CSC3170: Database System

Agriculture Internet of Things System

Dongyang Wu
118010324

Jiarui Li
118020229

Run Liu
118010188

Yuncong Cui
118010045

Zhixuan Liu
118010202

2020/2021 Semester 1

Table of Contents

Introduction and Motivation

As the fundamental industry of any countries in the world, the agriculture industry dealt with the basic energy need of human beings. Christiaensen and Demery (2007) point out that the contribution of economic growth to poverty reduction might differ across sectors because the benefits of growth might be easier for poor people to obtain if growth occurs where they are located [1]. Therefore, the revolution in the farming technology is of great significance to resolving the poverty and low income problem of many agriculture populations in China.

However, the under-developed agriculture technologies contributed to a large amount of resources waste and extra expenditure. In China, the government spent around 6 billion RMB into the energy supply and the annual water consumption of agriculture is 60% to 70% [2]. It is possible to largely reduce the production cost and raise the yield of agriculture to a brand new level, by deploying up-to-date technologies.

Precision agriculture is the application of technologies and principles to manage spatial and temporal variability associated with all aspects of agricultural production for the purpose of improving crop performance and environmental quality. Success in precision agriculture is related to how well it can be applied to assess, manage, and evaluate the space-time continuum in crop production [3].

Therefore, our motivation of the project is that, by constructing a farm management system, supported by carefully designed database, pleasingly simple and friendly user interface and high accuracy data analytic algorithms, which provides efficient and powerful services such as granular control, farming decision suggestion, data visualization etc.

Our group conducted our system development under the assumption of a intelligent agriculture company, which is offering modern solution to the quantified and specified management of agriculture works. This is an agriculture internet of things system which integrated multi-technologies including, sensoring, cloud services, database services, web services and learning algorithms. In the remaining part of the report, the contents are mainly focusing on database development (in general divided into system account, farm information and sensoring data), implementation of the frontend services and data analyzing algorithms.

