### VIỆN CÔNG NGHỆTHÔNG TIN VÀ TRUYỀN THÔNG

#### BÁO CÁO BÀI TẬP LỚN

### XÂY DỰNG HỆ THỐNG THUÊ XE

#### CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN

SINH VIÊN THỰC HIỆN

 Họ Tên
 MSSV

 Đỗ Viết Trí
 20173412

 Trần Văn Trí
 20173410

Nguyễn Ngọc Trinh 20173413

Nguyễn Mạnh Trường 20177022

Hà Nội, tháng 12/2020

## Nội dung

- I. Phân công công việc
- II. Một interation điển hình
- III. Biểu đồ lớp thiết kế
- IV. Design Consideration

# PHÂN CÔNG CÔNG VIỆC

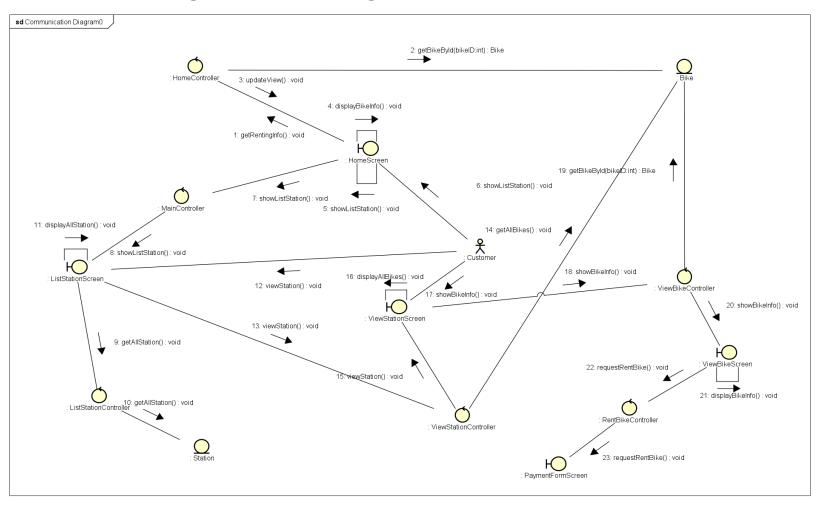
- I: Phân công nhiệm vụ và đánh giá phần trăm đóng góp
- 1. Phân công nhiệm vụ
- + Đỗ Viết Trí: Phân tích thiết kế, code và test cho usecase trả xe, tổng hợp readme
- + Trần Văn Trí: Phân tích thiết kế, code và test cho usecase thuê xe, Tìm kiếm xe
- + Nguyễn Ngọc Trinh: Phân tích thiết kế, code và test cho usecase xem thông tin bãi xe
- + Nguyễn Mạnh Trường: Phân tích thiết kế, code và test cho usecase xem thông tin chi tiết xe

## PHÂN CÔNG CÔNG VIỆC

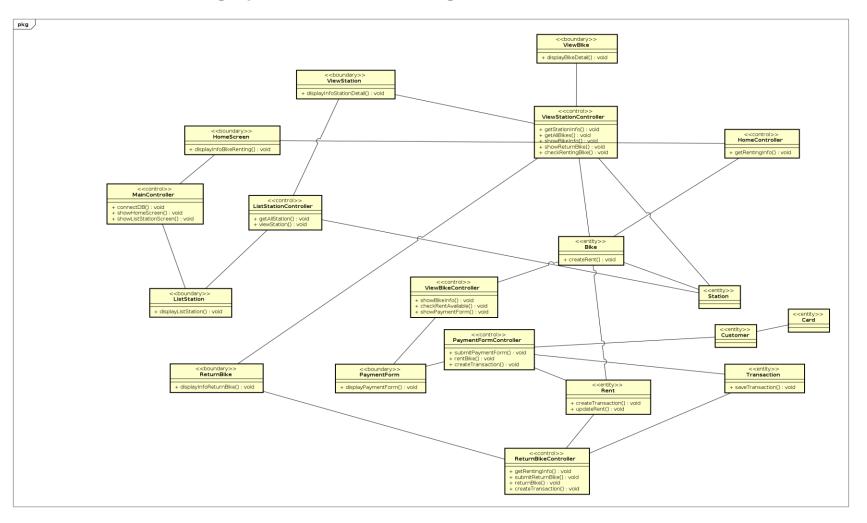
- I: Phân công nhiệm vụ và đánh giá phần trăm đóng góp
- 1. Đánh giá đóng góp
- + Nhóm không có nhân vật nổi trội, tuy nhiên làm việc rất tích cực
- + Đánh giá đóng góp:
  - + Đỗ Viết Trí: 25%
  - + Trần Văn Trí: 25%
  - + Nguyễn Ngọc Trinh: 25%
  - + Nguyễn Mạnh Trường: 25%

