Document

Inventory Rubber Tree by Drone and Rubber Tree-Tapping Robot

Contents

1 Introduction

In the rubber industry, the inspection of rubber plantations and latex tapping plays a crucial role in ensuring efficient production operations. However, labor scarcity and the large workload have posed many challenges to this process. To address this issue, we have decided to apply robotics technology to the working process to support and optimize labor-intensive tasks, while also digitizing the inspection and latex tapping process.

2 Project description

Rubber Plantation Area

We possess over 400,000 hectares of rubber plantation, including areas in Laos, Cambodia, and various regions in Southeast, Central Highlands, and South of Vietnam.

Utilizing Robots in the Workflow

We have decided to introduce robots into the working process to assist in labor-intensive tasks and digitize the workflow. The specially designed and programmed robots can automatically measure rubber trees and conduct tapping operations efficiently.

Inventory and Tapping Process

The rubber plantation inspection takes place at the end of the year, from July to November annually.

The inventory process involves the following steps:

- Measurement: Robots will autonomously measure the height and other parameters of rubber trees, from the base to approximately 1.3 meters above the ground.
- Reading: Robots accurately read and record data from the measured rubber trees.
- Data Input: The collected data is entered into the system through the robots, ensuring convenience and accuracy.
- Data Transmission to Head Office: After inspection, the data is transferred to the main headquarters for analysis and management purposes.

3 Structure

The following diagram provides an overview of database structure.