Database Design and Implementation

2.1 Database Design

2.1.1 ER Diagram

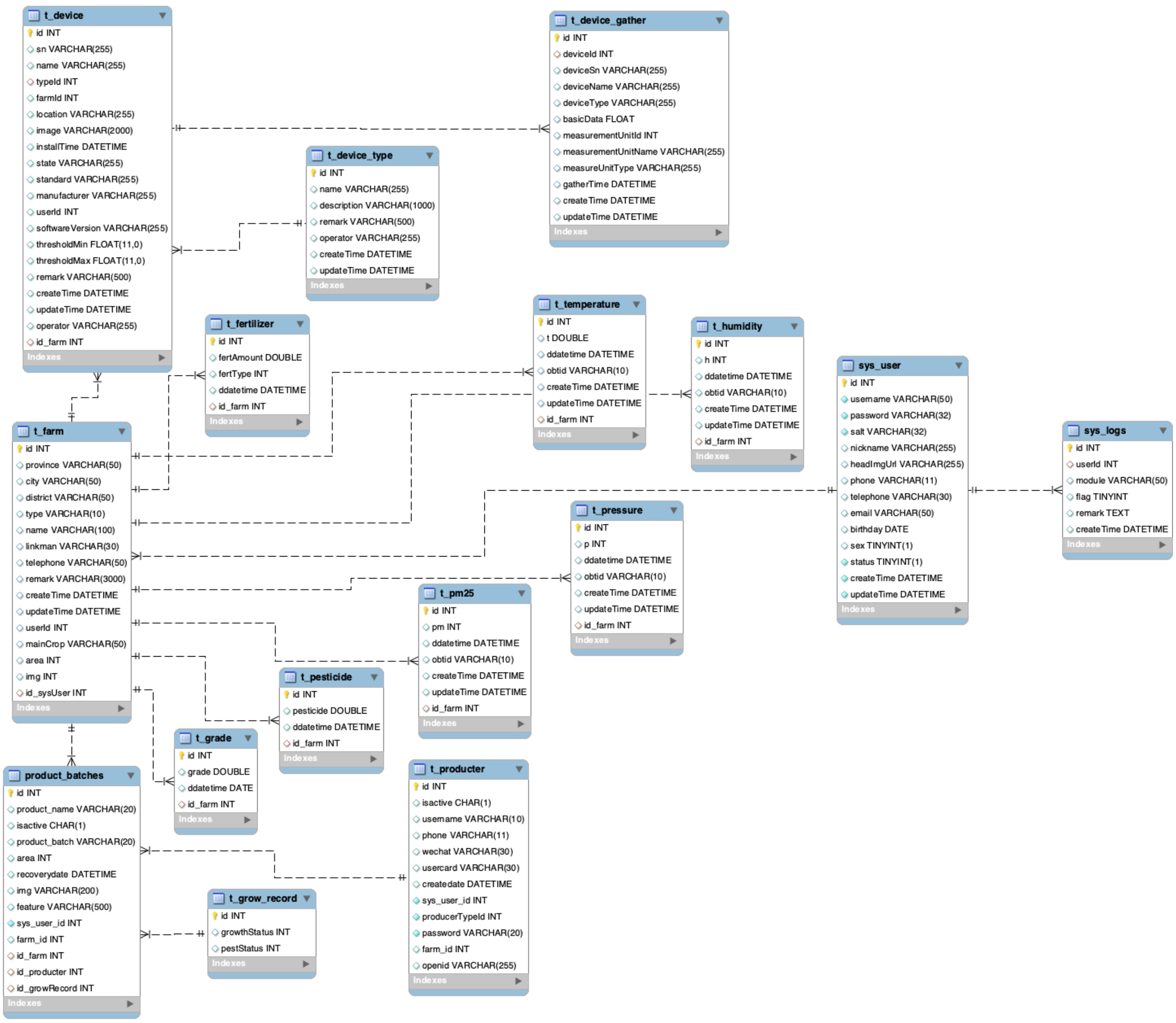


Figure 2-1: ER Diagram

2.1.2 Relational Schema

t_farm

id	province	city	district	type	name	linkman	telephone	remark	createTime	updateTime	userId	mainCrop	area	img	id_sysUser

t_device

id	sn	name	typeId	farmId	location	image	installTime	state	standard	manufacturer	userId
softwareVersion	thresholdMin	thresholdMax	remark	createTime	updateTime	operator	id_farm				

t_device_type

id	name	description	remark	operator	createTime	updateTime

t_device_gather

id	deviceId	deviceSn	deviceName	deviceType	basicData	measurementUnitId
measurementUnitName	measurementUnitType	gatherTime	createTime	updateTime		

sys_user

id	username	password	salt	nickname	headImgUrl	phone	telephone	email	birthday	sex	status	createTime	updateTime

sys_logs

id	userId	module	flag	remark	createTime

product_batches

id	product_name	isactive	product_batch	area	recoverydata	img	feature	sys_user_id	id_farm	id_producter	id_growRecord

t_producter

id	isactive	username	phone	wechat	usercard	createTable	sys_user_id	producerTypeId	password	farm_id	openid

t_grow_record

id	growthStatus	pestStatus

t_fertilizer

id	fertAmount	fertType	ddatetime	id_farm

t_temperature

id	t	ddatetime	obtid	createTime	updateTime	id_farm

t_humidity

id	h	ddatetime	obtid	createTime	updateTime	id_farm

t_pressure

id	p	ddatetime	obtid	createTime	updateTime	id_farm

t_pm25

id	pm	ddatetime	obtid	createTime	updateTime	id_farm

t_pesticide

id	pesticide	ddatetime	id_farm

t_grade

id	grade	ddatetime	id_farm

2.1.3 Assumptions

In order to find a cut into such a macroscopical project topic, we made the following assumptions to reduce the initial development complexity.

- The important factors that affects the growth of the plants, such as temperature, humidity, pressure, pm25, fertilizer, pesticide, are in identical level, if given that the farm is fixed.
- The farm manager and the product producer are different person. Manager is responsible for the farming jobs while the producer is focusing on the selling and transportation.
- The batch number is related to the producer. Each producer may produce overlapping batch number, while a single producer will make all their product batch numbers different.
- The data collected by the devices is extremely large in size, has meaningful information hidden and may contain many fluctuations.

2.1.4 Interpretation

The entities of our database can be decomposed into four main categories, including device, user, raw data and processed data. The following are their classification.

- Device: **t_device**, **t_device_type**
- User: **t_farm**, **sys_user**, **sys_logs**
- Raw Data: **t_device_gather**, **t_producter**, **product_batches**, **t_grow_record**
- Processed Data: **t_temperature**, **t_pm25**, **t_pressure**, **t_humidity**, **t_fertilizer**, **t_pesticide**, **t_grade**

For the device type, the entities are used to store the information of each single devices, located in each farms. Taking the **t_device** table as an example, with a surrogate id as the primary key, the table stores the name of the device, type of the device, location of the device, its image etc. All this information needed to identify a device participated in the agriculture internet of things system are kept in the device entities. Moreover, by adding the id_farm attribute into the t_device table as a foreign key, the device tables are linked with the farm entity. Therefore, given a device we can find the farm it belongs to, while given a farm, the devices it holds can also be found.

