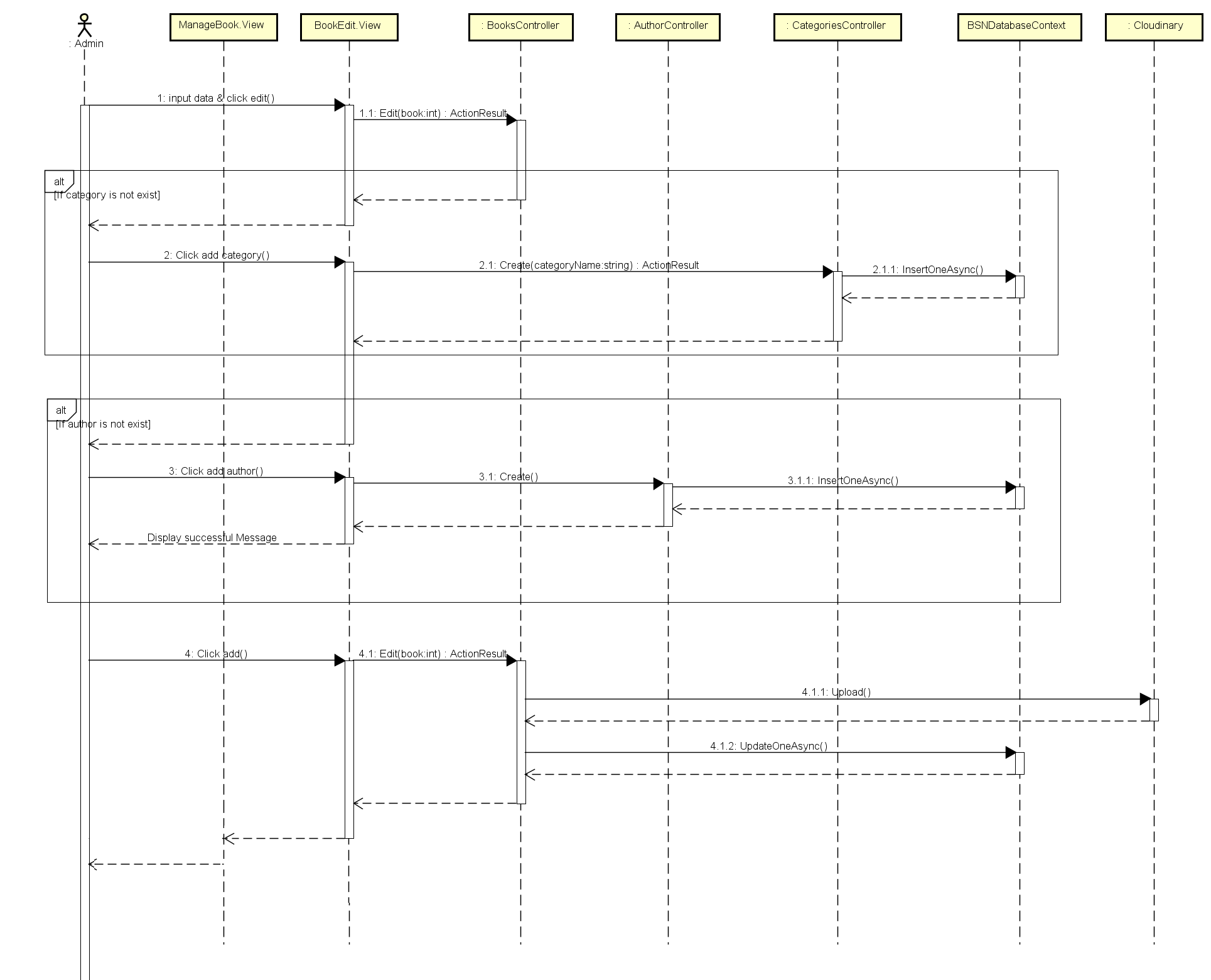


Figure 3‑12: Edit Book Sequence Diagram

##### Accept add-book request

##### Decline add-book request

### Sliders Management

#### Class diagram

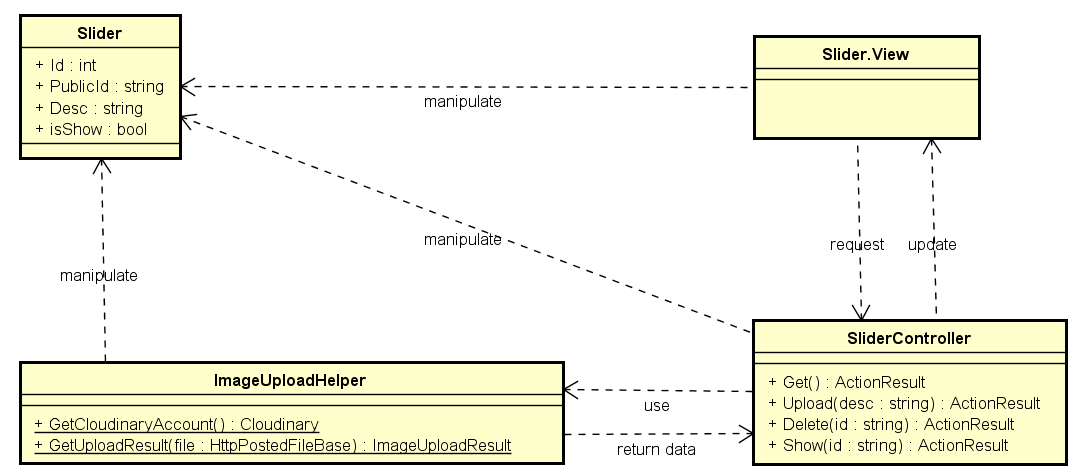


