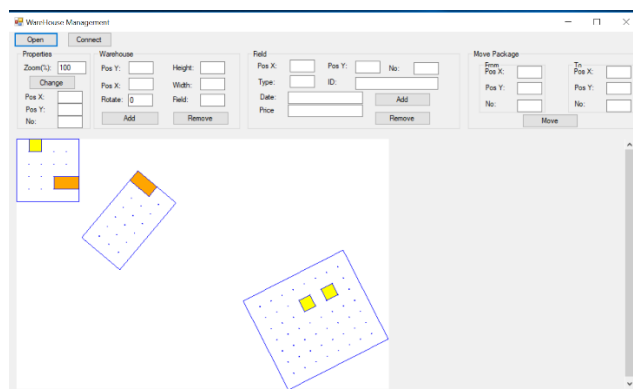


BẢO CÁO THỰC HÀNH



PHẦN MỀM QUẢN LÝ KHO ĐƠN GIẢN

Tên: Đỗ Thành Nhơn - 1512387

Phan Hồng Hạnh Trinh - 1512600

Châu Hoàng Long – 1512292

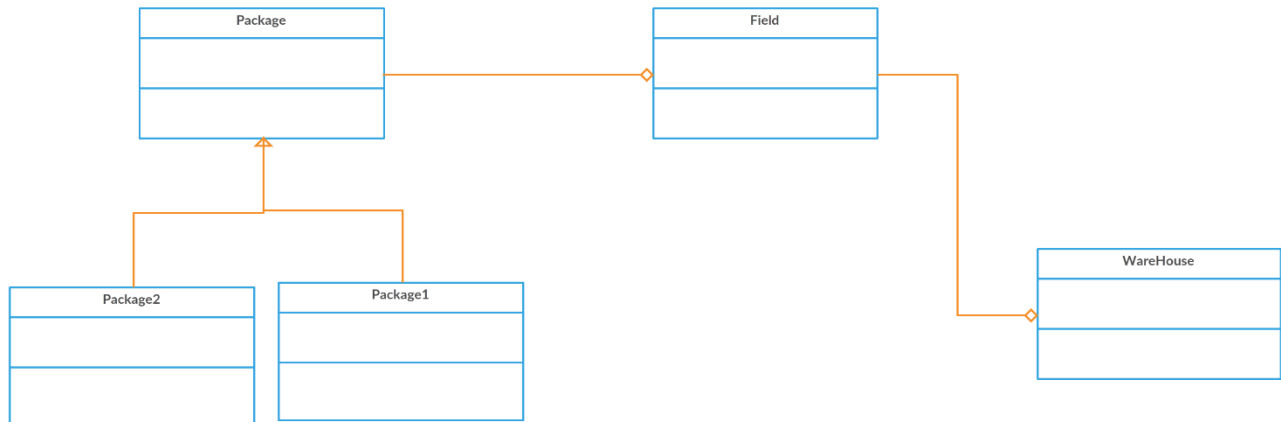
Trần Phước Lộc - 1512302

Lớp: 15CNTN

GVHD: Nguyễn Vinh Tiệp

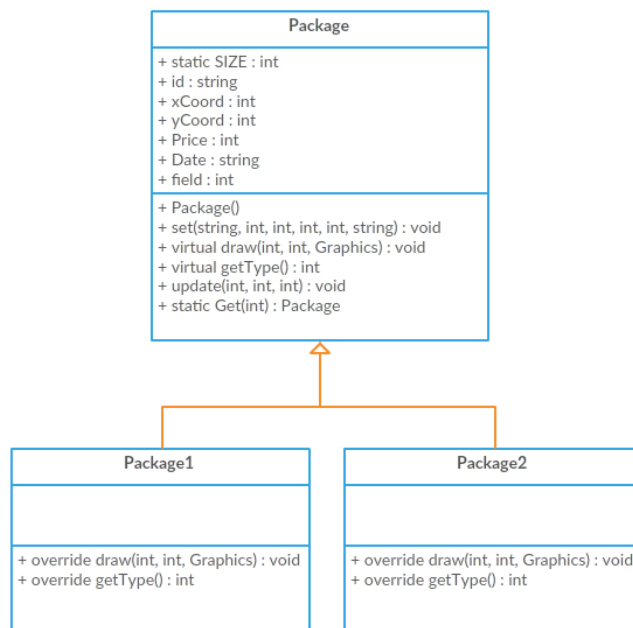
I. Tổ chức dữ liệu	3
1. Class Package	3
1.1. Class Package1	4
1.2. Class Package2	4
2. Class Field.....	4
3. Class Warehouse	6
II. Xử lý các sự kiện	7
1. Di chuyển chuột.....	8
2. Kéo thả chuột	8
III. Các công nghệ sử dụng	9
IV. Vấn đề và giải pháp	9
1. Vẽ khu vực có góc xoay	9
2. Xác định tọa độ trên màn hình	10
3. Đồng bộ hoá.....	10
V. Hướng dẫn sử dụng.....	10
1. Giao diện chương trình	10
2. Đọc dữ liệu từ file input.....	10
3. Thêm và xóa một khu vực mới	11
4. Thêm và xóa một kiện hàng	11
5. Di chuyển kiện hàng	12
6. Phóng to/ thu nhỏ.....	12
7. Thao tác trực tiếp trên một khu vực	12
8. Kết nối.....	13
VI. Kế hoạch thực hiện.....	13
VII. Phân công công việc	14

I. Tổ chức dữ liệu



1. Class Package

Mô phỏng kiện hàng



a. Thuộc tính

+ static int SIZE: hệ số phóng to

```
+ string id: mã hàng hóa

+ int xCoord: hoành độ của vị trí
+ int yCoord: tung độ của vị trí
+ int Price: giá kiện hàng
+ string Date: ngày nhập kho