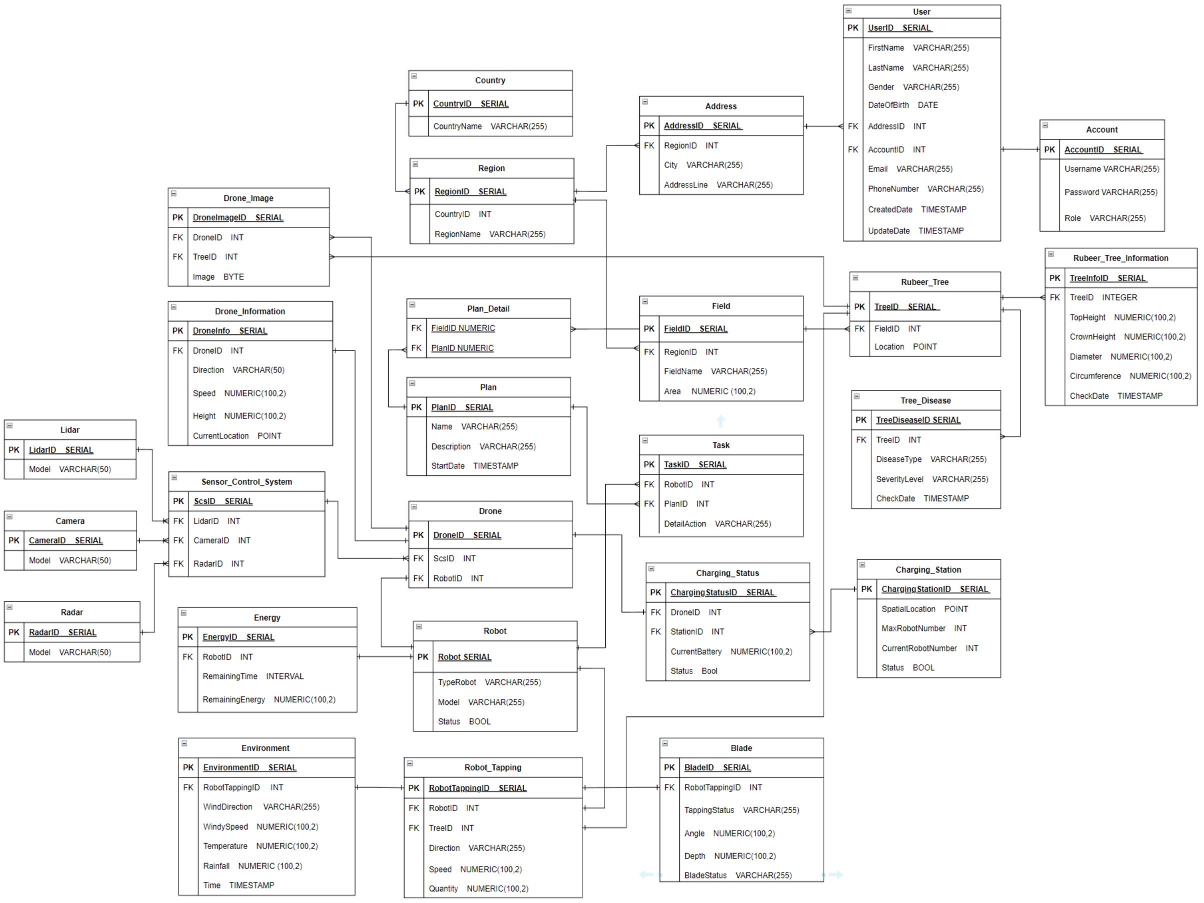

Figure 1: Diagram

3.1 Tables

The following sections describe the tables that make up the database central, in alphabetic order.

3.1.1 The Account table

The Account table lists information for all accounts.

Columns

	Name	Data type	N	Description / Attributes
🔑	AccountID	Serial		A surrogate primary key used to uniquely identify each information account in the table.
	Username	Varchar(255)		The username of the user.
	Password	Varchar(255)		The password of the user.
	Role	Varchar(255)		.

Linked from

	Table	Join	Title / Name / Description
+	User	**Account**.AccountID = User.AccountID	FK_User_Account_AccountID Foreign key constraint referencing Account.AccountID.

3.1.2 The Address table

The Address table contains address information for user.

Columns

	Name	Data type	N	Description / Attributes
🔑	AddressID	Serial		A surrogate primary key used to uniquely identify each address in the table.
	RegionID	Int		A foreign key pointing to the region table
	City	Varchar(255)		The name of the city.
	AddressLine	Varchar(255)		The house number and street name.

Links to

	Table	Join	Title / Name / Description
➤	Region	**Address**.RegionID = Region.RegionID	FK_Address_Region_RegionID Foreign key constraint referencing Region.RegionID.

Linked from

	Table	Join	Title / Name / Description
◄	User	**Address**.AddressID = User.AddressID	FK_User_Address_AddressID Foreign key constraint referencing Address.AddressID.

3.1.3 The Blade table

The Blade table

Columns

	Name	Data type	N	Description / Attributes
🔑	BladeID	Serial		A surrogate primary key used to uniquely identify each blade in the table.
	RobotTappingID	Int		A foreign key identifying the robot of this blade.
	TappingStatus	Varchar(255)		
	Angle	Numeric(100,2)		The angle of the blade determines the direction the blade moves.
	Depth	Numeric(100,2)		The depth of the blade is achieved while moving the blade over the rubber stem.
	BladeStatus	Varchar(255)		Describe the durability of the blade.

Links to

	Table	Join	Title / Name / Description
+	RobotTapping	**Blade**.RobotTappingID = RobotTapping.RobotTappingID	FK_Blade_RobotTapping_RobotTappingID Foreign key constraint referencing RobotTapping.RobotTappingID.

3.1.4 The Camera table

The Camera table lists information for all cameras.

Columns

	Name	Data type	N	Description / Attributes
🔑	CameraID	Serial		A surrogate primary key used to uniquely identify each camera in the table.
	Model	Varchar(255)		The model that the camera is using.

Linked from

	Table	Join	Title / Name / Description
≺	SensorControlSystem	**Camera**.CameraID = SensorControlSystem.CameraID	FK_SensorControlSystem_Camera_CameraID Foreign key constraint referencing Camera.CameraID.

3.1.5 The ChargingStatus table

The ChargingStatus table allows monitoring the progress and charging status of the robot at the charging station.

Columns

	Name	Data type	N	Description / Attributes
🔑	ChargingStatusID	Serial		A surrogate primary key that uniquely identifies the ChargingStatus.
	StationID	Int		A foreign key points to the ChargingStation table.
	DroneID	Int		A foreign key points to the drone table, identifies the drone is charging.
	CurrentBattery	Numeric(100,2)		The current battery level is charging.
	Status	Bool		Check the charging status is charging or not.