Considering the user category, the tables it contains are responsible to store the user's personal information and personalized system data. The **sys_user** table is straight foarward to understand, within which the user identification data are stored, such as username, password, phone etc. The **sys_log** entity is basically serving as a storage of the system tracker. It keeps the system logs and system outputs. For example, the flag and createTime attributes may indicate the fact that at which specific time the system raised a crash flag, and what is the supplementary information (remark) of this crush.

Things is a bit more complex in the set of raw data tables. For this type of entities, their similarity is that they all contains the first-hand data collected directly. The **t_device_gather** table, which is the most complicated entity among the set, holds the statistics collected from all different end devices, including different kinds of sensors, drones, local server etc. Therefore, the meaning of each line of data could be different. For example, the deviceType and measurementUnitType attributes will determine the interpretation of the collected basicData attributea. The database system is design to be this since the first hand raw data is important for our system to detect possible pest, misuse of fertilizer, incorrect watering etc. This data is then processed by the backend central server, where our prediction algorithms is constantly running at, and then sorted into the fourth category of entities, the processed data tables.

The last type of entities is the processed data entities. In this tables, the entries has been processed and selected by the server. Therefore, the data stored in these tables are precise, formatted and ready to be used to perform data visualization, data mining etc. By organizing them into these independent tables is also beneficial to the system efficiency, by lowering the searching overhead, which will be discussed in more detail in the next section.

2.2 Database Optimization

2.2.1 Index

There are some entities that are usually searched. In case of continuous traversal, the following index could be created:

```
CREATE INDEX Pos ON t_farm (province,city);

CREATE INDEX Crop ON t_farm (mainCrop);

CREATE INDEX Dev ON t_device (farmId, location);

```

```sql
CREATE INDEX User ON sys_user (username);

CREATE INDEX Prod ON product_batches (product_batch);

CREATE INDEX Gather ON t_device_gather (gathertim);
```

Pos is used for searching farms in a specific city.
Crop is used for searching farms growing a specific kind of crop.
Dev is used for searching devices in a specific farm or location.
User is used for searching username for users' login.
Prod is used for searching features of a specific product batch.
Gather is used for searching gathered data in a specific time period.

2.2.2 Normal Form

Most of schemas are in the third normal form. For example, in the schema t_temperature:

<u>id</u>	t	ddatetime	obtid	createTime	updateTime	id_farm

All the secondary keys depend and only depend on the primary key ”id”. However, the sys_user schema is not in 3NF; its primary key should be “username” instead of “id”. We made such a compromise as the size of “username” column is too large and will lead to performance degradation.

2.2.3 Two-stage Data

In this database raw data and processed data are both stored in different schemas. Such designation is aimed for a unified data process: the devices do not need to store data directly in different schemas; a unified operation could make it faster to categorize data into different schemas.

2.2.4 Further Optimization

Some optimization could be implemented by altering the data structure of database. For example, almost always continuous, the ”id” series could be stored in an expandible array with corresponding address; meanwhile, the ”username” and ”password” columns in sys_user schema could be hashed to reduce length and improve security.

2.3 UI Implementation

We provide a website for users to get access to the data in the database interactively. The website is built based on HTML, JS, and CSS for the frontend and PHP, MYSQL for the backend. We also use a package called Echarts to design our charts on the webpage.The whole structure is shown below:

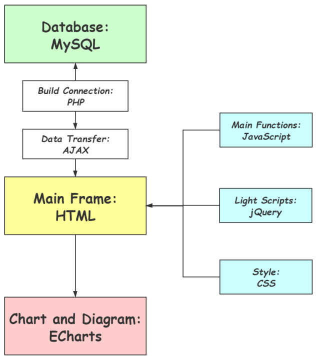


Figure 2-2: UI Framework

2.3.1 Register and Login

To access our database, we need users to register/log in to our website. These function is achieve by simple SQLs.

