# System Description

## Hello Tractor Ecommerce Marketplace

### Document Version: 1.0

### Date: 26th November 2024

## Contents

1. [Document Management](#document-management)  
   1.1 [Contributors](#contributors)  
   1.2 [Version Control](#version-control)
2. [Overview](#overview)  
   2.1 [Service Description](#service-description)  
   2.2 [Data Model](#data-model)  
   2.3 [Reporting](#reporting)  
   2.4 [Technology](#technology)  
   2.5 [Development Tools](#development-tools)  
   2.6 [Interfaces and Services](#interfaces-and-services)  
   2.7 [Access, Authentication, and Authorization](#access-authentication-and-authorization)  
   2.8 [Delivery](#delivery)
3. [Support Details](#support-details)  
   3.1 [Third Party](#third-party)  
   3.2 [Documentation](#documentation)  
   3.3 [Standard Tasks](#standard-tasks)  
   3.4 [Troubleshooting Guide](#troubleshooting-guide)
4. [API Documentation](#api-documentation)

­­­­­­

### 1. Document Management

#### 1.1 Contributors

| **Role** | **Unit** | **Name** |
| --- | --- | --- |
| Team Lead & Full Stack Dev | Entire Application | Andrew Kimani |
| Backend Engineer | Backend | Benjamin Rotich |
| Frontend Engineer | Frontend | Caleb Muthama |

### 2. Overview

#### 2.1 Service Description

The Hello Tractor Ecommerce Marketplace enables the sale of second-hand tractors and agricultural implements globally. Key features include:

* **Buyer Process**: Browse listings, contact sellers, and complete purchases with secure user accounts.
* **Seller Process**: List products, track analytics, and manage customer interactions.
* **Admin Features**: Approve listings, track user activity, and manage platform operations.

**Flow Diagram**:

* Buyers → Search & Filter → Contact Sellers → Purchase Tractors
* Sellers → Register → Create Listings → Manage Inventory

#### 2.2 Data Model

The platform is built with a relational database structure for managing users, tractors, spare parts, and transactions. Directus serves as the backend API layer.  
Key relationships:

* **Users** ↔ **Listings** (1-to-many)
* **Listings** ↔ **Categories** (many-to-one)
* **Transactions** ↔ **Users** (1-to-1)

#### 2.3 Reporting

Reporting is integrated directly into the application for sellers through an analytics dashboard. It includes:

* **Seller Insights**: Number of views, inquiries, and sales for each product.
* **Automated Jobs**: Daily updates for featured products and dealer availability.

#### 2.4 Technology

* **Frontend**: Next.js
* **Backend**: Directus
* **Deployment**: Vercel
* **APIs**: REST-based, documented in Section 4.

#### 2.5 Development Tools

* **IDE**: VS Code
* **Version Control**: GitHub
* **Automated Deployment**: Vercel CI/CD
* **Debugging Tools**: Chrome DevTools, Postman

#### 2.6 Interfaces and Services

* **Incoming APIs**: Secure registration, login, and product listing endpoints.
* **Outgoing APIs**: Notifications for sellers (email/SMS).
* **Web Services**: RESTful API provided by Directus for CRUD operations.

#### 2.7 Access, Authentication, and Authorization

* **Access**: Web-based via Vercel deployment.
* **Authentication**: Email/phone verification with JWT tokens.
* **Authorization**: Role-based permissions (Admin, Seller, Buyer).

#### 2.8 Delivery

The application is delivered as a standalone web platform.

### 3. Support Details

#### 3.1 Third Party

* **Backend**: Directus
* **Deployment**: Vercel
* **Primary Contact**: Hello Tractor Technical Support

#### 4. API Documentation

### Descriptive API Documentation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| API Endpoint | HTTP Method | Description | Input Parameters | Output |
| /api/signup | POST | Registers a new user. | Email, Name, Phone | Auth tokens |
| /api/signin | POST | Logs in a user. | Email, Password | Auth tokens or errors |
| /api/featuredProducts | GET | Fetches most viewed products. | None | List of products |
| /api/dealers | GET | Retrieves dealer information. | None | Dealer list |
| /api/shop | POST | Fetches product categories. | Filters (optional) | Product list |
| /api/addTractor | POST | Adds a tractor listing. | Tractor details | Success/Failure JSON |
| /api/addSparePart | POST | Adds a spare part listing. | Spare part details | Success/Failure JSON |
| /api/showContact | GET | Displays seller contact information. | None | Seller details |