+ int field: số kho mà kiện hàng thuộc về.
```

b. Phương thức

```
+ Package(): Phương thức khởi tạo mặc định với id = "", Date =
"", xCoord = 0, yCoord = 0, Price = 0

+ void set(string ID, int x, int y, int no, int price, string
date): thiết lập các thuộc tính của class theo các tham số tương ứng

+ void update(int x, int y, int no): cập nhật lại các giá trị
xCoord, yCoord, field của class

+ virtual void draw(int x, int y, Graphics gp): vẽ kiện hàng

+ virtual int getType(): trả về loại kiện hàng

+ static Package Get(int i): trả về đối tượng là kiện hàng loại
i.
```

1.1. Class Package1

Mô phỏng kiện hàng loại 1. Đây là class kế thừa class Package

Các phương thức override:

```
+ void draw(int x, int y, Graphics gp)
+ int getType().
```

1.2. Class Package2

Mô phỏng kiện hàng loại 2. Đây là class kế thừa class Package

Các phương thức override:

```
+ void draw(int x, int y, Graphics gp)
+ int getType().
```

2. Class Field

Mô phỏng khu vực chứa các kiện hàng

a. Thuộc tính

```
+ int x_coord, y_coord: hoành độ của vị trí
+ int height, width: chiều cao, chiều dài của khu vực
```

- + double rotate: góc xoay (đơn vị radian)
- + Dictionary<KeyValuePair<int, int>, Package> field: map các vị trí có kiện hàng với kiện hàng hiện hữu tại vị trí đó.