Links to

	Table	Join	Title / Name / Description
+	Drone	**ChargingStatus**.DroneID = Drone.DroneID	FK_ChargingStatus_Drone_DroneID Foreign key constraint referencing Drone.DroneID.
≻	ChargingStation	**ChargingStatus**.ChargingStationID = ChargingStation.ChargingStationID	FK_ChargingStatus_ChargingStation_ChargingStationID Foreign key constraint referencing ChargingStation.ChargingStationID.

3.1.6 The ChargingStation table

The ChargingStation table allows for the management and tracking of charging stations for robots. It provides information about the location, robot numbers, and current status of each station.

Columns

	Name	Data type	N	Description / Attributes
🔑	ChargingStationID	Serial		A surrogate primary key that uniquely identifies each ChargingStation.
	Location	Point		Location is determined by GPS coordinates.
	MaxRobotNumber	Int		The maximum number of robots that can be accommodated.
	CurrentRobotNumber	Int		The number of robots currently charging.
	Status	Bool		Check the status is still working or not.

Linked from

	Table	Join	Title / Name / Description
≺	ChargingStatus	**ChargingStation**.ChargingStationID = ChargingStatus.ChargingStationID	FK_ChargingStatus_ChargingStation_ChargingStationID Foreign key constraint referencing ChargingStation.ChargingStationID.

3.1.7 The Country table

The Country table contains a list of name countries.

Columns

	Name	Data type	N	Description / Attributes
🔑	CountryID	Serial		A surrogate primary key used to uniquely identify each country in the table.
	CountryName	Varchar(255)		The name of the country.

Linked from

	Table	Join	Title / Name / Description
≺	Region	**Country**.CountryID = Region.CountryID	FK_Region_Country_CountryID Foreign key constraint referencing Country.CountryID.

3.1.8 The Drone table

The Drone table lists information for all drone.

Columns

	Name	Data type	N	Description / Attributes
🔑	DroneID	Serial		A surrogate primary key used to uniquely identify the drone.
	RobotID	Int		A foreign key pointing at the robot table.
	ScsID	Int		A foreign key pointing at the SensorControlSystem table.

Links to

	Table	Join	Title / Name / Description
≻	SensorControlSystem	**Drone**.ScsID = SensorControlSystem.ScsID	FK_Drone_SensorControlSystem_ScsID Foreign key constraint referencing SensorControlSystem.ScsID.
+	Robot	**Drone**.RobotID = Robot.RobotID	FK_Drone_Robot_RobotID Foreign key constraint referencing Robot.RobotID.

Linked from

	Table	Join	Title / Name / Description
≺	DroneImage	**Drone**.DroneID = DroneImage.DroneID	FK_DroneImage_Drone_DroneID Foreign key constraint referencing Drone.DroneID.
+	DroneInformation	**Drone**.DroneID = DroneInformation.DroneID	FK_DroneInformation_Drone_DroneID Foreign key constraint referencing Drone.DroneID.
+	ChargingStatus	**Drone**.DroneID = ChargingStatus.DroneID	FK_ChargingStatus_Drone_DroneID Foreign key constraint referencing Drone.DroneID.

3.1.9 The DroneInformation table

The DroneInformation table describes the parameters of the drone when it is operating.

Columns

	Name	Data type	N	Description / Attributes
🔑	DroneInforID	Serial		A surrogate primary key used to uniquely identify the drone operation information.
	DroneID	Int		A foreign key pointing at the drone table.
	Direction	Varchar(255)		The direction the drone is moving.
	Speed	Numeric(100,2)		The current speed of the drone.
	Height	Numeric(100,2)		The current altitude of the drone.
	CurrentLocation	Point		The current location of drone.

Links to

	Table	Join	Title / Name / Description
+	Drone	**DroneInfromation**.DroneID = Drone.DroneID	FK_DroneInformation_Drone_DroneID Foreign key constraint referencing Drone.DroneID.

3.1.10 The DroneImage table

The DroneImage table stores information about images of trees taken by drones.

Columns

	Name	Data type	N	Description / Attributes
🔑	DroneImageID	Serial		A surrogate primary key used to uniquely identify each image.
	DroneID	Int		The drone which captured the image. This is a foreign key reference to the drone table.
	TreeID	Int		The tree associated with the image. This is a foreign key reference to the RubberTree table.
	Image	Byte		The image is captured. the image data as binary bytes.