Figure 3‑13: Slider Management Class Diagram

#### Sequence diagram

##### Add a slider

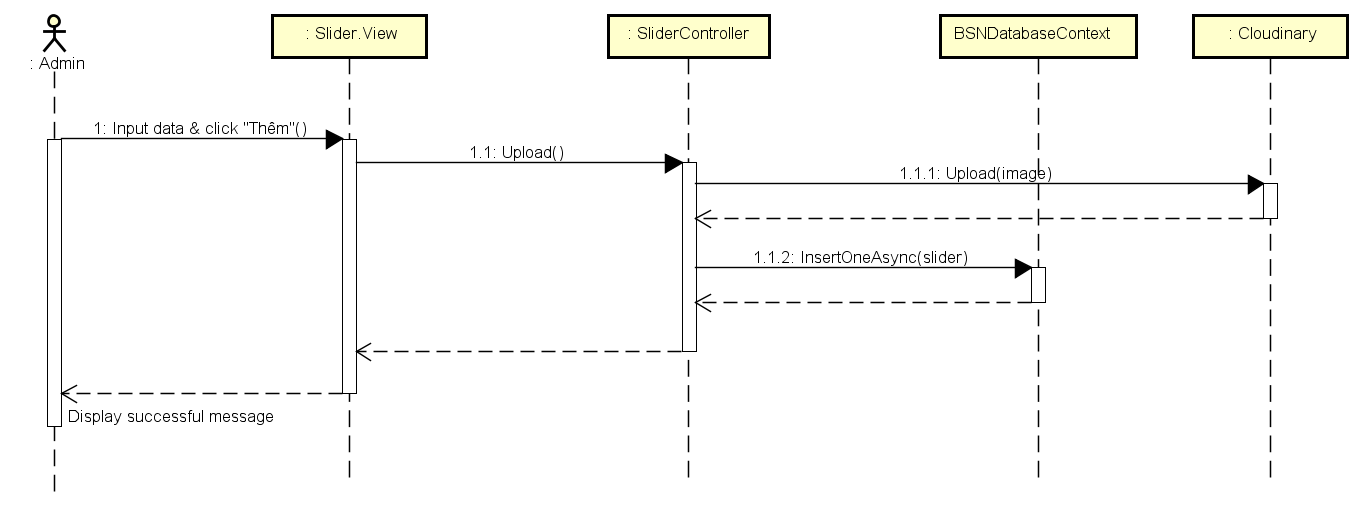


Figure 3‑14: Add Slider Sequence Diagram

##### Delete a slider

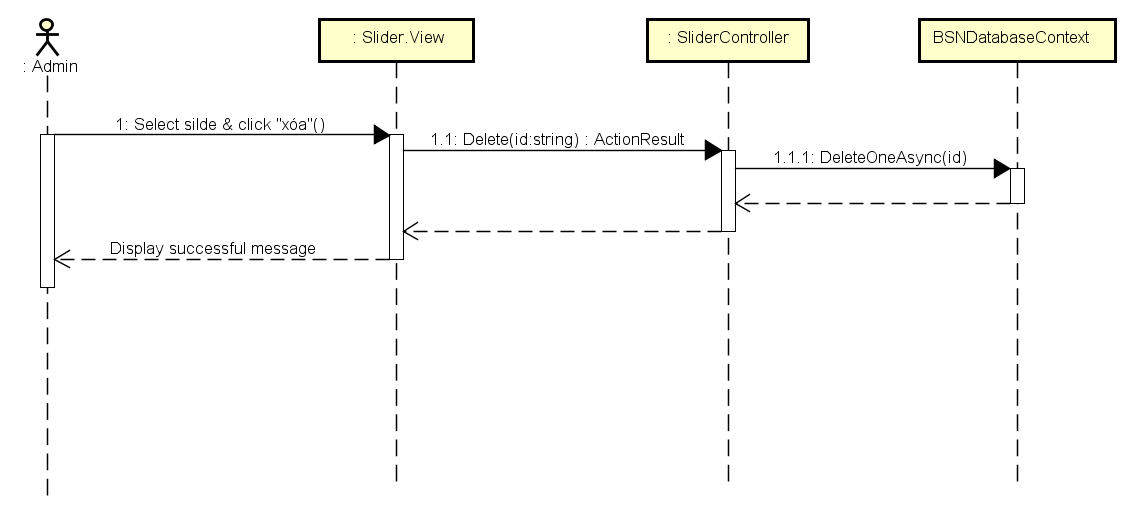