b. Phương thức

- + Field(int x, int y, int h, int w, double r): hàm khởi tạo khu vực theo các tham số tương ứng
- + bool isExist(KeyValuePair<int, int> p): kiểm tra xem tại vị trí p có kiện hàng nào không
- + Package getPackage(KeyValuePair<int, int> p): lấy ra package tại tọa độ p, nếu không tồn tại thì trả về null
- + bool add(KeyValuePair<int, int> pos, Package p): thêm kiện hàng p vào vị trí pos (nếu không thêm được thì không làm gì cả)
- + bool del(KeyValuePair<int, int> pos): xóa kiện hàng tại tọa độ pos
- + bool isTrueCoordinate(KeyValuePair<int, int> p): kiểm tra tọa độ p có nằm trong khuôn khổ của khu vực không
- + double getH(): lấy giới hạn bên dưới của khu vực
- + double getW(): lấy giới hạn bên phải của khu vực
- + void draw(Graphics gp): vẽ khu vực
- + static void ftranslate(ref double x, ref double y, double dx, double dy): tịnh tiến kiện hàng tại tọa độ (x, y) theo vector (dx, dy)
- + static void frotate(ref double x, ref double y, double a): xoay kiện hàng tại tọa độ (x, y) theo góc xoay rotate
- + KeyValuePair<int, int> convert(double x, double y): chuyển đổi tọa độ con trỏ chuột (x, y) trên màn hình sang tọa độ thực (x, y) của kiện hàng.

Field
<pre> + x_coord : int + y_coord : int + height : int + width : int + rotate : float + field Dictionary<KeyValuePair<int, int>, Package> ; </pre>
<pre> + Field(int, int, int, int, float) + isExist(KeyValuePair<int, int>) : bool + getPackage(KeyValuePair<int, int>) : Package + add(KeyValuePair<int, int>, Package) : bool + del(KeyValuePair<int, int>) : bool + isTrueCoordinate(KeyValuePair<int, int>) : bool + getH() : double + getW() : double + draw(Graphics) : void + static ftranslate(double, double, double, double) : void + static frotate(double, double, double) : void + convert(double, double) : KeyValuePair<int, int> </pre>

3. Class Warehouse

Mô phỏng kho chứa.

Warehouse
<pre> + xMax : double + yMax : double + lf : List<Field> </pre>
<pre> + Warehouse() + isEmpty() : bool + updateXY() : void + add(Field) : void + add(int, int, int, int, int) : bool + add(int, int, int, Package) : bool + draw(Graphics) : void + isExist(int, int) : Field + MovePackage(KeyValuePair<int, int>, int, KeyValuePair<int, int>, int) : bool + convert(double, double) : KeyValuePair<KeyValuePair<int, int>, int> + top_left(HashSet<int>) : int + inputInfo(string) : void + outputInfo(string) : void + removeField(int) : bool </pre>

a. Thuộc tính

+ double xMax, yMax: giới hạn phải và giới hạn dưới của khung hình khi vẽ

+ List<Field> lf: danh sách các khu vực.

b. Phương thức

+ bool isEmpty(): kiểm tra xem trong kho có khu vực nào không

+ void updateXY(): cập nhật lại giới hạn phải và giới hạn dưới của khung hình

+ void add(Field a): add thêm một khu vực vào Warehouse

+ bool add(int x, int y, int w, int h, double r): thêm một khu vực mới có kích thước w, h, vào tọa độ (x, y) với góc xoay r

+ bool add(int n, int x, int y, Package p): thêm một kiện hàng có tọa độ (x, y) vào khu vực thứ n

+ void draw(Graphics gp): vẽ kho

+ Field isExist(int x, int y): trả về Field tại tọa độ (x, y) nếu nó tồn tại

+ bool MovePackage(KeyValuePair<int, int> p1, int field1, KeyValuePair<int, int> p2, int field2): dời kiện hàng tại tọa độ p1 trong khu vực thứ field1 sang tọa độ p2 trong khu vực thứ field2

+ KeyValuePair<KeyValuePair<int, int>, int> convert(double x, double y): chuyển đổi tọa độ con trỏ chuột (x, y) trên màn hình sang tọa độ thực (x, y) của kiện hàng

+ int top_left(HashSet<int> set): tìm khu vực ở vị trí trái-trên nhất

+ void inputProcess(string s): xử lý thông tin nhập vào từ chuỗi s

+ string inputInfo(string fileName): nhập thông tin từ file vào chuỗi

+ void outputInfo(string fileName): xuất thông tin của Warehouse vào file

+ bool removeField(int index): xóa khu vực thứ index.