Links to

	Table	Join	Title / Name / Description
≻	Drone	**DroneImage**.DroneID = Drone.DroneID	FK_DroneImage_Drone_DroneID Foreign key constraint referencing Drone.DroneID.

3.1.11 The Energy table

The Energy table represents information about energy levels and remaining time for robots.

Columns

	Name	Data type	N	Description / Attributes
🔑	EnergyID	Serial		A surrogate primary key used to uniquely identify the energy status of each robot in the table.
	RobotID	Int		A foreign key pointing at the robot table.
	RemainingTime	Interval		Remaining time the robot can operate.
	RemainingEnergy	Numeric(100,2)		Current energy of the robot.

Links to

	Table	Join	Title / Name / Description
+	Robot	**Energy**.RobotID = Robot.RobotID	FK_Energy_Robot_RobotID Foreign key constraint referencing Robot.RobotID.

3.1.12 The Environment table

The Environment stores information about the surrounding environmental factors affecting the rubber tree.

Columns

	Name	Data type	N	Description / Attributes
🔑	EnvironmentID	Serial		A surrogate primary key used to uniquely identify each environment information.
	WindDirection	Varchar(255)		The direction the wind moves.
	WindSpeed	Numeric(100,2)		Measured wind speed.
	Temperature	Numeric(100,2)		Current temperature of the environment.
	Rainfall	Numeric(100,2)		The amount of rain falling.
	Time	Timestamp		The time when the information was measured.

Links to

	Table	Join	Title / Name / Description
+	RobotTapping	**Environment**.RobotTappingID = RobotTapping.RobotTappingID	FK_Environment_RobotTapping_RobotTappingID Foreign key constraint referencing RobotTapping.RobotTappingID.

3.1.13 The Field Table

The Field table lists information for all fields.

Columns

	Name	Data type	N	Description / Attributes
🔑	FieldID	Serial		A surrogate primary key used to uniquely identify each field.
	RegionID	Int		A foreign key identifying the region that the field belongs to.
	FieldName	Varchar(255)		The name of field.
	Area	Numeric(100,2)		The area of the field.

Links to

	Table	Join	Title / Name / Description
≻	Region	**Field**.RegionID = Region.RegionID	FK_Field_Region_RegionID Foreign key constraint referencing Region.RegionID.

Linked from

	Table	Join	Title / Name / Description
≺	RubberTree	**Field**.FieldID = RubberTree.FieldID	FK_RubberTree_Field_FieldID Foreign key constraint referencing Field.FieldID.
≺	PlanDetail	**Field**.FieldID = PlanDetail.FieldID	FK_PlanDetail_Field_FieldID Foreign key constraint referencing Field.FieldID.

3.1.14 The Lidar table

The Lidar table lists information for all lidars.

Columns

	Name	Data type	N	Description / Attributes
🔑	LidarID	Serial		A surrogate primary key used to uniquely identify each lidar in the table.
	Model	Varchar(255)		The model that the lidar is using.

Linked from

	Table	Join	Title / Name / Description
←	SensorControlSystem	**Lidar**.LidarID = SensorControlSystem.LidarID	FK_SensorControlSystem_Lidar_LidarID Foreign key constraint referencing Lidar.LidarID.

3.1.15 The Plan table

The Plan table lists information for all plans.

Columns

	Name	Data type	N	Description / Attributes
🔑	PlanID	Serial		A surrogate primary key used to uniquely identify the plan.
	Name	Varchar(255)		The name of plan.
	Description	Varchar(255)		Detailed description of information about the plan.
	StartDate	Timestamp		The time to start implementing the plan.

Linked from

	Table	Join	Title / Name / Description
←	PlanDetail	**Plan**.PlanID = PlanDetail.PlanID	FK_PlanDetail_Plan_PlanID Foreign key constraint referencing Plan.PlanID.
←	Task	**Plan**.PlanID = Task.PlanID	FK_Task_Plan_PlanID Foreign key constraint referencing Plan.PlanID.

3.1.16 The PlanDetail table

The PlanDetail table is used to support a many-to-many relationship between plans and field.

The PlanDetail table refers to the Plan and Field tables using foreign keys.

