

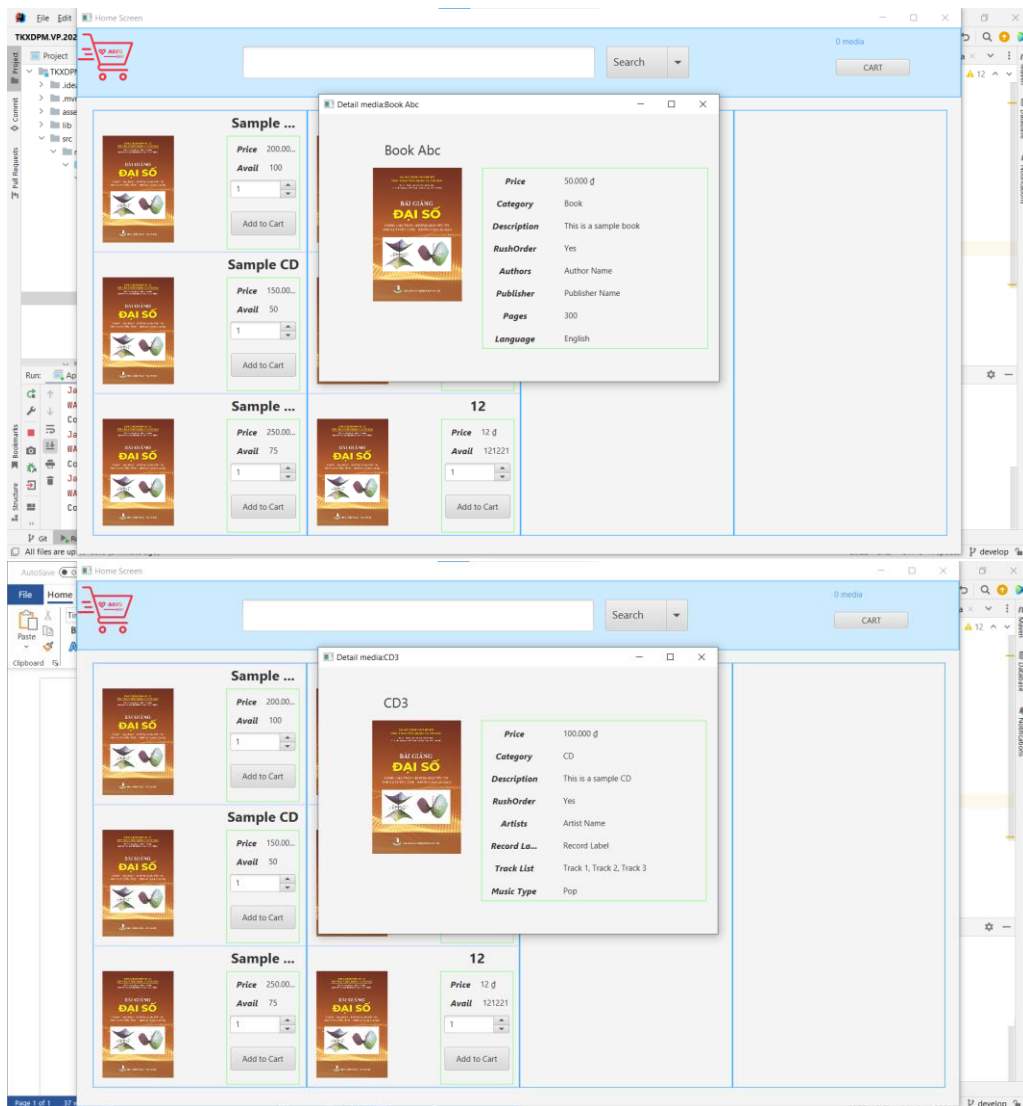
Họ và tên: Nguyễn Đình Dương

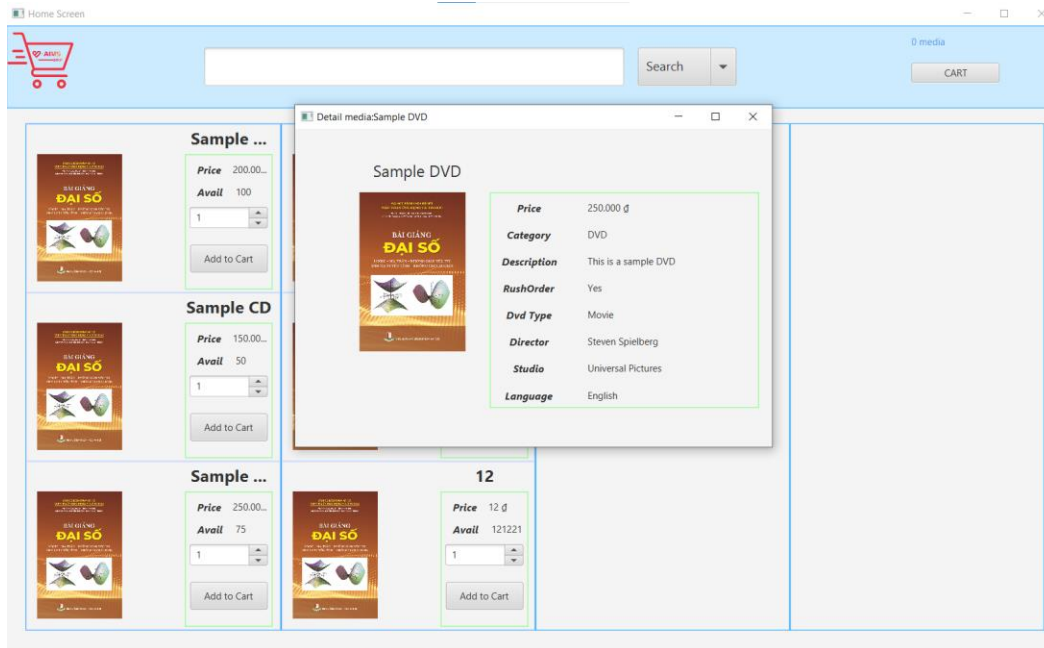
MSSV: 20205201

Nội dung Update bài tập lớn

1. Hiện thị chi tiết sản phẩm

- Để xem chi tiết 1 sản phẩm: ấn vào ảnh sản phẩm đó. Sau đó sẽ có 1 màn hình hiển thị chi tiết sản phẩm. Với mỗi thể loại khác nhau, người dùng có thể thấy các thông tin khác với mỗi loại sản phẩm như bên dưới
- Demo:





- Code:

```
mediaImage.setOnMouseClicked(event -> {
    try {
        showMediaDetails(media);
    } catch (IOException e) {
        throw new RuntimeException(e);
    }
});

private void showMediaDetails(Media media) throws IOException {
    MediaDetailHandler mediaDetailHandler = new MediaDetailHandler(this.stage, Configs.HOME_MEDIA_DETAIL_PATH, media);
    mediaDetailHandler.setScreenTitle("Detail media:" + media.getTitle());
    mediaDetailHandler.show();
}
```

1 usage DinhDuong

```
public void setMediaDetails(Media media) throws SQLException {
    detailedMediaTitle.setText(media.getTitle());
    detailedMediaPrice.setText(Utils.getCurrencyFormat(media.getPrice()));
    detailedMediaDescription.setText(media.getDescription());
    detailedMediaCategory.setText(media.getCategory());
    String rushOrder;
    if (media.getRushOrderSupport() == true) {
        rushOrder = "Yes";
    } else {
        rushOrder = "No";
    }
    detailedMediaRushOrder.setText(rushOrder);
    if (media.getCategory().equals("Book")) {
        info1.setText("Authors");
        info2.setText("Publisher");
        info3.setText("Pages");
        info4.setText("Language");
        Book media1 = (Book) HomeController.getBookById(media.getId());
        detailedInfo1.setText(media1.getAuthors());
        detailedInfo2.setText(media1.getPublisher());
        detailedInfo3.setText(String.valueOf(media1.getPages()));
        detailedInfo4.setText(media1.getLanguage());
    }

    if (media.getCategory().equals("DVD")) {
        info1.setText("Dvd Type");
        info2.setText("Director");
        info3.setText("Studio");
        info4.setText("Language");
    }
}
```

