**Cover page**

**Mô tả phiên bản**

Mục lục

I. Giới Thiệu1

1.1 Mục đích2

1.2 Phạm vi2

1.3 Tổng quan2

1.4 Tài liệu tham khảo2

1.5 Định nghĩa và viết tắt2

II. Tổng quan hệ thống4

III. Kiến trúc hệ thống4

3.1 Bản thiết kế kiến trúc hệ thống (Architectural Design)5

3.2 Mô tả phân rã (Decomposition Description)5

3.3 Lý do thiết kế (Design Rationale)5

VI. Thiết kế dữ liệu4

4.1 Mô tả dữ liệu (Data Description)5

4.2 Từ điển dự liệu (Data Dictionary)5

V. Thiết kế bộ phận (Component Design)4

VI. Thiết kế giao diện tương tác4

6.1 Tổng quan giao diện tương tác5

6.2 Hình ảnh màn hình5

6.3 Đối tượng màn hình và hành động5

VII. Ma trận yêu cầu4

VIII. Phụ lục4

I. Giới Thiệu

1.1 Mục đích

1.2 Phạm vi

1.3 Tổng quan

1.4 Tài liệu tham khảo

1.5 Định nghĩa và viết tắt

II. Tổng quan hệ thống

III. Kiến trúc hệ thống

3.1 Bản thiết kế kiến trúc hệ thống (Architectural Design)

3.3.1 Bản thiết kế hệ thống (System Architecture)

3.3.2 Mô hình lớp (Class diagram)

3.2 Mô tả phân rã (Decomposition Description)

3.2.1 Register

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.2.2 Registration

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram)**

3.3.3 Customer registration

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.4 Technician registration

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.5 Evaluate

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.6 CustomerGUI

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.7 Printinfo

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.8 CustomerOrder

Đối tượng của class này sẽ được tạo ra trong class Customer GUI. Class này chứa thông tin về đơn hàng của khách hàng. Class này chứa những phương thức để thiết đặt dữ liệu đơn hàng, lấy địa chỉ của khách hàng thông qua GPS, đóng gói dữ liệu khách hàng.

3.3.8.1 SetData

**Mục đích:** Thiết đặt dữ liệu của khách hàng

**Prototype:** public void setData(name : string, age : int, address : int, phone : string, externalStandard : specType)

**Ngõ vào:** name - string, age - integer, address - integer, phone - string, externalStandard - specType

**Ngõ xuất: Không**

**Gọi bởi: CustomerGUI**

**Gọi: Không**

**Giải thuật:**

1. Nhận tham số truyền vào và lưu vào data (kiểu orderRecord)

**Biểu đồ tuần tự (Squence diagram):**

3.3.8.2 getGPSdata

**Mục đích:** Lấy dữ liệu vị trí của khách hàng

**Prototype:** public boolean getGPSdata()

**Ngõ vào:** Không

**Ngõ xuất:** Không

**Gọi bởi: CustomerGUI**

**Gọi:** Không

**Giải thuật:**

1. Gọi API tương tác để lấy dữ liệu từ GPS (kinh độ, vĩ độ)
2. Dùng API GoogleMap để tra cứu kinh độ và vĩ độ
3. Trả về kết quả tìm được và lưu vào biến private GPSaddress

**Biểu đồ tuần tự (Squence diagram):**

3.3.8.3 getData

**Mục đích:** Đóng gói dữ liệu và vận chuyển

**Prototype:** public orderType getData ()

**Ngõ vào:** Không

**Ngõ xuất:** Object kiểu orderType

**Gọi bởi: OrderHandler**

**Gọi:** Không

**Giải thuật:**

1. Khởi tạo biến kiểu orderType
2. Lấy dữ liệu từ các biến private.
3. Truyền dữ liệu từ các biến private sang biến orderType.

**Biểu đồ tuần tự (Squence diagram):**

* + 1. OrderHandler

Đối tượng của class là một đối tượng duy nhất (Singleton) chịu trách nhiệm xử lý đơn hàng khi có đơn hàng tới.

3.3.9.1 getOrderData

**Mục đích:** Nhận đơn hàng

**Prototype:** public boolean getOrderData(inputData : CustomerOrder)

**Ngõ vào:** inputData - CustomerOrder

**Ngõ xuất:** Không

**Gọi bởi:** Không

**Gọi:**

**Giải thuật:**

1. Nhận tham số truyền vào

**Biểu đồ tuần tự (Squence diagram):**

3.3.10 ChooseTechnicalHandler

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.11 Confirm

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.12 FeedbackHandler

**Mục đích:**

**Prototype:**

**Ngõ vào:**

**Ngõ xuất:**

**Gọi bởi:**

**Gọi:**

**Giải thuật:**

**Biểu đồ tuần tự (Squence diagram):**

3.3.13 Technical

Class này chứa thông tin về Kỹ thuật viên ( idTechnical, GPSaddress, stateFree). Class được tạo khi KTV login vào hệ thống và đang trong trạng thái onjob