Columns

	Name	Data type	N	Description / Attributes
🔑	FieldID	Int		A foreign key identifying the field.
🔑	PlanID	Int		A foreign key identifying the plan.

Links to

	Table	Join	Title / Name / Description
≻	Plan	**PlanDetail**.PlanID = Plan.PlanID	FK_PlanDetail_Plan_PlanID Foreign key constraint referencing Plan.PlanID.
≻	Field	**PlanDetail**.FieldID = Field.FieldID	FK_PlanDetail_Field_FieldID Foreign key constraint referencing Field.FieldID.

3.1.17 The Radar table

The Radar table lists information for all radars.

Columns

	Name	Data type	N	Description / Attributes
🔑	RadarID	Serial		A surrogate primary key used to uniquely identify each radar in the table.
	Model	Varchar(255)		The model that the radar is using.

Linked from

	Table	Join	Title / Name / Description
≺	SensorControlSystem	**Radar**.RadarID = SensorControlSystem.RadarID	FK_SensorControlSystem_Radar_RadarID Foreign key constraint referencing Radar.RadarID.

3.1.18 The Region table

The Region table lists information for all regions.

Columns

	Name	Data type	N	Description / Attributes
🔑	RegionID	Serial		A surrogate primary key used to uniquely identify each region in the table.
	CountryID	Int		.
	RegionName	Varchar(255)		Name of the region.

Links to

	Table	Join	Title / Name / Description
≻	Country	**Region**.CountryID = Country.CountryID	FK_Region_Country_CountryID Foreign key constraint referencing Country.CountryID.

Linked from

	Table	Join	Title / Name / Description
≺	Address	**Region**.RegionID = Address.RegionID	FK_Address_Region_RegionID Foreign key constraint referencing Region.RegionID.
≺	Field	**Region**.RegionID = Field.RegionID	FK_Field_Region_RegionID Foreign key constraint referencing Region.RegionID.

3.1.19 The Robot table

The Robot table lists information for all robots.

Columns

	Name	Data type	N	Description / Attributes
🔑	RobotID	Serial		A surrogate primary key used to uniquely identify each robot in the table.
	TypeRobot	Varchar(255)		Robot type (drone or tapping).
	Model	Varchar(255)		Model of robot.
	Status	Bool		Check the status is working or not.

Linked from

	Table	Join	Title / Name / Description
+	Energy	**Robot**.RobotID = Energy.RobotID	FK_Energy_Robot_RobotID Foreign key constraint referencing Robot.RobotID.
+	Drone	**Robot**.RobotID = Drone.RobotID	FK_Drone_Robot_RobotID Foreign key constraint referencing Robot.RobotID.
+	RobotTapping	**Robot**.RobotID = RobotTapping.RobotID	FK_RobotTapping_Robot_RobotID Foreign key constraint referencing Robot.RobotID.
<	Task	**Robot**.RobotID = Task.RobotID	FK_Task_Robot_RobotID Foreign key constraint referencing Robot.RobotID.

3.1.20 The RobotTapping table

The RobotTapping table lists information for all robot tappings.

Columns

	Name	Data type	N	Description / Attributes
🔑	RobotTappingID	Serial		A surrogate primary key used to uniquely identify each robot tapping in the table.
	RobotID	Int		A foreign key identifying the robot that the robot tapping belongs to.
	TreeID	Int		A foreign key identifying the tree that the robot tapping belongs to.
	Direction	Varchar(255)		The direction in which the robot performs rubber tapping.
	Speed	Numeric(100,2)		The speed of a robot in performing rubber tapping.
	Quantity	Numeric(100,2)		The amount of latex obtained from the tree. When a certain amount is reached, an announcement will be made.

Links to

	Table	Join	Title / Name / Description
+	Robot	**RobotTapping**.RobotID = Robot.RobotID	FK_RobotTapping_Robot_RobotID Foreign key constraint referencing Robot.RobotID.
+	RubberTree	**RobotTapping**.TreeID = RubberTree.TreeID	FK_RobotTapping_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.