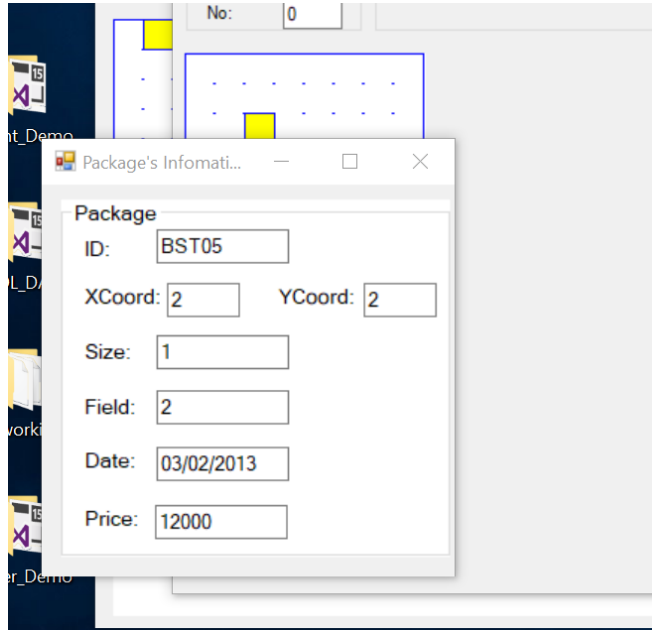
+ string toText(): Chuyển đổi dữ liệu sang chuỗi string đã được mã hoá để gửi đến các Client khác.

+ bool setCons(string s): từ chuỗi bị mã hoá nhận được từ sever, hàm này sẽ giải mã chuỗi và tiến hành cập nhật dữ liệu để đồng bộ.

II. Xử lý các sự kiện

1. Di chuyển chuột

Khi di chuyển chuột, nếu con trỏ chuột ở vị trí của kiện hàng nào thì chương trình sẽ hiển thị một form chứa thông tin của kiện hàng đó. Khi con trỏ chuột rời khỏi, form được tắt đi.



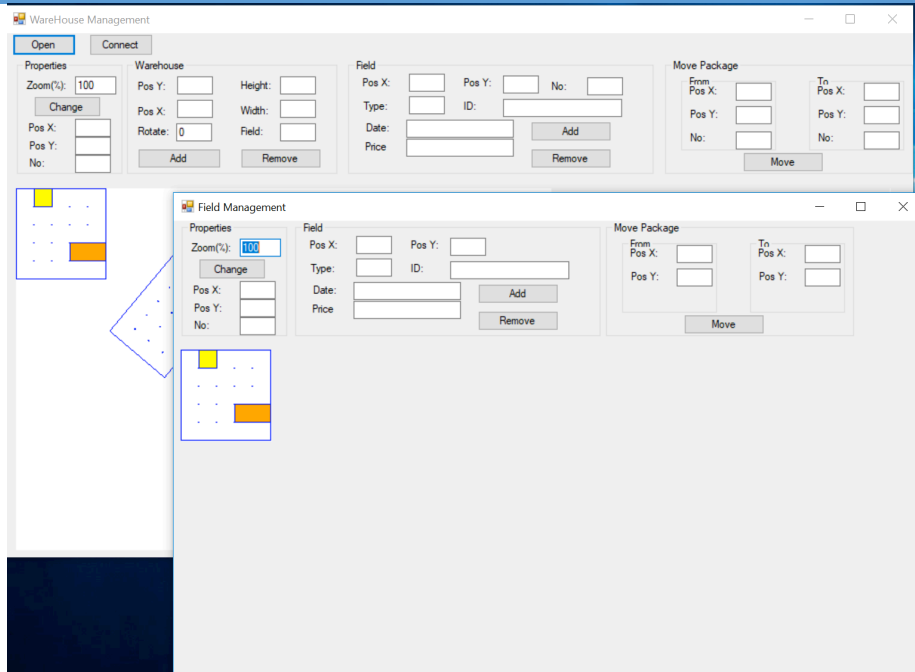
2. Kéo thả chuột

a. Chuột trái

Khi click chuột trái vào một kiện hàng và di chuyển nó tới vị trí khác, kiện hàng sẽ di chuyển tới vị trí tương ứng.