## Một interaction điển hình

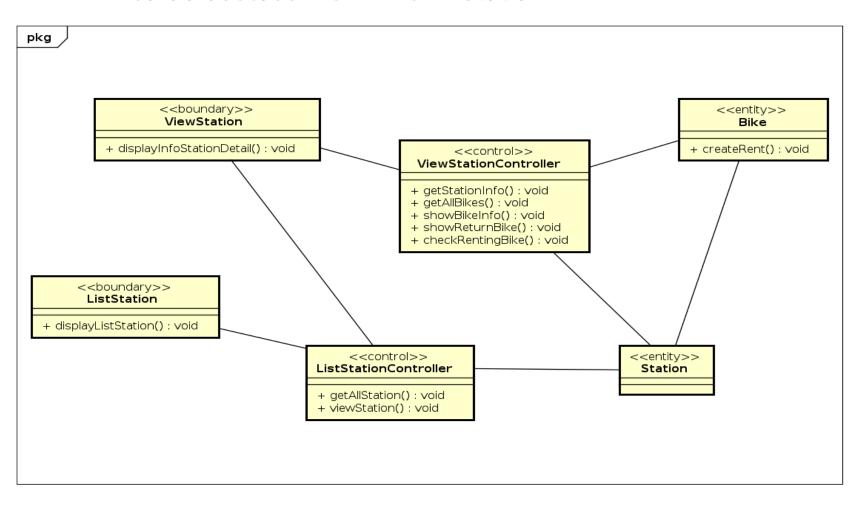
### II: Biểu đồ tương tác xem thông tin chi tiết xe



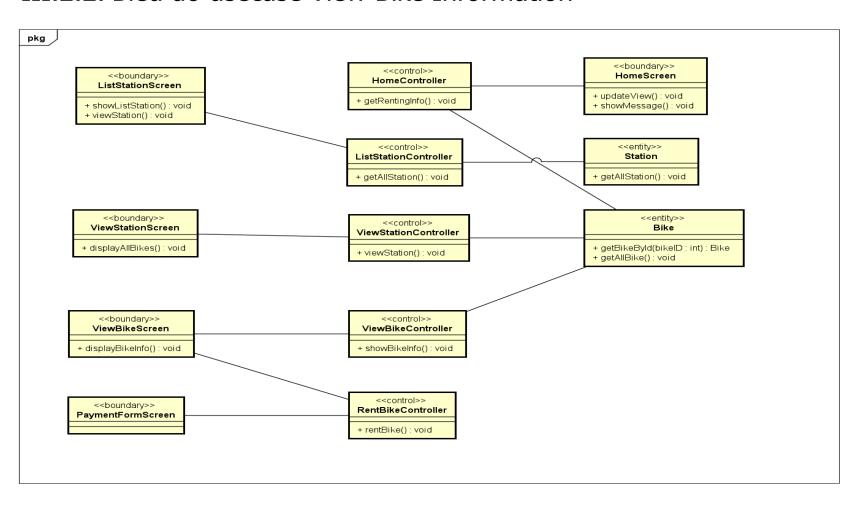
### III.1. Biểu đồ tổng quan của hệ thống