Linked from

	Table	Join	Title / Name / Description
+	Blade	**RobotTapping**.RobotTappingID = Blade.RobotTappingID	FK_Blade_RobotTapping_RobotTappingID Foreign key constraint referencing RobotTapping.RobotTappingID.
+	Environment	**RobotTapping**.RobotTappingID = Environment.RobotTappingID	FK_Environment_RobotTapping_RobotTappingID Foreign key constraint referencing RobotTapping.RobotTappingID.

3.1.21 The RubberTree table

The RubberTree table stores a list of all the rubber trees in each fields.

Columns

	Name	Data type	N	Description / Attributes
🔑	TreeID	Serial		A surrogate primary key used to uniquely identify each rubber tree in the table.
	FieldID	Int		A foreign key identifying the field that the tree belongs to.
	Location	Point		Location is determined by GPS coordinates.

Links to

	Table	Join	Title / Name / Description
≻	Field	**RubberTree**.FieldID = Field.FieldID	FK_RubberTree_Field_FieldID Foreign key constraint referencing Field.FieldID.

Linked from

	Table	Join	Title / Name / Description
+	RobotTapping	**RubberTree**.TreeID = RobotTapping.TreeID	FK_RobotTapping_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.
≺	RubberTreeInformation	**RubberTree**.TreeID = RubberTreeInformation.TreeID	FK_RubberTreeInformation_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.
≺	TreeDisease	**RubberTree**.TreeID = TreeDisease.TreeID	FK_TreeDisease_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.
≺	PlanDetail	**RubberTree**.TreeID = PlanDetail.TreeID	FK_PlanDetail_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.

3.1.22 The RubberTreeInformation table

The RubberTreeInfomation table lists information for all rubber trees.

Columns

	Name	Data type	N	Description / Attributes
🔑	TreeInforID	Serial		A surrogate primary key used to uniquely identify each rubber tree in the table.
	TreeID	Int		A foreign key identifying the rubber tree that the information belongs to.
	TopHeight	Numeric(100,2)		The height from the base to the top of the tree.
	CrownHeight	Numeric(100,2)		The height from the base to the crown of the tree.
	Diameter	Numeric(100,2)		
	Circumference	Numeric(100,2)		The circumference of the rubber tree trunk.
	CheckDate	Timestamp		The date when data was collected.

Links to

	RubberTree	Join	Title / Name / Description
≻	RubberTree	**RubberTreeInformation**.TreeID = RubberTreeInformation.TreeID	FK_RubberTreeInformation_RubberTree_TreeID Foreign key constraint referencing RubberTreeInformation.TreeID.

3.1.23 The SensorControlSystem table

The SensorControlSystem table lists information for all SensorControlSystem.

Columns

	Name	Data type	N	Description / Attributes
🔑	ScsID	Serial		A surrogate primary key used to uniquely identify the sensor control system in the table.
	LidarID	Int		A foreign key pointing at the Lidar table; identifies the Lidar of the control system.
	CameraID	Int		A foreign key pointing at the camera table; identifies the camera of the control system.
	RadarID	Int		A foreign key pointing at the radar table; identifies the radar of the control system.

Links to

	Table	Join	Title / Name / Description
≻	Camera	**SensorControlSystem**.CameraID = Camera.CameraID	FK_SensorControlSystem_Camera_CameraID Foreign key constraint referencing Camera.CameraID.
≻	Lidar	**SensorControlSystem**.LidarID = Lidar.LidarID	FK_SensorControlSystem_Lidar_LidarID Foreign key constraint referencing Lidar.LidarID.
≻	Radar	**SensorControlSystem**.RadarID = Radar.RadarID	FK_SensorControlSystem_Radar_RadarID Foreign key constraint referencing Radar.RadarID.

Linked from

	Table	Join	Title / Name / Description
≺	Drone	**SensorControlSystem**.DroneID = Drone.DroneID	FK_SensorControlSystem_Drone_DroneID Foreign key constraint referencing Drone.DroneID.

3.1.24 The Task table

The Task table lists information for all tasks of plan.

Columns

	Name	Data type	N	Description / Attributes
🔑	TaskID	Serial		A surrogate primary key used to uniquely identify the task in the table.
	RobotID	Int		The robot which is being applied to. This is a foreign key reference to the customer table.
	PlanID	Int		A foreign key identifying the plan that the task belongs to.
	DetailAction	Varchar(255)		.

