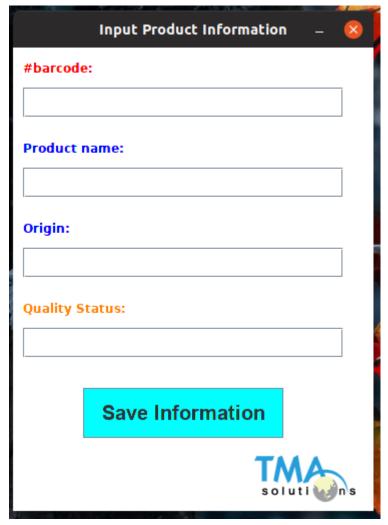
## Úng dụng java GUI:

- Cho người dùng nhập thông tin sản phẩm ban đầu
- Gồm có barcode để phân biệt các sản phẩm với nhau và dùng để cập nhật trạng thái sản phẩm trong quá trình vận chuyển
- Khi bấm save chương trình sẽ kết nối với smart contract trên blockchain và đưa data lên chain, cả quá trình mất khoảng 15s, tùy thuộc vào tốc độ mạng và gas price (gas price càng cao thì các miner sẽ ưu tiên giao dịch của chúng ta), chương trình sẽ dừng ở đó cho đển khi giao dịch hoàn tất trên chain và những ô nhập liệu sẽ tự động xóa để chuẩn bị cho dữ liêu mới



## Source code GUI java được quản lý bằng gradle có cấu trúc như sau

```
build.gradle

plugins {
    id 'java'
    id 'org.web3j' version '4.1.2'
    }

//jar {
    baseName = 'GUI'
    version = '1.0.3'
    manifest {
        | attributes 'Main-Class': "org.web3j.sample.Application" }
    /*/
    group 'org.web3j'
    version '0.1.2'

sourceCompatibility = 1.8

repositories {
    mavenCentral()
    y
    web3j {
        generatedPackageName = 'org.web3j.sample.contracts.generated'
        excludedContracts = ['Mortal']

    web3jVersion = '4.1.1'
    logbackVersion = '1.2.3'
    junitVersion = '4.12'
    dependencies {
        compile files('libs/json-simple-1.1.1.jar');
        inplementation "org.web3jVersion",
        "ch.qos.logback:logback-core:$logbackVersion",
        "ch.qos.logback:logback-classic:$logbackVersion"
        testImplementation "junit:junitVersion"
}
```

```
■ Project ▼ ♣ ★ −

▼ sample-project-gradle ~/Desktop/test
▶ gradle
▶ idea
▶ build
▶ gradle
▶ src
♣ .gitignore
♠ .project

■ build.gradle
□ gradlew
♣ gradlew.bat
♣ README.md
♣ sample-project-gradle.iml
■ settings.gradle

▶ Illi External Libraries
▼ Scratches and Consoles
```

ở phần dependencies ta thấy có dòng implementation org.web3j:code đây là phần khai báo sử dụng thư viện web3j, nếu ở thư mục local của gradle chưa có thư viện web3, gradle sẽ tự tải và cấu hình classpath tích hợp thư viện vào project.

```
OPEN EDITORS 1 UNSAVED
                                                                                          src > main > solidity > Product_manager > $ Products_manager.sol
1 pragma solidity >= 0.4.26 <0.7.0;</pre>
 • 🖺 Untitled-1
                                                                                                              address owner;
struct product{
> 🖙 .gradle
                                                                                                                     uint256 barCode;
> 🖿 .idea
> 💌 .vscode
                                                                                                                      string origin:
 > 🛅 build
                                                                                                                     uint256 temp;
uint256 humid;
   🔛 gradle
                                                                                                                     uint256 latitude:
                                                                                                             mapping (address => product) public productMapping;
function test() public pure returns (string memory)
                                                                                                                      address uniqueID = address(uint256(sha256(abi.encodePacked(msg.sender, now))));
productMapping[uniqueID].barCode = BarCode;
productMapping[uniqueID].name = Name;
productMapping[uniqueID].origin = Origin;
productMapping[uniqueID].quality = Quality;
return uniqueID;
        resources
          🗟 logback.xml

}
function update(address uniqueID, uint256 Temp, uint256 Humid, uint256 Latitude, uint256 Longtitude) public retur
productMapping[uniqueID].temp = Temp;
productMapping[uniqueID].humid = Humid;
productMapping[uniqueID].latitude = Latitude;
productMapping[uniqueID].longitude = Longtitude;
return "Product is undatad":

   > 🌅 resources
 JAVA DEPENDENCIES
 MAVEN PROJECTS
```

File smart contract Products\_manager.sol được đặt tại thư mục như trên.

Sau khi build, một class wrapper implement các hàm của smart contract dưới dạng ngôn ngữ java và chúng ta có thể sử dụng trong ứng dụng java của mình

```
src > main > java > org > web3j > sample > __ Application.java > _ Application | * <a nret= nttps://www.rinkeby.lo/#raucet >xinkeby raucet</a>
OPEN EDITORS 1 UNSAVED
 • 🖺 Untitled-1
                                                                                                                           public class Application {
    private static JFrame frame;
    private static JLabel barcode, proc_name, proc_origin, proc_quality;
    private static JTextField barcodeInput, proc_nameInput, proc_originInput, proc_qualityInput;
    // endougn_static_JMenugan_proc_qualityChencing;
> 🛅 bin
 > 🛅 build
 > 🔛 gradle
 > 📭 libs
                                                                                                                                            // private static JMenuBar proc quality(hoosing; // private static JMenu quality; // private static JMenu quality; // private static JMenu quality1, quality2, quality3; private static JMenuTutem quality1, quality2, quality3; private static int width, labelsize; private static int i i = 0;
∨ 🔯 src
     ∨ 📻 java
           ✓ is sample
■ Application.java
                                                                                                                                    Run|Debug
public static void main(String[] args) throws Exception {
    frame = new JFrame("Input Product Information");
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    Garage setLecationRelativeTo(frame);
                                                                                                                                                               rrame.setDefaultCloseOperation()Frame.EXIT_ON_CLOSE
frame.setLocationRelativeTo(frame);
frame.setSize(360, 500);
frame.getContentPane().setBackground(Color.WHITE);
frame.setLayout(null);
frame.setVisible(true);
frame.setResizable(false);
width = 320;
labelsize = 200;
// Objects
      > 🐚 resources
     > solidity
   > 🛅 resources
   > 📑 test
     build.gradle
     💉 gradlew
                                                                                                                                                                 barcode = new JLabel("<html><font color='red'>#barcode:</font><html>");
frame.add(barcode);
JAVA DEPENDENCIES
 MAVEN PROJECTS
```

Ứng dụng chính nằm trong thư mục như hình trên.