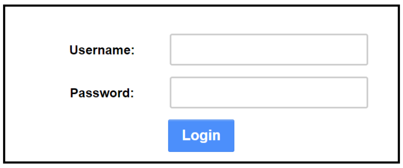
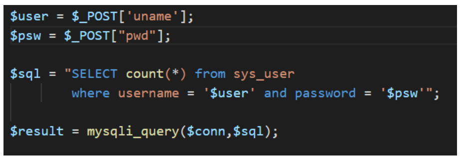


Figure 2-3: Log in page



```
$user = $_POST['uname'];
$psw = $_POST["pwd"];

$sql = "SELECT count(*) from sys_user
        where username = '$user' and password = '$psw'";

$result = mysqli_query($conn,$sql);
```

Figure 2-4: SQL for Log in

2.3.2 Farm Information Display

In our assumption, a farmer owns only one farm. Therefore, after log in our web-page, we will display the farm information he owns by using the simple SQL query.

```
Select t_farm.name, t_farm.id, t_farm.linkman, t_farm.telephone,
          t_farm.province, t_farm.city, t_farm.district
from t_farm, sys_user
where t_farm.userId = sys_user.id
and t_farm.userId = 1;
```

Figure 2-5: Daily information query in web-page

2.3.3 Daily Information Display

Farmers need to know the detailed data to understand the environment information on each day. In this part, we provide the farmer a quick method to query the information of the specific factors on a certain day. The information will be displayed in the report table, the UI of this function is shown below:

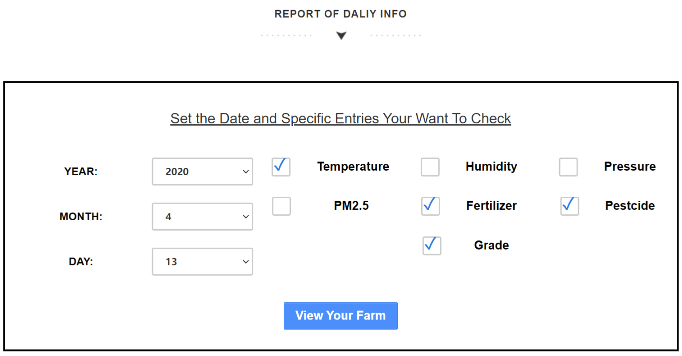
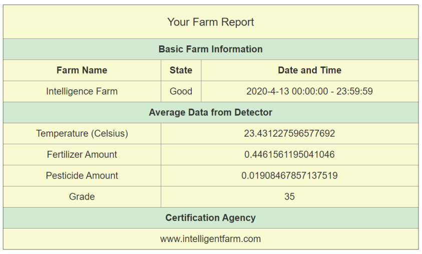
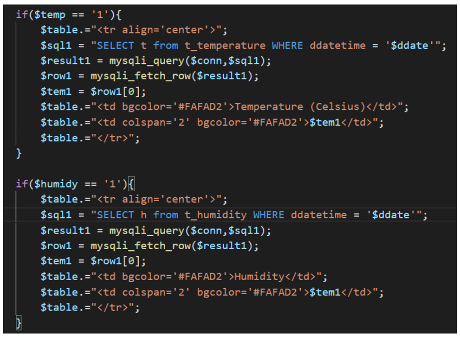


Figure 2-6: Daily information query in web-page



Your Farm Report		
Basic Farm Information		
Farm Name	State	Date and Time
Intelligence Farm	Good	2020-4-13 00:00:00 - 23:59:59
Average Data from Detector		
Temperature (Celsius)	23.431227596577692	
Fertilizer Amount	0.4461561195041046	
Pesticide Amount	0.01908467857137519	
Grade	35	
Certification Agency		
www.intelligentfarm.com		

Figure 2-7: UI for the report



```php
if($temp == '1'){
    $table.="<tr align='center'>";
    $sql1 = "SELECT t from t_temperature WHERE ddatetime = '$ddate'";
    $result1 = mysqli_query($conn,$sql1);
    $row1 = mysqli_fetch_row($result1);
    $tem1 = $row1[0];
    $table.="<td bgcolor='#FAFAD2'>Temperature (Celsius)</td>";
    $table.="<td colspan='2' bgcolor='#FAFAD2'>$tem1</td>";
    $table.="</tr>";
}

if($humidy == '1'){
    $table.="<tr align='center'>";
    $sql1 = "SELECT h from t_humidity WHERE ddatetime = '$ddate'";
    $result1 = mysqli_query($conn,$sql1);
    $row1 = mysqli_fetch_row($result1);
    $tem1 = $row1[0];
    $table.="<td bgcolor='#FAFAD2'>Humidity</td>";
    $table.="<td colspan='2' bgcolor='#FAFAD2'>$tem1</td>";
    $table.="</tr>";
}
```

Figure 2-8: Some of the queries for this function

2.3.4 Charts of the Annual Data

In this part, we visualize our database to find the inner correlation. We connect Echarts' API to our database and display the charts to the user.