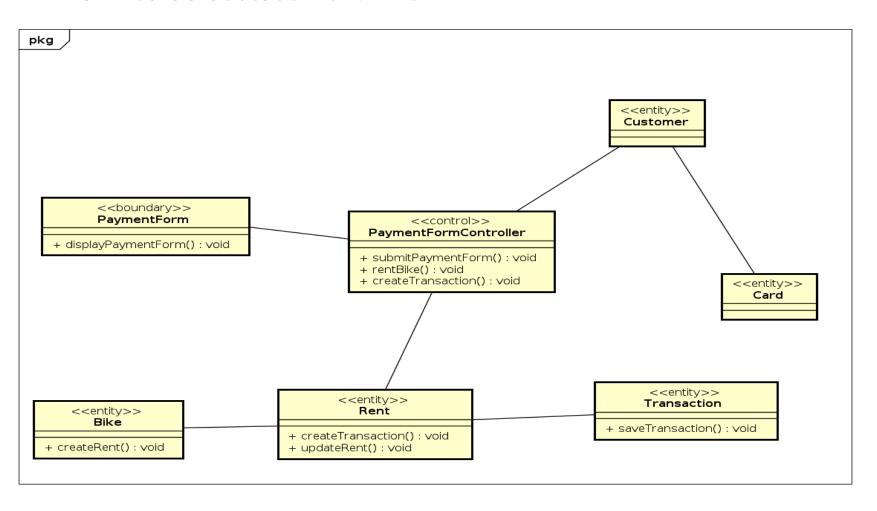
#### III.2.1. Biểu đồ usecase View Bike In Station



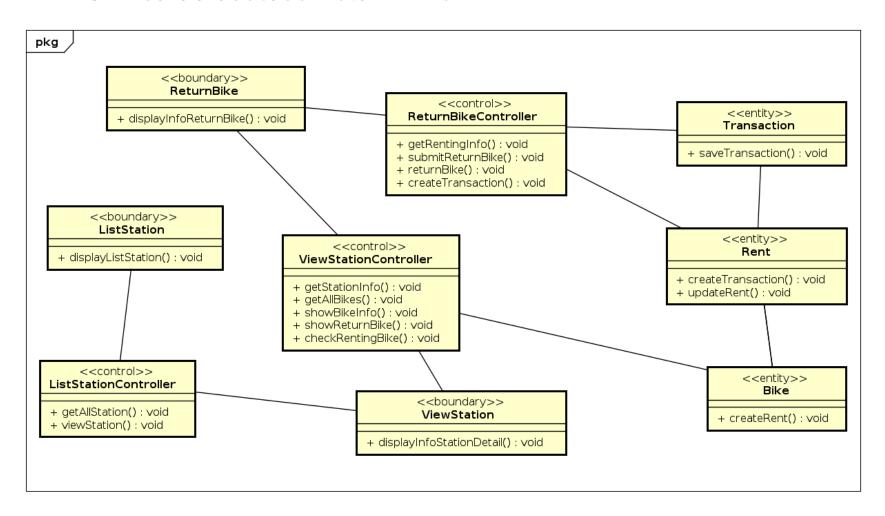
### III.2.2. Biểu đồ usecase View Bike Information