b. Chuột phải

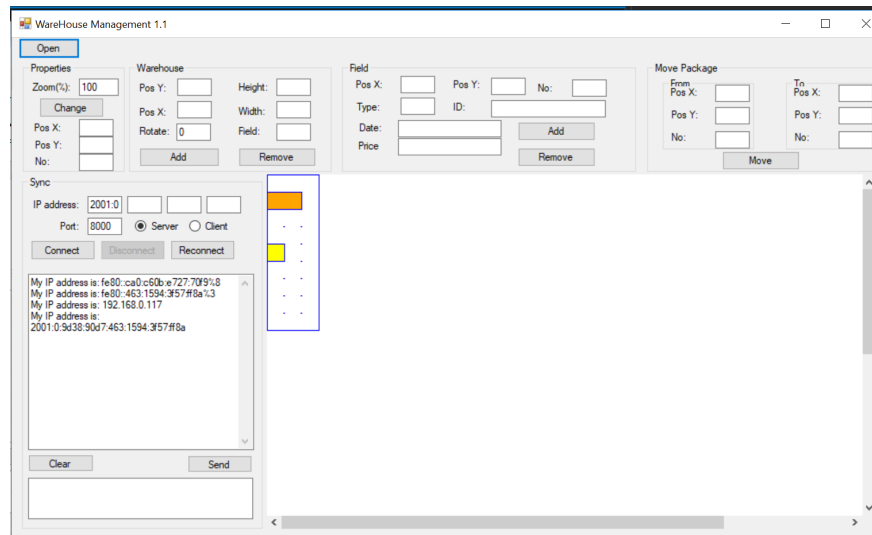
Trong phạm vi của vùng kéo thả, kích hoạt một form mới hiển thị thông tin của khu vực trái-trên nhất và cho phép người dùng thao tác trên các kiện hàng.



III. Các công nghệ sử dụng

- Windows Form Application C#
- Graphics GDI+
- Socket communication TCP Sever/Client

IV. Vấn đề và giải pháp



1. Về khu vực có góc xoay

Vấn đề: Thư viện đồ họa Graphics của C# xoay đối tượng xung quanh trục, mà ta cần xoay đối tượng xung quanh tâm của nó.

CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

Giải pháp: Tịnh tiến khu vực sao cho tâm của nó trùng với gốc tọa độ, xoay đối tượng quanh gốc theo góc xoay tương ứng. Sau đó tịnh tiến khu vực lại vị trí sao cho tâm của nó trùng với tâm ban đầu.

2. Xác định tọa độ trên màn hình

Vấn đề: Tọa độ của con trỏ chuột trên màn hình khác với tọa độ thực của khu vực và kiện hàng. Vì vậy không thể sử dụng tọa độ của con trỏ chuột để xác định kiện hàng.

Giải pháp: Chia tọa độ của con trỏ chuột cho hệ số nhân. Tính tâm của của khu vực. Xoay khu vực với góc tương ứng rồi dời khu vực sao cho tâm của nó trùng với gốc tọa độ.