The first figure displayed is the annual information, each dot is the average value of an event on one day, and this graph shows the tendency of the data. We also generated a heat map for farmers to help him find the relationship between the environment factors and the scores of the farm. Large rectangle means the product performances under that condition is better, small rectangle means the performances is relatively low. The color of each rectangle represents the event (such as fertilizer usage) level.

2.3.5 Data Visualization

In our web-page, we also provide a tools for users to customize the data visualization of his database. In this part, we connect Echarts to our database and user can customize each axis of his chart. Some examples are shown below.

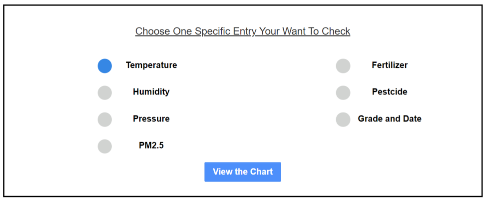


Figure 2-9: User Choose the Event

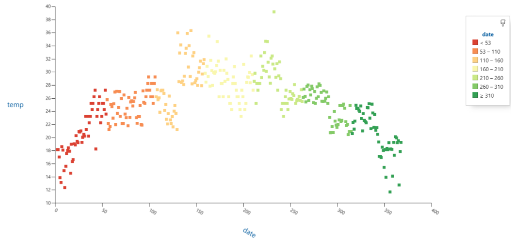


Figure 2-10: Scatter Diagram for Temp

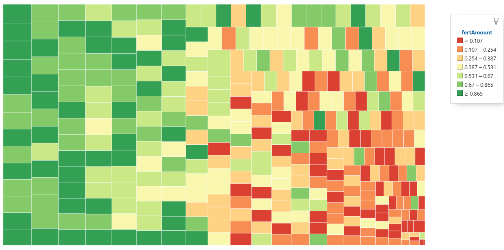


Figure 2-11: Heat Map for Fertilizer

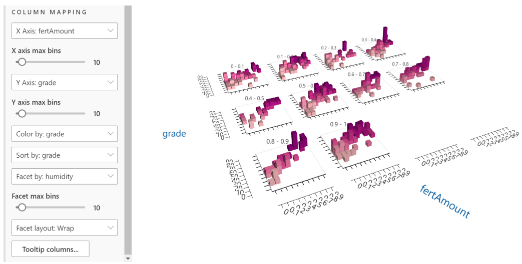


Figure 2-12: Customized Database Visualization

2.3.6 Other Useful SQL Queries

Besides of the queries that have been embedded into our website, we also provide some useful queries for users to implement.

```sql
--Some Other Useful SQL Queries Provided

--See all the product performances in order
select * from t_grade order by grade desc;

--See the date of the farm with grade 63
select ddatetime from t_grade where grade = 63;

--find the fertilizer use level when the grade is 100
select fertAmount from t_fertilizer, t_grade
where t_fertilizer.ddatetime = t_grade.ddatetime
and t_grade.grade = 100;

--find the information of the device of a farm owned by a farmer
select t_device.name from t_device,t_farm, sys_user
where t_device.id_farm = t_farm.id and sys_user.id = t_farm.id_sysUser
and sys_user.name = 'Deng Mi';
```

Data Mining

From the heat map we generated above, we discovered that there may be some correlation between the product performances of a farm and some environment values for example the fertilizers and pesticides. To dig more into this, we calculate the correlation coefficient r for each environment variables, the sample codes are shown below:

```
drop table if exists sample;
create table sample( x float not null, y float not null );
insert into sample select grade, h from t_grade natural join t_humidity;

select @ax := avg(x),
       @ay := avg(y),
       @div := (stddev_samp(x) * stddev_samp(y))
from sample;

select sum( ( x - @ax ) * (y - @ay) ) / ((count(x) -1) * @div) as "r for
    Humidityt_farm" from sample;
```

And the result is:

Variable	r
Fertilizers	0.758
Pesticides	0.494
Temperature	0.276
Pressure	0.022
PM2.5	0.074
Humidity	0

By correlation coefficient analysis, we suggests that a farmer should put his eyes more on the usage of pesticides and fertilizers in order to get good product performances.

However, this is actually an ethical issue. Because pesticides and fertilizers are often related to pollution and healthy problems on humans. And recently, there are many debates about the negative impacts of overusing the pesticides and fertilizers on the environment and human beings. For example, Stanford has a research on the "Overuse of Fertilizers and Pesticides in China".