Ở đây các info1-4 sẽ chứa thông tin khác khi category media là Book hoặc DVD hoặc CD hoặc LP

Tương ứng thông tin sẽ được set lần lượt từ detailInfo1-4

1 usage DinhDuong

```
public static Media getBookById(int id) throws SQLException{  
    return new BookDAO().getMediaById(id);  
}
```

1 usage DinhDuong

```
public static Media getDvdById(int id) throws SQLException{  
    return new DVDDAO().getMediaById(id);  
}
```

1 usage DinhDuong

```
public static Media getCdById(int id) throws SQLException{  
    return new CDDAO().getMediaById(id);  
}
```

Để lấy thông tin khác trong các media em dùng downcasting từ media xuống các sản phẩm con

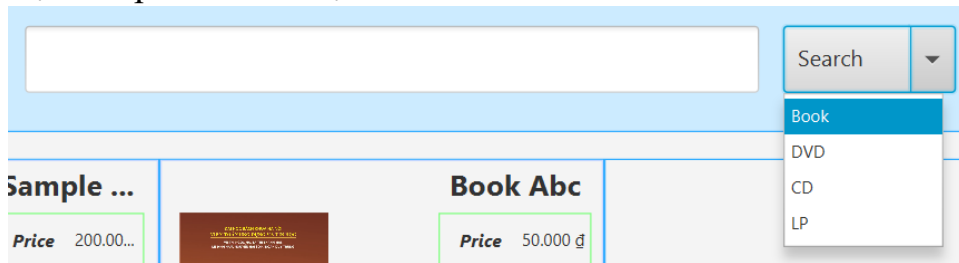
```
Book media1 = (Book) HomeController.getBookById(media.getId());  
detailedInfo1.setText(media1.getAuthors());  
detailedInfo2.setText(media1.getPublisher());  
detailedInfo3.setText(String.valueOf(media1.getPages()));  
detailedInfo4.setText(media1.getLanguage());
```

```
Dvd media1 = (Dvd) HomeController.getDvdById(media.getId());  
detailedInfo1.setText(media1.getDvdType());  
detailedInfo2.setText(media1.getDirector());  
detailedInfo3.setText(media1.getStudio());  
detailedInfo4.setText(media1.getLanguage());
```

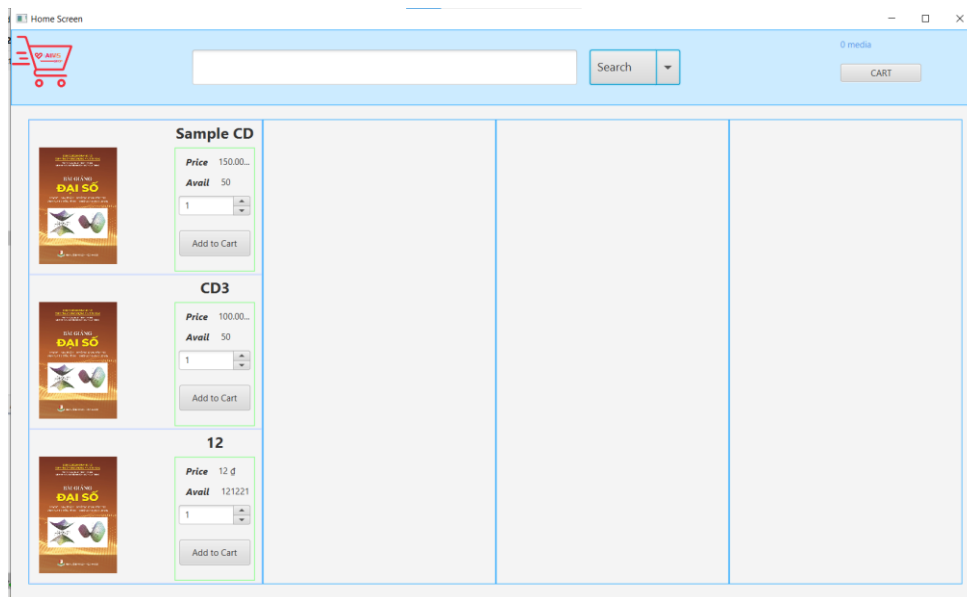
```
CdAndLp media1 = (CdAndLp) HomeController.getCdById(media.getId());  
detailedInfo1.setText(media1.getArtists());  
detailedInfo2.setText(media1.getRecordLabel());  
detailedInfo3.setText(media1.getTrackList());  
detailedInfo4.setText(media1.getMusicType());
```