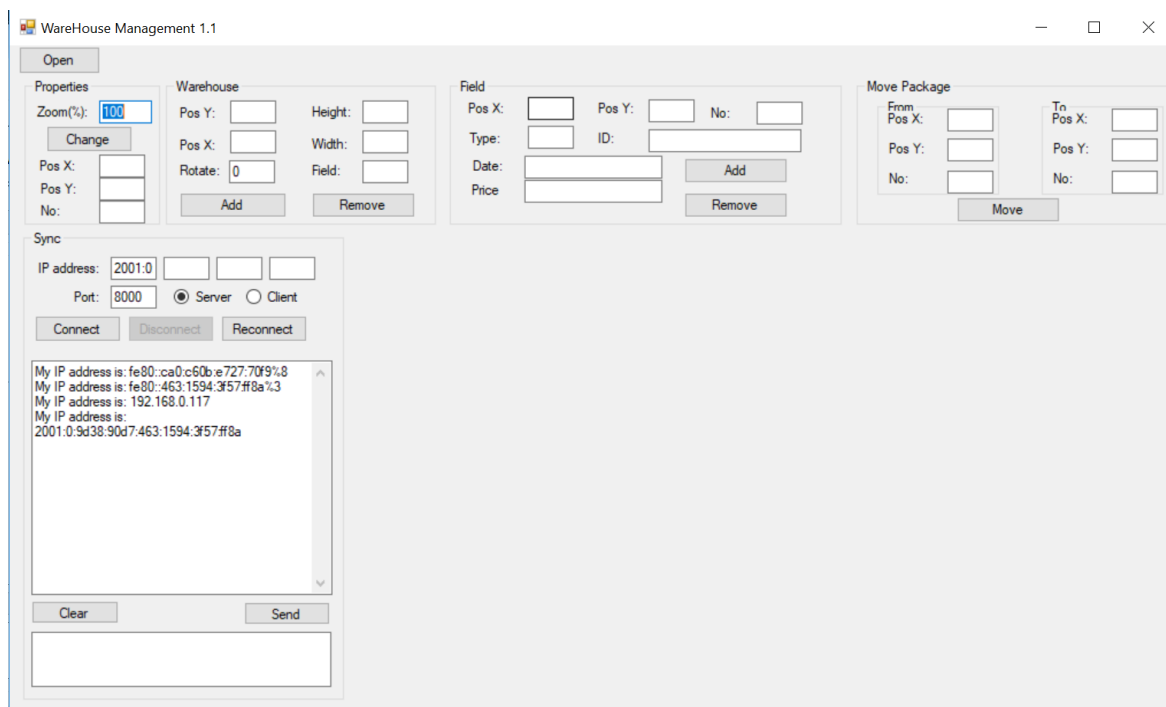
3. Đồng bộ hoá.

Vấn đề: Đồng bộ hoá theo thời gian thực giữa cái máy tính.

Giải pháp: sử dụng Socket để kết nối các máy tính lại với nhau. Có 1 sever và nhiều Client. Khi sever nhận tín hiệu thay đổi từ 1 Client thì ngay lập tức Sever sẽ gửi thông báo và gói tin là 1 string đến cho tất cả các Client được kết nối để cập nhật.