*3.3.13.1: DoComplete*

**Mục đích**: xác nhận hoàn thành công việc

**Prototype:** public boolean DoComplete()

**Ngõ vào:** không

**Ngõ xuất:** boolean

**Gọi bởi:** Technical

**Gọi:** không

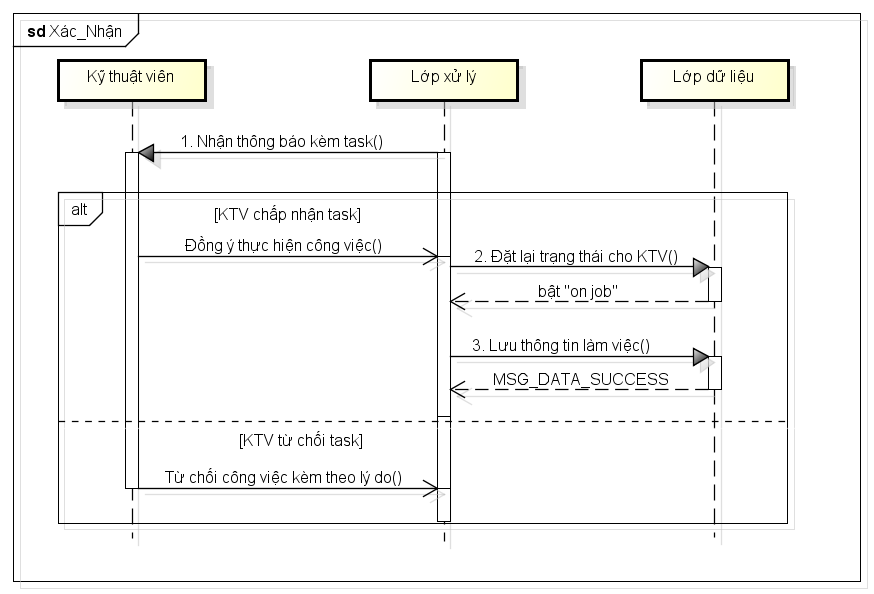
**Giải thuật:**

1. Set giá trị trả về của hàm thành “true”

3.3.14 ReceiveMenuTask

Class này được gọi khi Technical được tạo

**Sequence Diagram**



*3.3.14.1: getListTask*

**Mục đích:** nhận danh sách công việc mà hệ thống đã ứng với ID định danh của KTV

**Prototype:** public List<orderType> getListTask (int idTechnical)

**Ngõ vào:** int idTechnical (ID định danh của KTV)

**Ngõ xuất:** List<orderType> (gồm 1 menuTask)

**Gọi bởi****:** ReceiveMenuTask

**Gọi:** không

**Giải thuật:**

1. Gọi list orderType ứng với ID technical từ cơ sở dữ liệu
2. Hiển thị ListView chứa số lượng, số thứ tự các task

*3.3.14.2: upDateTask*

**Mục đích:** cập nhật lại danh sách công việc sau khoảng thời gian nhất định (10s), hoặc khi công việc đã được nhận bởi kĩ thuật viên khác

**Prototype:** public boolean upDateTask(int idTechnical)

**Ngõ vào:** int idTechnical (ID định danh của KTV)

**Ngõ xuất:** boolean

**Gọi bởi:** ReceiveMenuTask

**Gọi:** getListTask

**Giải thuật:**

1. Gọi getListTask sau khoảng thời gian 10s.
2. Trả về true nếu đã update.

*3.3.14.3: getInfomationTaskbyID*

**Mục đích:** mỗi Task sẽ có 1 ID riêng, hàm này sẽ lấy thông tin task được KTV chọn trong menuTask

**Prototype:** public List<orderType> getInfomationTaskbyID (int orderID)

**Ngõ vào**: intorderID

**Ngõ xuất:** List<orderType>

**Gọi bởi:** ReceiveMenuTask

**Gọi:** không

**Giải thuật:**

1. Lấy thông tin orderType ứng với ID order (ID của đơn hàng)
2. Hiển thị ListView chứa thông tin name, số điện thoại, địa chỉ, giá của đơn hàng

*3.3.14.4 :* *AcceptTask*

**Mục đích:** hàm được gọi khi KTV chấp nhận task (chọn “Accept”) hàm này sẽ được gọi

**Prototype:** public boolean AcceptTask (int idTask, int idTechnical)

**Ngõ vào:** int idTask, int idTechnical

**Ngõ xuất:** boolean

**Gọi bởi:** ReceiveMenuTask

**Gọi:** không

**Giải thuật:**

1. thông tin về đơn hàng và KTV đó sẽ được lưu vào hệ thống

2. đặt lại--> stateFree = false

*3.3.14.5 :* *RejectTask*

**Mục đích:** khi KTV từ chối task hàm này sẽ được gọi

**Prototype:** public boolean RejectTask (int idTask, int idTechnical)

**Ngõ vào:** int idTask, int idTechnical

**Ngõ xuất:** boolean

**Gọi bởi:** ReceiveMenuTask

**Gọi:** không

**Giải thuật:**

1. gọi TextBox yêu cầu KTV nêu lý do

2. stateFree = true

3.3 Lý do thiết kế (Design Rationale)

VI. Thiết kế dữ liệu

4.1 Mô tả dữ liệu (Data Description)

4.2 Từ điển dự liệu (Data Dictionary)

V. Thiết kế bộ phận (Component Design)

VI. Thiết kế giao diện tương tác

6.1 Tổng quan giao diện tương tác

6.2 Hình ảnh màn hình

6.3 Đối tượng màn hình và hành động

VII. Ma trận yêu cầu

VIII. Phụ lục