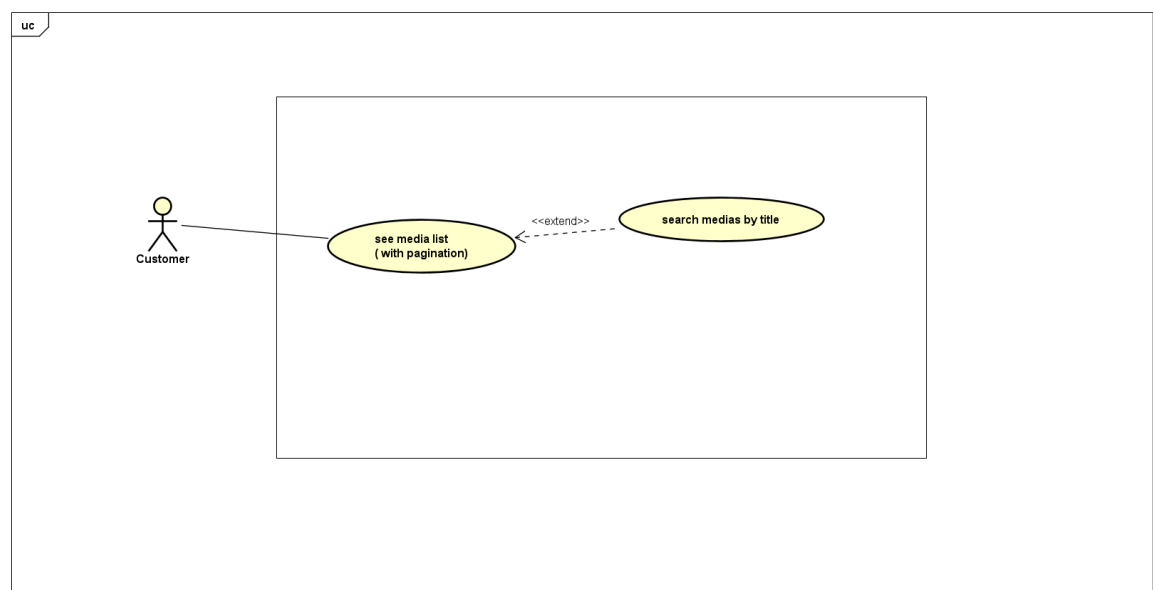
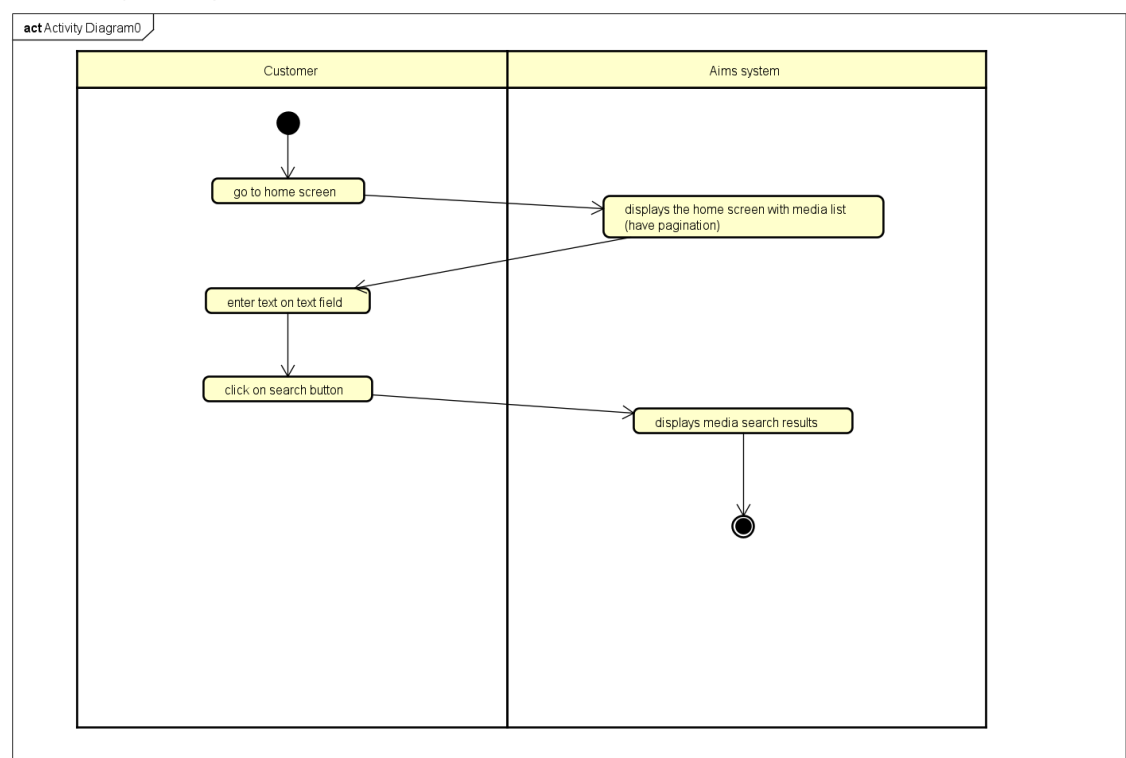