V. Hướng dẫn sử dụng

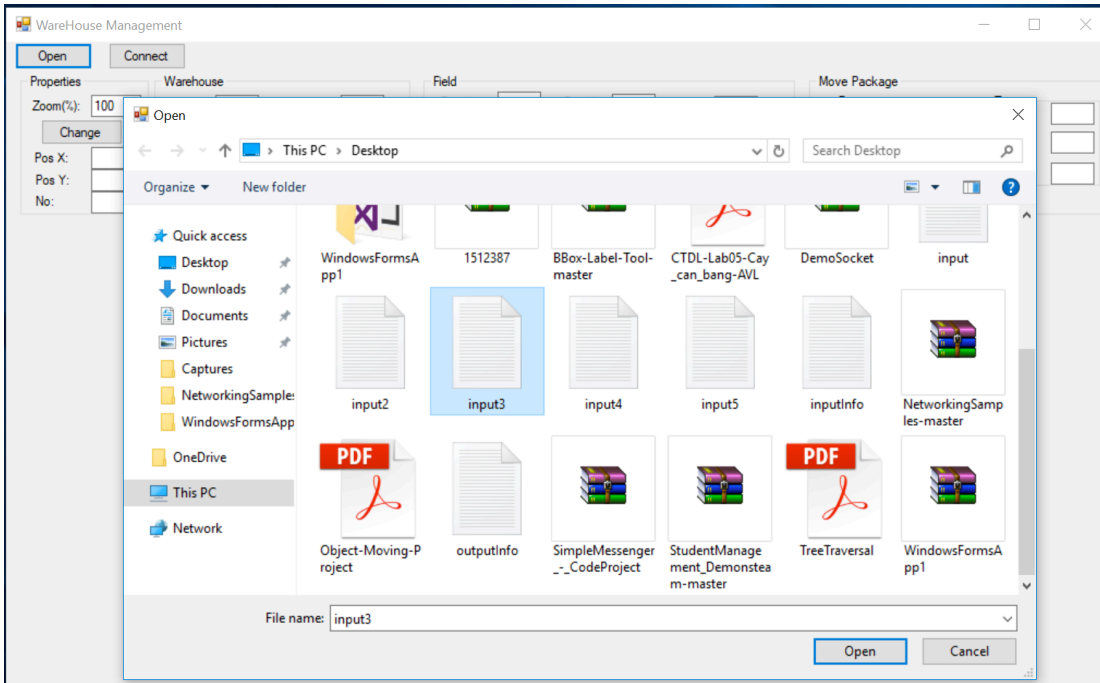
1. Giao diện chương trình



2. Đọc dữ liệu từ file input

CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

Khi người dùng click chuột vào button "open", một file dialog hiện lên cho người dùng chọn file input. Khi nhấn "open", chương trình load dữ liệu từ file input, hiển thị ra màn hình.



3. Thêm và xóa một khu vực mới

Tại groupbox WareHouse, người dùng điền thông tin của khu vực, sau đó nhấn nút "add" để thêm vào hoặc "remove" để xóa.

Warehouse	
Pos Y:	<input type="text"/>
Height:	<input type="text"/>
Pos X:	<input type="text"/>
Width:	<input type="text"/>
Rotate:	<input type="text" value="0"/>
Field:	<input type="text"/>
<input type="button" value="Add"/>	
<input type="button" value="Remove"/>	

4. Thêm và xóa một kiện hàng

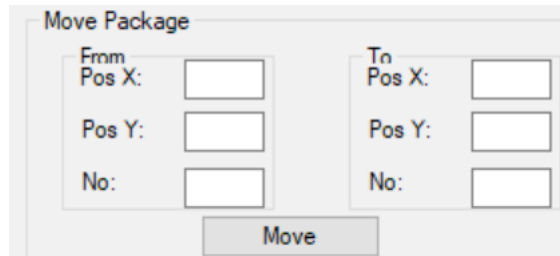
Tại groupbox Field, người dùng điền thông tin của kiện hàng, sau đó nhấn nút "add" để thêm vào hoặc "remove" để xóa.

Field					
Pos X:	<input type="text"/>	Pos Y:	<input type="text"/>	No:	<input type="text"/>
Type:	<input type="text"/>	ID:	<input type="text"/>		
Date:	<input type="text"/>				
Price:	<input type="text"/>				
<input type="button" value="Add"/>					
<input type="button" value="Remove"/>					

5. Di chuyển kiện hàng

Có hai cách để di chuyển

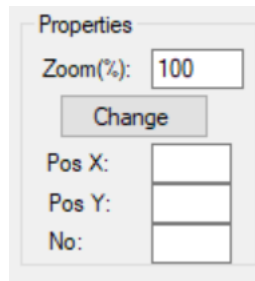
a. Tại groupbox "MovePackage", người dùng điền tọa độ của kiện hàng cần được di chuyển và tọa độ nơi cần di chuyển tới, sau đó nhấn "move".



b. Người dùng có thể trực tiếp dùng chuột trái để kéo thả.