Therefore, we provide farmers with a function to control the usage of pesticides and fertilizers based on AI. Basically, a farmer only needs to input the temperature, humidity etc. and we will provide him with a reasonable use level of pesticides and fertilizers.

To implement this function, we use a three-layer Deep Neuron Network. The training dataset is all the data from the original database which has a product performance grade over 80. Basically, our loss function is to minimize the cost and maximize the performance, we also refer to the global standard and set two constraints on the maximum usage of the pesticide and the fertilizer. This function is embedded into our website by JS to import python and we use pytorch to do the DNN.

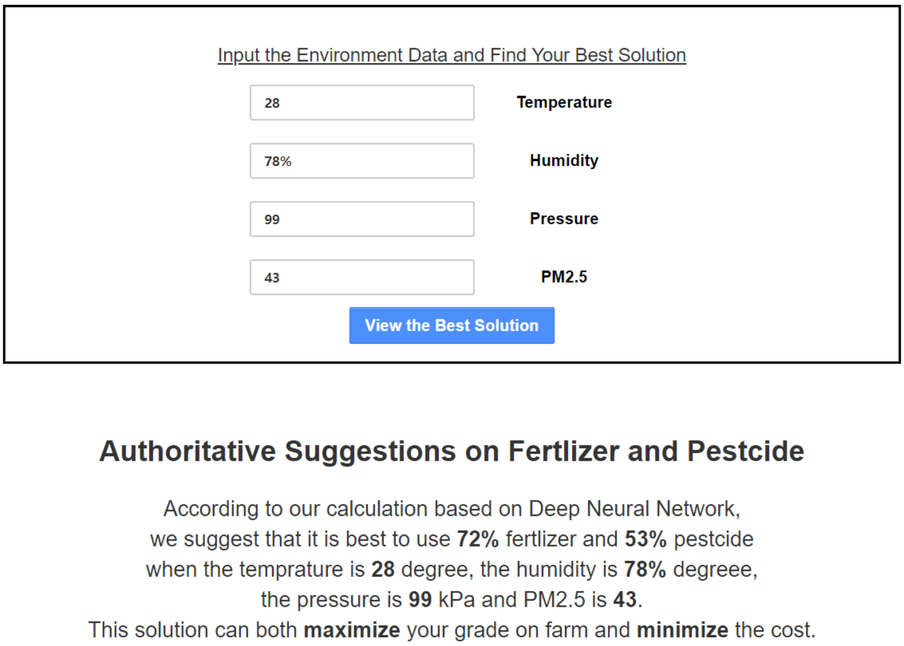


Figure 3-1: Customized Database Visualization

Conclusion and Self-evaluation

In our project, first we construct the database to store all the information of a farm. Then we do the database optimization and obtain the current database schema. We also build a website for farmers and provide many useful functions, including information query and customized data visualization. Moreover, based on data mining, we give farmers reasonable suggestions on the usage of pesticides and fertilizers.

In conclusion, we build an agriculture internet of things system which integrated multi-technologies including, sensoring, cloud services, database services, web services, data visualization, data analysis and learning algorithms. This system aims at assisting the farm managers controlling the overall running of their farms and making decisions. With the system, the farming job will become nice and handy.

4.1 Contribution of each member

4.1.1 Zhixuan LIU (118010202)

UI design, SQL and UI connection, function implementations using SQL queries, data exporting, data analysis.

4.1.2 Yuncong CUI (118010045)

UI Implement and I/O API Design of UI

4.1.3 Run LIU (118010188)

Website environment setting, data generating and loading, data mining and data analysis.

4.1.4 Jiarui LI(118020299)

Project idea, Database design

4.1.5 Dongyang WU (118010324)

Project idea, Database optimization

References

[1] Luc J Christiaensen. *Down to earth: agriculture and poverty reduction in Africa*. World Bank Publications, 2007.

[2] *Ministry of Agriculture andRural Affairs of the People's Republic of China*. URL: http://www.moa.gov.cn.

[3] Francis J. Pierce and Peter Nowak. "Aspects of Precision Agriculture". In: ed. by Donald L. Sparks. Vol. 67. Advances in Agronomy. Academic Press, 1999, pp. 1–85. DOI: https://doi.org/10.1016/S0065-2113(08)60513-1. URL: https://www.sciencedirect.com/science/article/pii/S0065211308605131.