Report

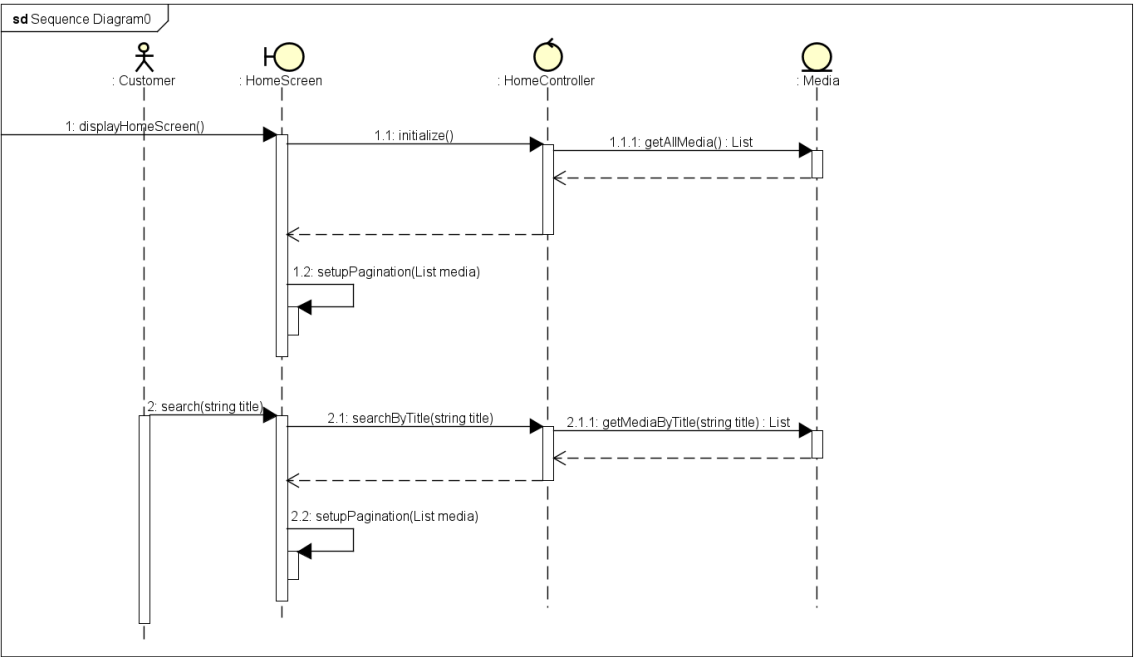
Usecase Diagram



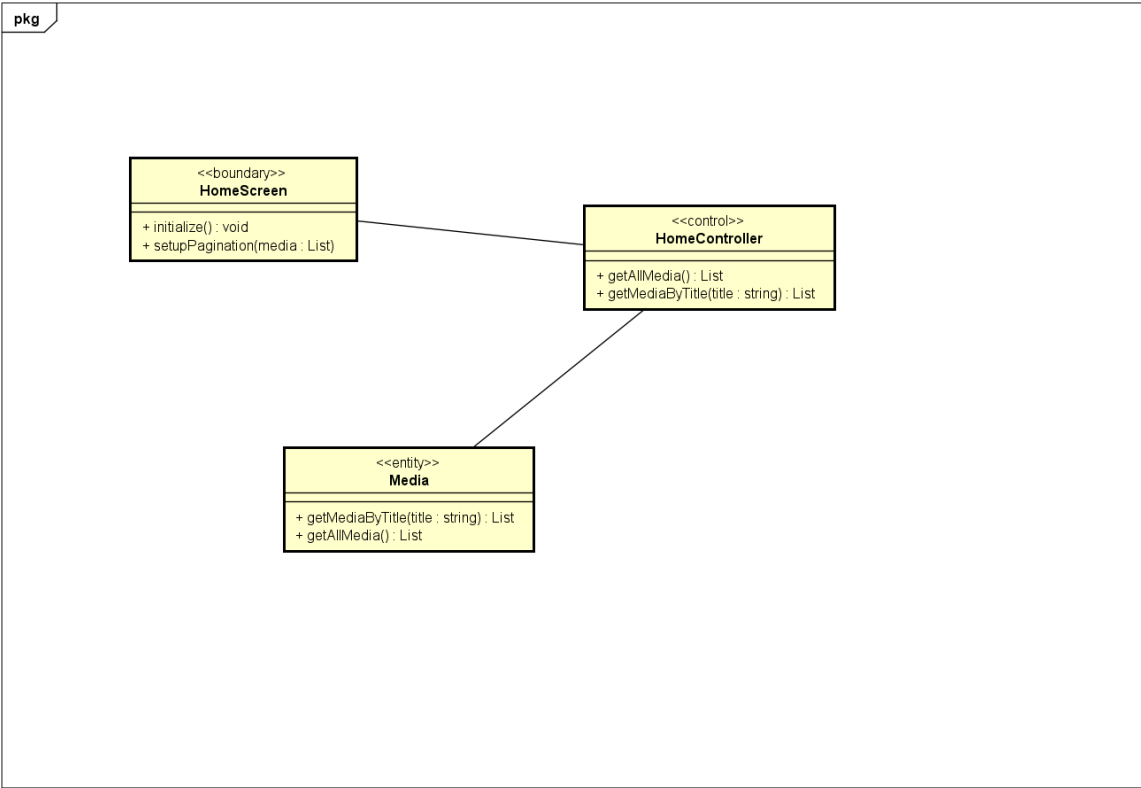
Activity Diagram



Sequence Diagram



Class Diagram



Specification of Usecase UC004 - “See Media List”

1. Use case code

UC004

2. Brief Description

This use case describes the interaction between customer and AIMS system when customer go to home screen

3. Actors

Customer

4. Preconditions

Customer is at the home page

5. Basic Flow of Events

Step 1. Go to home screen

Step 2. Select pages in pagination

Specification of Use case UC005 – “Search Medias by title”