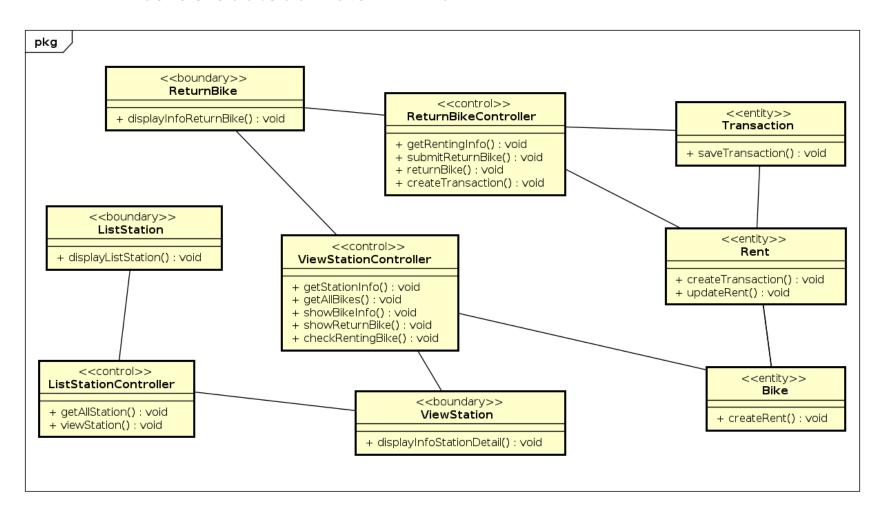
#### III.2.3. Biểu đồ usecase Rent Bike



### III.2.3. Biểu đồ usecase Return Bike



#### III.2.4. Biểu đồ usecase Return Bike



#### IV.1. Design Concepts

- Một số Common Coupling trong hệ thống:
- + Trong lớp ViewStationController, phương thức searchBike và phương thức addEvent đề sử dụng chung thuộc tính **bikeIDSelected**

```
/**
 * Tîm kiếm xe theo mã id xe
 */
public void searchBike() {
   if (txtBikeCode.getText().isEmpty()) return ;
   else {
      int id = Integer.parseInt(txtBikeCode.getText());
      int bikeID = station.getBikeByID(id).id;
      if (bikeID != 0) {
            Contants.bikeIDSelected = bikeID;
            showBikeInfo();
      }
      else {
            showMessage("Không tìm thấy xe !");
      }
   }
}
```

```
public void addEvents() {
   btnViewBike.setOnMouseClicked(e -> {
       if(tbvListBike.getSelectionModel().getSelectedItem() != null) {
           Contants.bikeIDSelected = tbvListBike.getSelectionModel().getSelectedItem().id;
           showBikeInfo():
   });
   btnReturnBike.setOnMouseClicked(e -> {
       checkBikeRenting();
   });
   btnRefresh.setOnMouseClicked(e -> {
       qetAllBikes();
   btnSearch.setOnMouseClicked(e -> {
       searchBike();
   tbvListBike.setRowFactory(tv -> {
       TableRow<Bike> row = new TableRow<>();
       row.setOnMouseClicked(e -> {
           if (e.getClickCount() == 2 && !(row.isEmpty())) {
               Contants.bikeIDSelected = row.getItem().id;
               showBikeInfo();;
       });
```

#### IV.1. Design Concepts

- Một số Common Coupling trong hệ thống:
- + Trong lớp ListStationController, phương thức searchBike và phương thức addEvent đề sử dụng chung thuộc tính **currentRenID**

```
/**
  * Lay thong tin thuê xe neu người đó đang thuê xe
  */
public void getRentingInfo() {
    rent.setRentFromID(Contants.currentRentID);
    bike.setBikeFromID(rent.bikeID);
    card.setCard(HomeController.currentUser.getCustomerCard());
    // update view
    updateView();
}
```

```
* Nhiêm vu: xử lý trả xe, goi đến api trả xe
 * @param code: lênh giao dich: "pay" hay "refund"
 * @param totalMoney: tổng số tiền giao dịch
public void returnBike( String code, int totalMoney) {
   String res;
        res = InterbankService.processTransaction(card, code, totalMoney);
   // System.out.print("Code: " + res);
        if("00".equals(res)) {
            bike.updateBike("available", Contants.stationIDSelected);
            rent.updateRent(current, totalTimeRent);
            Rent rent = new Rent(Contants.currentRentID);
            rent.createTransaction("return", "Return bike" , totalMoney);
            updateRentingBike();
            showMessage(Configs.MESSAGE SUCCESS);
            Stage stage = (Stage) btnClose.getScene().getWindow();
            stage.close();
            showMessage(Contants.response(res));
   } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
```

#### IV.1. Design Concepts

-Hệ thống vi phạm Logical cohesion: cùng truy cập vào database thì nên đặt vào một package riêng nhưng trong hệ thống, package model và package utilities đều có lớp truy cập vào database

- Lớp Contants trong package utilities

```
🚦 Package Explorer 🛭 Ju JUnit 📅 🗖 🕡 Contants.java 🛱
                                                   public static Connection getSubserverConnection() throws ClassNotroungexception. Subexception:
                                                                                                                                                                           8=
                                                       String dblRL = "idbc:sqlserver://localbost:databaseName=FcoBikeRentalDatabase:user=group18:password=123456":
    Connection conn = DriverManager.getConnection(dbURL);
       ▶ III Bike.java
       ▶ ☐ Card.java
                                             49
        La Customer.java
       ▶ 🛺 Rent.java
                                                   * Nhiêm vụ: Lấy tất cả các bải xe có trong cơ sở dữ liệu
       Machine Station.iava
                                                   * @return danh sách bāi xe
                                                    * @throws ClassNotFoundException
       ▶ □ Transaction.iava
                                                   * @throws SQLException
    ▶ ⊕ resources
    ▶ 攝 test
                                                   public static ArrayList<Station> getAllStations() throws ClassNotFoundException, SOLException (
                                             57
    58
                                                      Connection conn = getSQLServerConnection();
       Configs.java
                                             59
       ▶ 	 Contants.java
                                                      ArrayList<Station> list = new ArrayList<Station>():
       ▶ InterbankService.java
                                                      String sel = "SELECT * FROM Station";
                                             63
                                                      Statement stmt = conn.createStatement();
         HomeScreen.fxml
                                                      ResultSet rs = stmt.executeQuerv(sel);

← ListStation.fxml

                                                      while(rs.next()) {
         PaymentForm.fxml
                                                         int id = rs.getInt(1);
         ReturnBike fxml
                                                          String name = rs.getString(2);
                                                          String address = rs.getString(3);
         ViewBike.fxml
                                                          int totalBike = rs.getInt(4);
         ViewStation.fxml
                                                          int available = rs.getInt(5);
  ▶ M JRE System Library [ire]
                                            73
                                                          Station st = new Station(id, totalBike, available, name, address);
  ▶ 

M HTTP
                                             74
  JavaFx
                                             75
                                                                   Writable
```

