FarmResult®
managementsystems

Internship Project – PoultryResult

The Development of Poultry Farm Management

Cross Platform Mobile Application using Google Flutter

Name : Danniar Reza Firdausy / s2337630
danniarrezafirdausy@student.utwente.nl

Company Supervisor : Richard ten Cate
UT Supervisor : Maya Daneva

Date : 15/07/2020 – 31/10/2020

Table of Contents

1. Background

1.1. Company Description

FarmResult B.V. is an Information Technology & Service provider company located in Wierden, the Netherlands, and currently operational in Europe, Latin, South America and Indonesia. Founded in 2011, FarmResult provides an independent data platform and delivers cloud-based livestock farm management systems focused on integrating sensor data from multiple farm houses and locations with supply chain data.

The company delivers PoultryResult, a management system designed specifically to support farmers in managing poultry (broiler and layer) farms and its supply chain operations, by structurally aggregating real-time farm data and integrating it with supply chain data. PoultryResult helps farmers to gain instant insights towards their farms, increase control, reduce risks, optimize productivity and animal health, reduce costs/increase profit, make farm management processes easier and save time. This solution also aids individual farmers and large integrators in their decision making by benchmarking results and data analytics through, what the company calls, Fact-Based Farm Management.

The company also delivers PigResult, which is a management system developed for the grower to finisher stage in the pig lifecycle, offering farmers the ability to control all aggregated data in an overview within one solution. Specific for trial farms the trial system ResearchResult is developed. ResearchResult assists research farms in data acquisition, data storage and research comparisons over different trials.

1.2. Fact-Based Farming

FarmResult emphasizes on the value of Fact-Based Farming, which is to make an optimal usage of data that is present in the farms to improve technical, operational and financial processes. This is enabled, as stated previously, by aggregating real-time farm data and integrating it with supply chain data. This real-time farm data which comprises data such as water, weight, feed, relative humidity, indoor climate, outdoor climate, etc., are automatically gathered by IoT sensors installed in each house location. These IoT sensors gather farm data every certain period of time, calculate the values in the company's edge computer FarmBox to standardize the format, then send the standardized data to the company's IoTCloud database. While this IoTCloud database stores a lower level granularized data, the higher granularity level version of the farm data (aggregated or latest value) are being synchronized to the PoultryResult database every certain period of time.

On the other hand, other types of farm data which are regarded as crucial and hold a high value in improving the performance output of individual farms are also present. These essential farm data consist of information regarding the animal mortality, weight, water consumption, feed consumption, additives and medications usage along with the egg production specifically for the Layers farm. However, these types of data are mostly, at the present time, hard to be automatically and periodically measured using IoT sensors. Therefore, manual field observation and data entry is still the optimal means of data gathering in this case. In order to facilitate the farmers to do this process, FarmResult has developed and been providing their clients with PoultryResult mobile app.

1.3. The Assignment

The PoultryResult solution is a farm management system that uses automated data entry as much as possible by connecting the system to partners in the supply chain along with establishing connection to IoT sensors inside the poultry houses such as to gather (and control) wind speed, water use, weight, feed, CO2, NH3, etc. The solution provides 2 interfaces to the users, one is in the form of a web-based application that offers a more holistic view and control over the farm data. However, a smaller but important set of information needs to be observed and gathered on-field by the farmers and using the same full-sized web-based application will be too inconvenient. Hence, in this case, a smart phone mobile application is provided. This smaller information comprises the health status of the animals, daily average weight, daily water consumptions, daily feed consumptions, used additives like Vitamin C, used medications, the amount of eggs productions and the average eggs weight. All of this information will be entered inside the broiler or layer house, where internet connection is mostly scarce. This implies that the mobile app is expected to work not only when the phone is online, but also when offline. Moreover, through feedback that the company received from their clients and the future goal that the company wants to achieve, the capability of using the device's hardware such as the camera to take a photo or video for observation is desired.