1. Use case code

UC005

2. Brief Description

This use case describes the interaction between customer and AIMS system when customer want to search medias by title

3. Actors

Customer

4. Preconditions

Customer is at the home page

5. Basic Flow of Events

Step 1. Go to home screen

Step 2. Enter text in text field input

Step 3. Click on search button

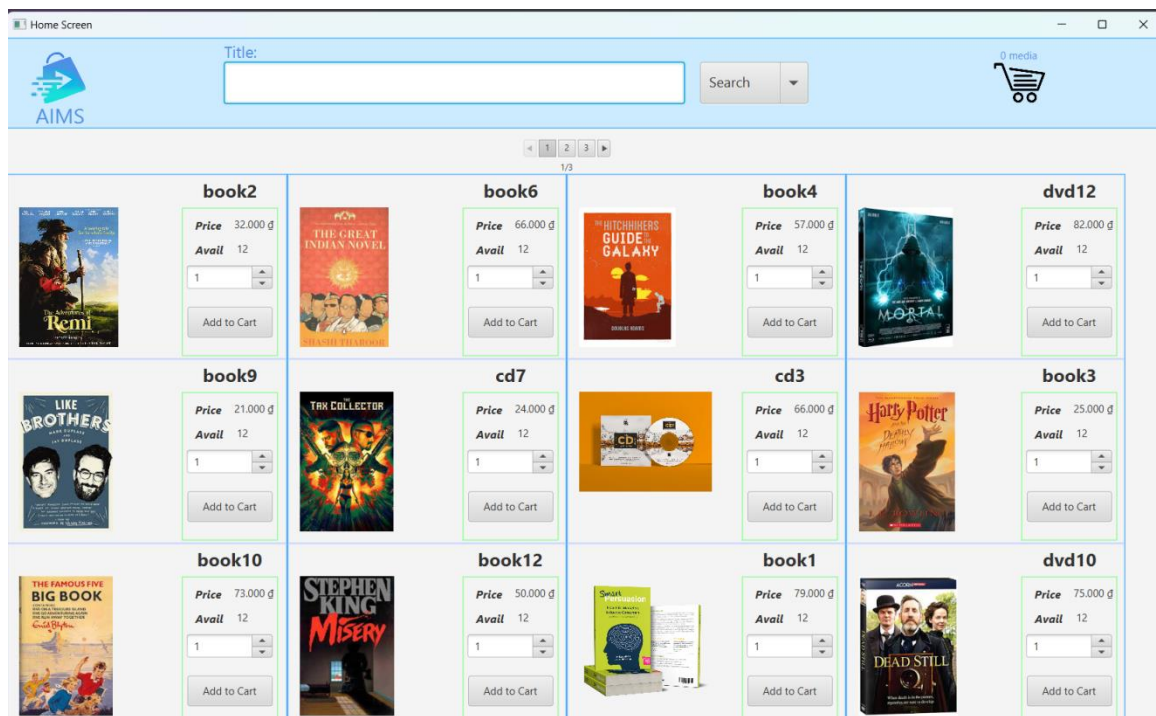
6. Alternative flows

7. Input data

Table A-Input data of text field input

N o	Data fields	Descriptio n	Mandato ry	Valid condition	Example
1	title		No		book