### IV.1. Design Concepts

-Hệ thống vi phạm Logical cohesion: cùng truy cập vào database thì nên đặt vào một package riêng nhưng trong hệ thống package model và package utilities đều có lớp có thể truy cập vào database

-Ví dụ 1 lớp Bike trong package model

```
Package Explorer ⋈ Ju JUnit
                               □ □ D Contants.iava D Bike.iava 🛭
                                        49
                                                  this.name = name:
                                         51
                                                  this.status = status:
    52
      ▶ 🎝 Bike.java
                                         53
                                                  this.description = description:
      Card.java
                                         54
                                         55
      Customer.java
                                         56⊖
      ▶ № Rent.iava
                                               * <u>Nhiêm vu</u>: <u>khởi tạo</u> bike
      Station.java
                                               * @param bikeID
                                         59
      In Transaction.java
                                               public void setBikeFromID(int bikeID) {
    ▶ ∰ resources
    ▶ 🚜 test
                                                    String select1 = "SELECT * FROM Bike WHERE bikeID = " + bikeID;
                                         63
                                                    Statement stmt = Contants.com.createStatement():
    64
                                                     ResultSet rs = stmt.executeOuerv(select1):
      Configs.java
                                                  while(rs.next()) {
      67
                                                       this.id = bikeID:
      ▶ № InterbankService.java
                                                        this.type = rs.getString(2);
                                                       this.description = rs.getString(3):
                                                       this.price = rs.getInt(4);
        M HomeScreen.fxml
                                         71
                                                       this.name = rs.getString(5);
        ListStation.fxml
                                         72
                                                        this.status = rs.getString(6);
        RaymentForm.fxml
                                                        this.battery = rs.getInt(7);
                                                        this.stationID = rs.getInt(8);
        ReturnBike.fxml
                                         75

√ ViewBike.fxml

                                                 } catch (SQLException e) {
        ViewStation.fxml
                                                    // TODO Auto-generated catch block
                                         78
                                                     e.printStackTrace();
  ▶ 

Mark JRE System Library [jre]
                                         79
  ▶ ➡ HTTP
                                         80
  JavaFx
                                         81
```

### IV.1. Design Concepts

- Ví dụ về việc sử dụng Sequential Cohesion: đầu ra của phương thức này là thành một thành phần trong phương thức khác
- -Ví dụ: Trong lớp PaymentFormController, phương thức checkBlankField() là đầu vào cơ sở của phương thức submitPaymentForm() kiểm tra xem các trường nhập vào có hợp lệ hay không

```
/**
 * Nhiêm vu: Kiểm tra xem tắt cả các trường trong form đã được nhập chưa
 * @return true hoặc false
 */
public boolean checkBlankField() {
    if (txtCardHolderName.getText().isEmpty()) return false;
    if (txtCardNumber.getText().isEmpty()) return false;
    if (txtCardBank.getText().isEmpty()) return false;
    if (txtCardExpirationDate.getText().isEmpty()) return false;
    if (txtCardCVV.getText().isEmpty()) return false;
    return true;
}
```

```
/**
  * Nhiêm vu: xử lý khi submit form
  */
public void submitPaymentForm() {
    if (checkBlankField()) {
        card.cardHolderName = txtCardHolderName.getText();
        card.expirationDate = txtCardExpirationDate.getText();
        card.securityCode = txtCardCVV.getText();
        card.issuingBank = txtCardBank.getText();
        current = Contants.getCurrentTime();
        rentBike(bike.getDepositMoney());
    } else {
        showMessage("Hāy nhập đầy đủ thông tin !");
    }
}
```



### TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