Figure 3‑15: Delete Slider Sequence Diagram

##### Active/de-active a slider

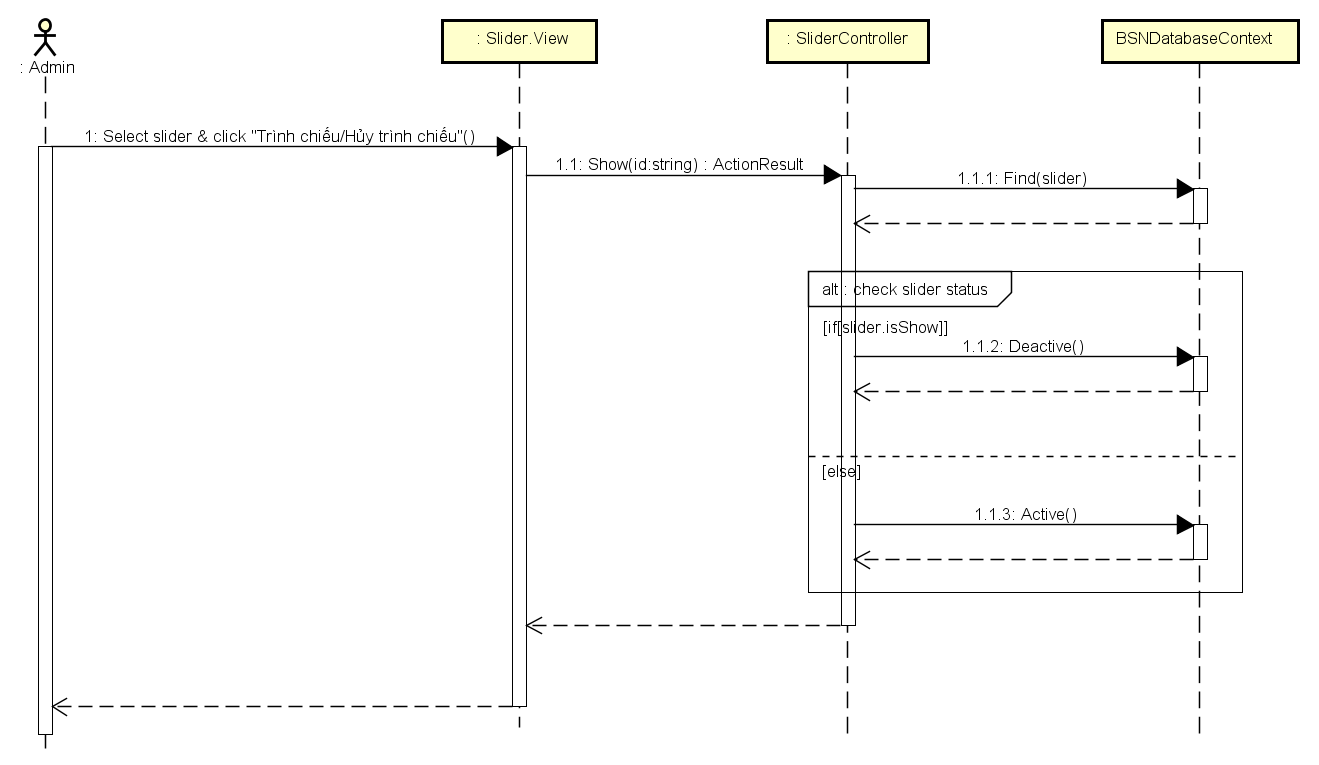


Figure 3‑16: Active/De-active Slider Sequence Diagram

##### Show sliders

### Publishers Management

#### Class diagram