GUI



Design Principles

Low coupling

Class Media

```
public List getMediaByTitle(String title) throws SQLException {
    List mediaList = new ArrayList<>();
    String sql = "SELECT * FROM Media WHERE title LIKE ?";
    try (PreparedStatement preparedStatement = AIMSDB.getConnection().prepareStatement(sql)) {
        preparedStatement.setString(1, "%" + title + "%");
        ResultSet resultSet = preparedStatement.executeQuery();
        while (resultSet.next()) {
            Media media = new Media()
                .setId(resultSet.getInt(columnLabel: "id"))
                .setTitle(resultSet.getString(columnLabel: "title"))
                .setQuantity(resultSet.getInt(columnLabel: "quantity"))
                .setCategory(resultSet.getString(columnLabel: "category"))
                .setMediaURL(resultSet.getString(columnLabel: "imageUrl"))
                .setPrice(resultSet.getInt(columnLabel: "price"))
                .setType(resultSet.getString(columnLabel: "type"));
            mediaList.add(media);
        }
    }
    return mediaList;
}
```

Class Media không trực tiếp tạo ra đối tượng AIMSDB, mà thay vào đó, nó sử dụng một phương thức tĩnh từ AIMSDB (AIMSDB.getConnection()) để nhận được kết nối.

HomeController

```
public List getMediaByTitle(String title) throws SQLException {  
    return new Media().getMediaByTitle(title);  
}
```

HomeController không trực tiếp tạo ra đối tượng Media, mà thay vào đó, nó tận dụng các phương thức từ Media để lấy thông tin.

High cohesion

Trong HomeScreenHandler

Các hàm liên kết chặt chẽ với nhau

```
private void searchByTitle(String title) throws IOException {  
    try {  
        List searchResults = getBController().getMediaByTitle(title);  
  
        //ArrayList modifiedSearchResult = new ArrayList(searchResults);  
  
        List<MediaHandler> Resultst = new ArrayList<>();  
        for (Object object : searchResults) {  
            Media media = (Media) object;  
            MediaHandler m1 = new MediaHandler(Configs.HOME_MEDIA_PATH, media, home: this);  
            Resultst.add(m1);  
        }  
  
        addMediaHome(Resultst);  
  
        setupPagination(Resultst);  
    } catch (SQLException e) {  
        LOGGER.info(msg: "Error occurred during search: " + e.getMessage());  
        e.printStackTrace();  
    }  
}
```

```

public void addMediaHome(List items) {
    // Reset the content of the hboxMedia
    vboxMedia1.getChildren().clear();
    vboxMedia2.getChildren().clear();
    vboxMedia3.getChildren().clear();
    vboxMedia4.getChildren().clear();

    // Add media items to the hboxMedia
    ArrayList mediaItems = new ArrayList(items);
    while (!mediaItems.isEmpty()) {
        for (int j = 0; j < 3 && !mediaItems.isEmpty(); j++) {
            if (!mediaItems.isEmpty()) {
                MediaHandler media = (MediaHandler) mediaItems.get(0);
                vboxMedia1.getChildren().add(media.getContent());
                mediaItems.remove(media);
            }
        }
        for (int j = 0; j < 3 && !mediaItems.isEmpty(); j++) {
            if (!mediaItems.isEmpty()) {
                MediaHandler media = (MediaHandler) mediaItems.get(0);
                vboxMedia2.getChildren().add(media.getContent());
                mediaItems.remove(media);
            }
        }
        for (int j = 0; j < 3 && !mediaItems.isEmpty(); j++) {
            if (!mediaItems.isEmpty()) {

```

```

private void setupPagination(List items) {
    int itemsPerPage = 12;
    int numPages = (int) Math.ceil(((double) items.size() / itemsPerPage));
    paging.setPageCount(numPages);
    paging.setPageFactory((Integer pageIndex) -> createPage(pageIndex, items));
}

private VBox createPage(int pageIndex, List items) {
    VBox pageBox = new VBox();

    int itemsPerPage = 12;
    int startIndex = pageIndex * itemsPerPage;
    int endIndex = Math.min(startIndex + itemsPerPage, items.size());

    addMediaHome(items.subList(startIndex, endIndex));

    return pageBox;
}

```

Design Pattern

MVC pattern