Links to

	Table	Join	Title / Name / Description
≻	Plan	**Task**.PlanID = Plan.PlanID	FK_Task_Plan_PlanID Foreign key constraint referencing Plan.PlanID.
≻	Robot	**Task**.RobotID = Robot.RobotID	FK_Task_Robot_RobotID Foreign key constraint referencing Robot.RobotID.

3.1.25 The TreeDisease table

The TreeDisease table lists information for all tree.

Columns

	Name	Data type	N	Description / Attributes
🔑	TreeDiseaseID	Serial		A surrogate primary key used to uniquely identify each tree_disease in the table.
	TreeID	Int		A foreign key pointing at the rubber_tree table; identifies which tree is infected.
	DiseaseType	Varchar(255)		The type of disease the tree is suffering from.
	SeverityLevel	Smallint		The severity of the plant being infected.
	CheckDate	Timestamp		Date of going to check the health status of the tree.

Linkes to

	Table	Join	Title / Name / Description
≻	RubberTree	**TreeDisease**.TreeID = RubberTree.TreeID	FK_TreeDisease_RubberTree_TreeID Foreign key constraint referencing RubberTree.TreeID.

3.1.26 The User table

The User table lists personal information for all users.

Columns

	Name	Data type	N	Description / Attributes
🔑	UserID	Serial		A surrogate primary key used to uniquely identify each user in the table.
	FisrtName	Varchar(255)		The user first name.
	LastName	Varchar(255)		The user last name.
	Gender	Varchar(5)		The user gender.
	DateOfBirth	Date		Date of birth
	AddressID	Int		TA foreign key identifying the customer address in the address table.
	AccountID	Int		A foreign key identifying the account in the account table.
	Email	Varchar(255)		The user email address.
	PhoneNumber	Varchar(11)		The user phone number.
	CreatedDate	Timestamp		The date the user was added to the system. This date is automatically set using a trigger during an INSERT.
	UpdateDate	Timestamp		When the row was created or most recently updated.

Links to

	Table	Join	Title / Name / Description
+	Account	**User**.AccountID = Account.AccountID	FK_User_Account_AccountID Foreign key constraint referencing Account.AccountID.
≻	Address	**User**.AddressID = Address.AddressID	FK_User_Address_AddressID Foreign key constraint referencing Address.AddressID.

4 Relationship

4.1 One-to-one

4.1.1 User and Account

- **Key Fields**:

- Account Table: Primary Key – AccountID
- User Table: Foreign Key – AccountID

- **Description**: Each user has an account, and each account is associated with only one user. The accountID field in the User table links back to the AccountID field in the Account table, establishing the relationship between the two tables.

4.1.2 Drone and DroneInformation

- **Key Fields**:

- Drone Table: Primary Key – DroneID
- DroneInformation Table: Foreign Key – DroneID

- **Description**: Each piece of information about the specifications of a drone belongs to only one drone, and each drone has its own information record. This relationship allows for the separation of drone-specific details into a separate table while maintaining a direct and unique connection to each drone record.

4.1.3 RotbotTaping and Environment

- **Key Fields**:

- RotbotTaping Table: Primary Key – RobotTappingID
- Environment Table: Foreign Key – RobotTappingID

- **Description**: Each robot tapping will have sensors that receive information about the environment, and the environment information will provide feedback to each robot. The environmental data is accurately recorded and tracked for each robot tapping operation.

4.1.4 RotbotTaping and Blade

- **Key Fields**:

- RotbotTaping Table: Primary Key – RobotTappingID
- Blade Table: Foreign Key – RobotTappingID

- **Description**: Each robot will be equipped with a blade for tapping latex, and each blade will belong to a specific robot tapping.

4.1.5 Rotbot and Energy

- **Key Fields**:

- Rotbot Table: Primary Key – RobotID
- Energy Table: Foreign Key – RobotID

- **Description**: Each robot will have its own energy level information. Similarly, each energy record in the Energy table is linked to one specific robot in the Robot table.

4.1.6 Drone and ChargingStatus

- **Key Fields**:

- Drone Table: Primary Key – DroneID
- ChargingStatus Table: Foreign Key – DroneID

- **Description**: Each drone will have its own information about the battery charge status, and the battery charge status will be updated separately for each drone.