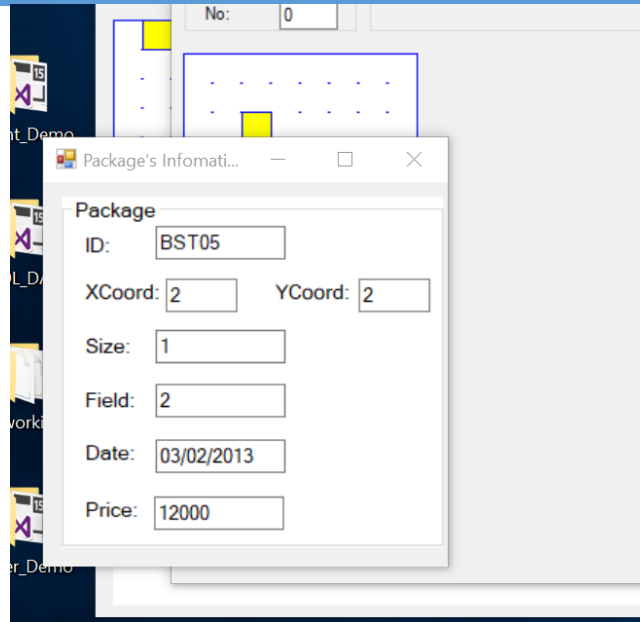
6. Phóng to/ thu nhỏ

Tại groupbox "Properties", người dùng điền vào thông số đơn vị % và nhấn nút "change".



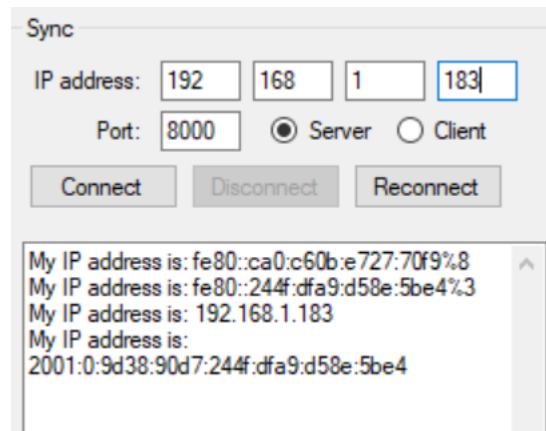
7. Thao tác trực tiếp trên một khu vực

Người dùng sử dụng chuột phải để chọn một vùng trên màn hình. Khu vực trái-trên nhất sẽ được hiển thị trên một form mới. Người dùng có thể thao tác thay đổi các kiện hàng trực tiếp trên form này và thông tin sẽ được cập nhật lại khi tắt form.



8. Kết nối.

Người dùng sẽ nhập địa chỉ IP vào các ô ở Groupbox Sync sau đó nhấn nút Connect. Lưu ý Sever thì chọn Sever và Connect với chính địa chỉ IP của mình, còn Client thì sẽ nhập địa chỉ IP của Sever và chọn Client.



VI. Kế hoạch thực hiện

Tuần	Công việc
1	Tìm hiểu ngôn ngữ lập trình C# và các công nghệ có liên quan
2	<ul style="list-style-type: none"> - Thiết kế và cài đặt cấu trúc dữ liệu - Thiết kế giao diện người dùng của chương trình

CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

3	Cài đặt, xử lý các sự kiện theo yêu cầu đề bài
4	<ul style="list-style-type: none">- Tiến hành kiểm tra và sửa lỗi- Viết báo cáo

VII. Phân công công việc

STT	Thành viên	Công việc
1	Trình	Cài đặt cấu trúc dữ liệu
2	Lộc	Vẽ và các hàm có liên quan
3	Nhơn	Thêm, xóa, di chuyển, kéo thả
4	Long	Kết nối, đồng bộ dữ liệu