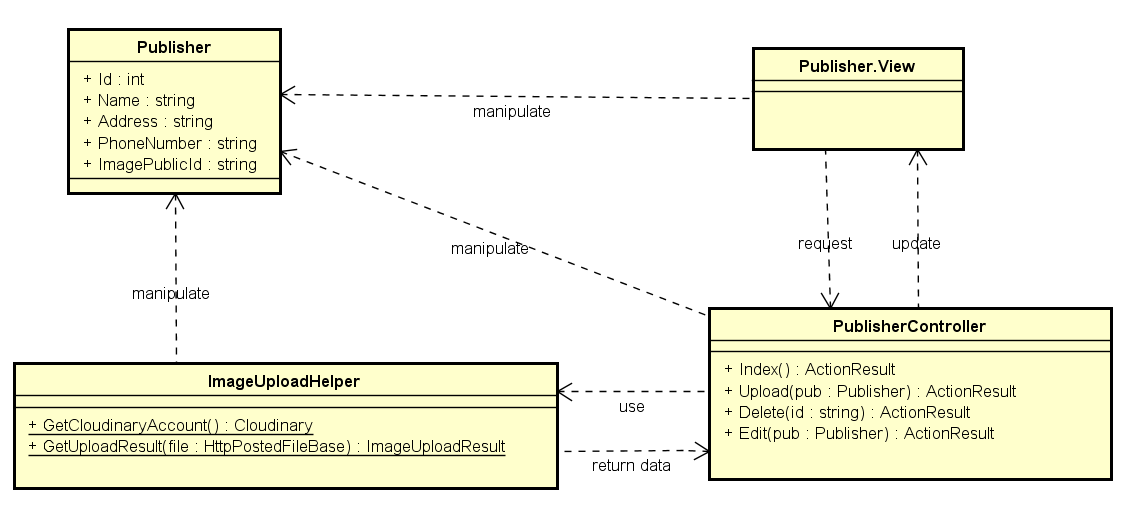


Figure 3‑17: Publisher Management

#### Sequence diagram

##### Add a new publisher

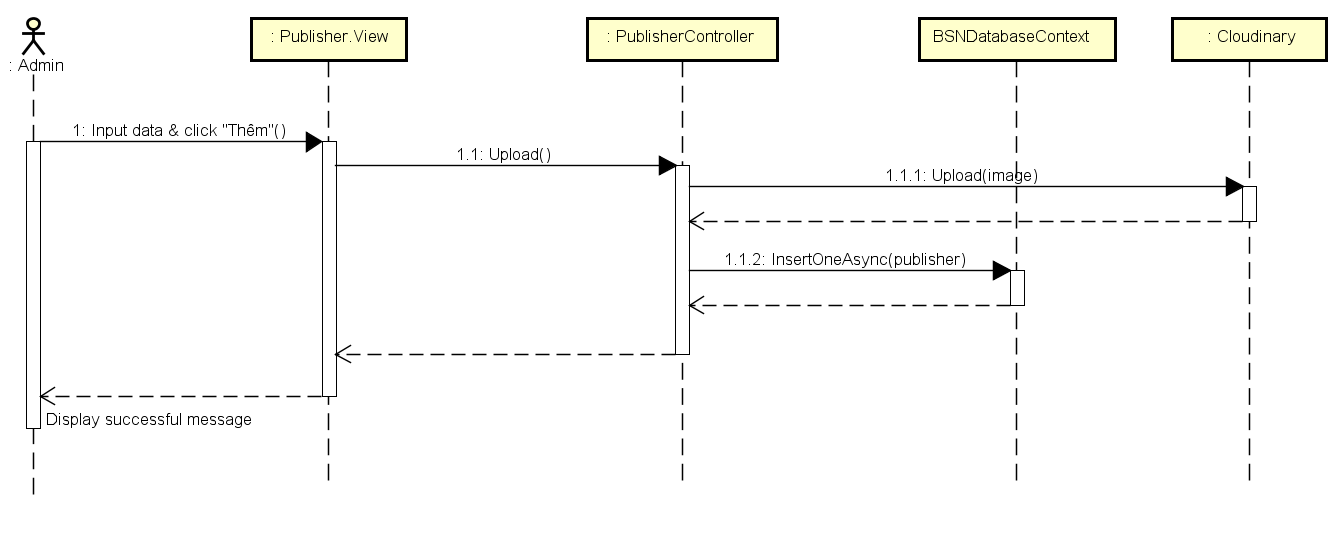


Figure 3‑18: Add publisher Sequence Diagram