4.2 One-to-many

4.2.1 Lidar and SensorControlSystem

- **Key Fields**:

- Lidar Table: Primary Key – LidarID
- SensorControlSystem Table: Foreign Key – LidarID

- **Description**: The Lidar table will store information about different types of lidar devices, and a sensor system will use a specific type of lidar. There can be multiple sensor systems using the same type of lidar.

4.2.2 Camera and SensorControlSystem

- **Key Fields**:

- Camera Table: Primary Key – CameraID
- SensorControlSystem Table: Foreign Key – CameraID

- **Description**: The Camera table will store information about different types of Camera devices, and a sensor system will use a specific type of camera. There can be multiple sensor systems using the same type of camera.

4.2.3 Radar and SensorControlSystem

- **Key Fields**:

- Radar Table: Primary Key – RadarID
- SensorControlSystem Table: Foreign Key – RadarID

- **Description**: The Radar table will store information about different types of Radar devices, and a sensor system will use a specific type of radar. There can be multiple sensor systems using the same type of radar.

4.2.4 Country and Region

- **Key Fields**:

- Country Table: Primary Key – CountryID
- Region Table: Foreign Key – CountryID

- **Description**: One country can have multiple regions associated with it, while each region belongs to only one specific country.

4.2.5 Region and Address

- **Key Fields**:

- Region Table: Primary Key – RegionID
- Address Table: Foreign Key – RegionID

- **Description**: One region can have multiple addresses associated with it, while each address belongs to only one specific region.

4.2.6 Address and User

- **Key Fields**:

- Address Table: Primary Key – AddressID
- User Table: Foreign Key – AddressID

- **Description**: Each address can be associated with multiple users, but each user is associated with a particular address.

4.2.7 Region and Field

- **Key Fields**:

- Region Table: Primary Key – RegionID
- Field Table: Foreign Key – RegionID

- **Description**: Each region can have many fields, but each field only belongs to a certain region.

4.2.8 Field and RubberTree

- **Key Fields**:

- Field Table: Primary Key – FieldID
- RubberTree Table: Foreign Key – FieldID

- **Description**: Each field has many rubber trees, but each rubber tree belongs to only one field.

4.2.9 RuberTree and RubberTreeInformation

- **Key Fields**:

- RuberTree Table: Primary Key – TreeID
- RuberTreeInformation Table: Foreign Key – TreeID

- **Description**: Each rubber tree will store information about it in multiple inventories, and each of that information will belong to a rubber tree.

4.2.10 RuberTree and TreeDisease

- **Key Fields**:

- RuberTree Table: Primary Key – TreeID
- TreeDisease Table: Foreign Key – TreeID

- **Description**: Each rubber tree will store information about its disease status in multiple inventories, and each of that information will belong to a rubber tree.

4.2.11 Tree and DroneImage

- **Key Fields**:

- Tree Table: Primary Key – TreeID
- DroneImage Table: Foreign Key – TreeID

- **Description**: Each tree will have images of that tree and each image will belong to a certain tree.

4.2.12 Drone and DroneImage

- **Key Fields**:

- Drone Table: Primary Key – DroneID
- DroneImage Table: Foreign Key – DroneID

- **Description**: Each drone will store information about the photos it has taken, and each photo will belong to a drone.

4.2.13 Plan and Task

- **Key Fields**:

- Plan Table: Primary Key – PlanID
- Task Table: Foreign Key – PlanID

- **Description**: Each plan will have different tasks and each task will belong to a certain plan.

4.2.14 Robot and Task

- **Key Fields**:

- Robot Table: Primary Key – RobotID
- Task Table: Foreign Key – RobotID

- **Description**: Each robot can be assigned multiple tasks, but each task can only be performed by one robot.

4.3 Many-to-Many

4.3.1 Plan and Field

- **Key Fields**:

- Plan Table: Primary Key – PlanID
- Field Table: Primary Key – FieldID
- PlanDetail Table: Foreign Key – PlanID and FieldID

- **Description**: The PlanDetail table is used to support a many-to-many relationship between plan and field. For each field in a given plan, there will be one row in the plan_detail table listing the field and plan. A field can be specified in multiple plans and a plan can be implemented on multiple fields.