2. Lọc và tìm kiếm sản phẩm theo tên

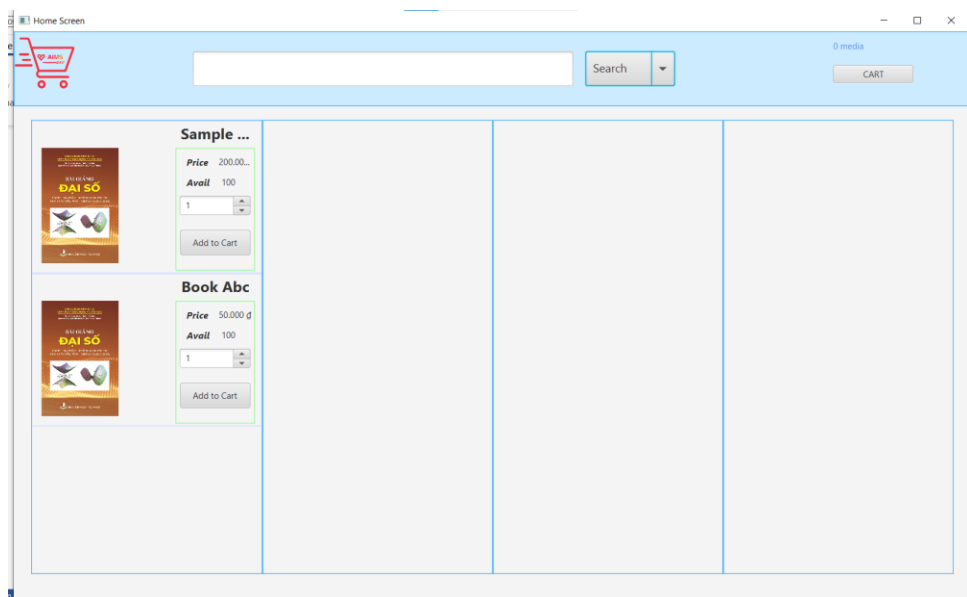
- Lọc sản phẩm theo loại media



Khi ấn CD



Khi ấn Book



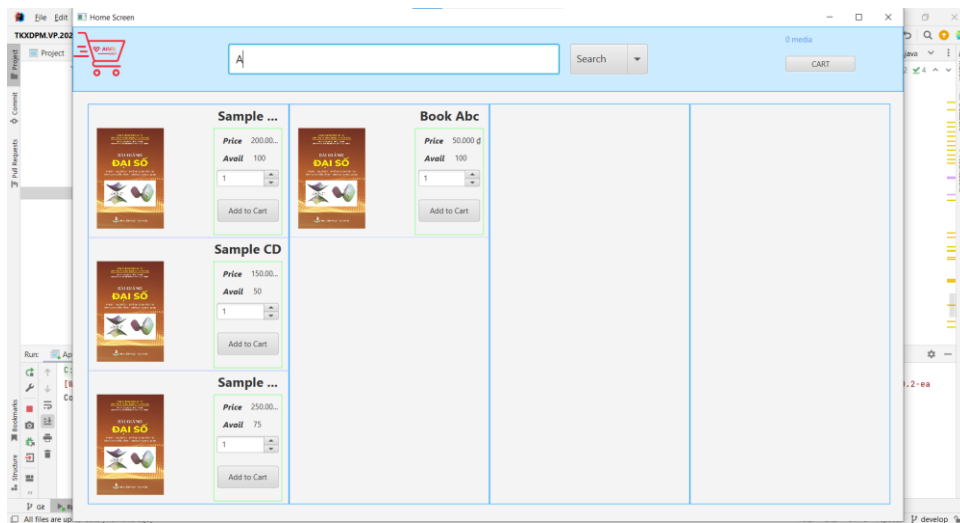
Code :

```
.....
List<MediaHandler> filteredItems = new ArrayList<>();

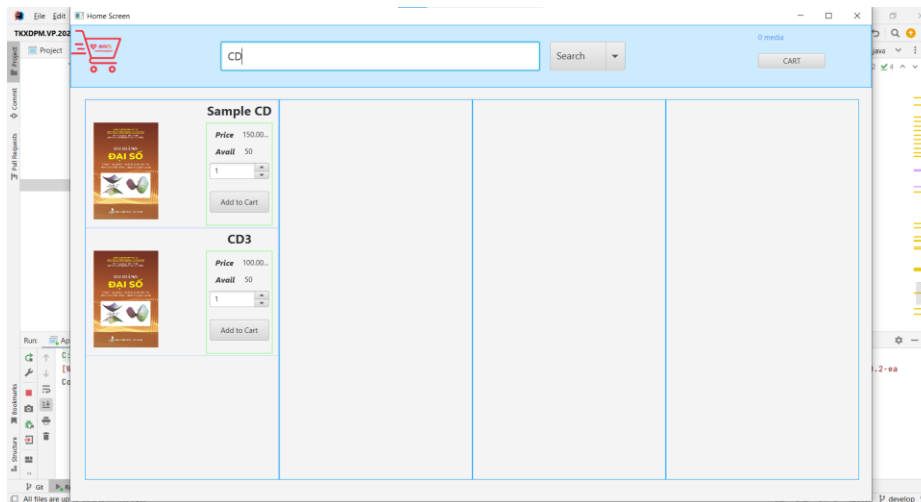
homeItems.forEach(me -> {
    MediaHandler media = (MediaHandler) me;
    if (media.getMedia().getCategory().equalsIgnoreCase(text)) {
        filteredItems.add(media);
    }
});
```

- Tìm kiếm sản phẩm theo tên

Khi nhập vào từ “A”



Khi nhập vào từ “CD”



Code

2 usages

@FXML

```
private TextField searchTextField;
```

1 usage DinhDuong

```
private void handleSearch(KeyEvent event) {  
    String searchTerm = searchTextField.getText().toLowerCase().trim();  
  
    List<MediaHandler> filteredItems = new ArrayList<>();  
  
    homeItems.forEach(me -> {  
        MediaHandler media = (MediaHandler) me;  
        if (media.getMedia().getTitle().toLowerCase().contains(searchTerm)  
            filteredItems.add(media);  
        }  
    });  
    // Update the displayed media based on the filtered items  
    addMediaHome(filteredItems);  
}
```

3. Cập nhật phương thức tính phí

- Hiện tại em chưa xử lý được phần tính toán khi truyền vào khối lượng sản phẩm
- Khi người dùng không chọn giao hàng nhanh

Với hóa đơn trên 100.000đ được tính phí bằng 0

```
public class CalculateShippingFee_v1 implements ShippingFeeCalculator{  
    1 usage DinhDuong *  
    public int calculateShippingFee(OrderService order){  
        Random rand = new Random();  
        int fees;  
        if (order.getAmount() >= 100000){  
            return 0;  
        } else {  
            fees = (int) (((rand.nextFloat() * 10) / 100) * order.getAmount());  
        }  
  
        return fees;  
    }  
}
```

- Khi người dùng chọn giao hàng nhanh với mỗi sản phẩm sẽ cộng thêm thêm 10000đ

```
public int calculateRushOrderShippingFee(OrderMedia orderMedia) {  
    if (orderMedia == null || orderMedia.getQuantity() == 0) {  
        System.out.print("The order must not be empty!");  
        return 0;  
    }  
    int rushOrderFees = orderMedia.getQuantity() * 10000;  
    return rushOrderFees;  
}
```