##### Edit publisher' information

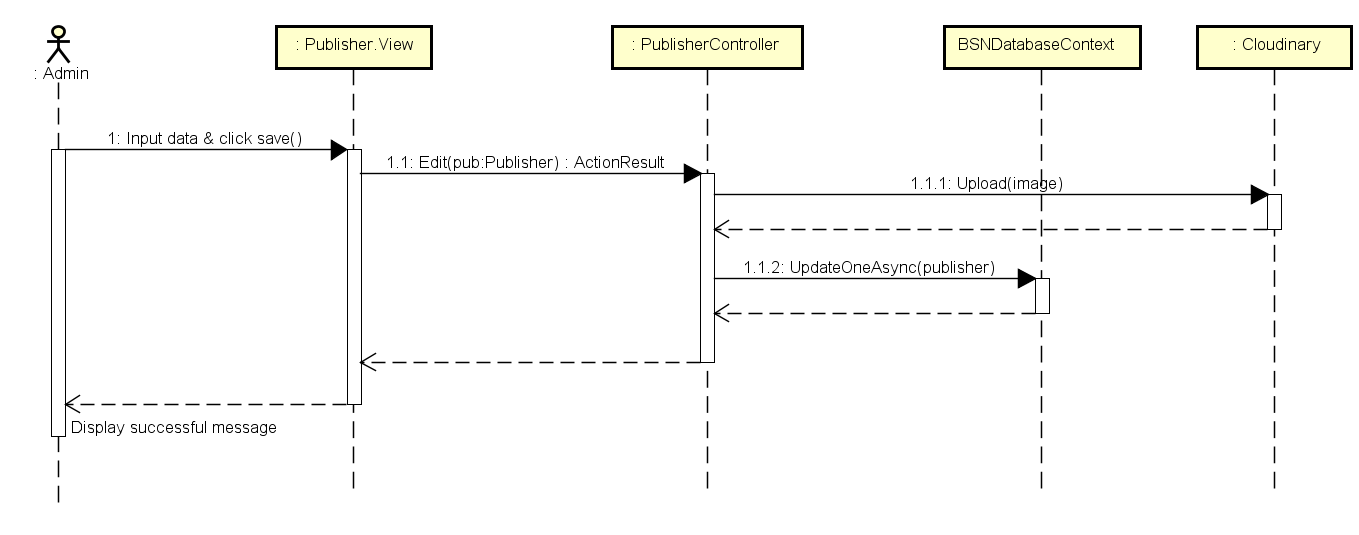


Figure 3‑19: Edit Publisher Sequence Diagram

##### Delete a publisher

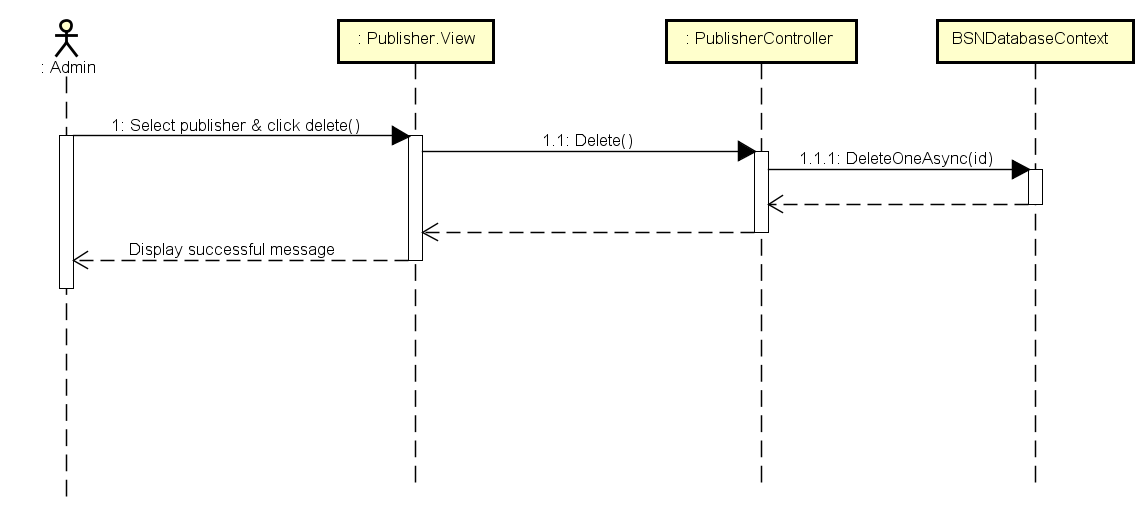


Figure 3‑20: Delete Publisher Sequence Diagram

### Statistics