The current version of the mobile app has not been able to provide the last two requirements (among others) previously mentioned so far, since it was developed using a Java-based framework which mostly intended for web application development. At the same time, a cross-platform development framework called Flutter has been created and maintained by Google and has been used by thousands of developers in organizations across the world for production apps since its launch in 2017 (Payne, 2019). Knowing this, the company desires to revamp their existing mobile app solution to a new one that employs Flutter framework. This brings to the main assignment of this internship project, to develop a mobile app using Google's Flutter framework in which the final outcome of this project will be used to replace the current ''PoultryResult app". The final goal of this assignment is to evaluate Google Flutter as a development tool for this app requirement. However, as a complementary to the final goal, an investigation towards other development approaches is to be expected since there seems to be a trend between enterprises to consider using a low-code or no-code platform. Ultimately, at the end of this assignment, the advantages and disadvantages of the developed application using Flutter in comparison to the current web-based PoultryResult mobile app should be made clear. The first users of this to-be developed app will be the stakeholders in the company, while the end-users will be the farmers as the company's client.

2. Project Report

2.1. Management Summary

This report presents the development of a remodeled PoultryResult mobile application which is aimed to replace the current version of the company's solution for their poultry farmer clients. The assignment was conducted as part of an internship at FarmResult B.V. in Wierden, The Netherlands. FarmResult wishes to increase the quality of one of their solutions by introducing a new set of value-added features and functionalities that can bring extended benefits to their clients. This decision has been considered due to the perceived evolving business requirements, dynamic constraints in the real-world operational environment and the constructive feedback they have obtained from their clients so far. However, since remodeling has been decided and an evolving trend in the industry has been taken into account, multiple development approaches such as decisions between low-code and raw-code development are being considered.

The development of this new version of PoultryResult mobile application attempts to use both raw-code approach using Google Flutter as well as evaluating the possibilities in using low-code development approach supported by Mendix platform. It is discovered that even though Mendix enables quick and easy to use mobile apps development platform, the out-of-the-box components are still limited and not yet able to satisfy the desired PoultryResult's requirements. The development using Google Flutter on the other hand, has proved that it is able to fulfill all of the defined mobile app requirements. This finding can be brought to light due to the fact that the framework has provided a complete set of widgets and built-in functions necessary to support and realize the requirements of the mobile application. This advantage also enables the introduction of a new set of value-added functionalities, such as the implementation of the daily observation measurements dashboard and KPIs, to this new version of PoultryResult mobile application. Furthermore, due to the low learning curve of the framework caused by reliance only on a single programming language and the availability of comprehensive documentations, the concerns associated with the technical requirements of developing using raw-code approach can be reduced.

By the end of this project, a Usability Testing was carried out in order to evaluate the delivered prototype of PoultryResult mobile application. The testing result has shown that the new mobile application has reached the effectiveness rate of 86% and overall relative efficiency of 88%, which are considered as above the average of the testing method's common result. In addition, it can be deduced that the participants are strongly agreed that the new version of the mobile application has provided an easy-to-use user interface and user experience indicated by the achievement of the satisfaction level of 87%. All things considered, the Google Flutter framework is a suitable development approach for the company's PoultryResult mobile application solution and further development of this project using Google Flutter is worth to be carried on in the future.

2.2. Introduction

PoultryResult solution has been delivered by FarmResult to their poultry farmer clients through 2 types of interfaces, one is in the form of a web-based application and the other one is in the form of mobile-based application. Though it is mobile-based application, but the technology used behind it is still using the same Java-based framework used for its web-based application development called Java Server Faces (JSF, recently renamed into Jakarta Server Faces) which simplifies construction of user interfaces (UI) mainly for server-based applications by using reusable

UI components in a page (Distante, Pedone, Rossi, & Cafora, 2007). The current version of the mobile app has been enabling farmers (or field operators specifically) as the mobile app users to view, register and update daily observations data that are essential for managing their poultry farm and only these simple activities are provided so far.

2.3. Problem Statement and Project Goals

The current version of the PoultryResult mobile app, using the previously mentioned technology, has not provided the latest requirements that the company has in mind. These latest requirements are mostly related with the access towards the mobile device's hardware. An example for this is to enable the field operator to take a picture of their daily observation, providing the farm manager the means to visually recognize the problem that the field operator registered, and since FarmResult has formulated a vision to employ machine learning methods in the future for instance to recognize the size and volume of the chicken based on the image uploaded, then to store the images as potential data training will be a critical mission. Another example that is related to the device's hardware is the ability to detect the device's internet connection and capability to operate under offline connection. This will be translated into the requirement of the ability to employ a local database inside the mobile app. Furthermore, user interface and user experience were not considered as the top priority of the development of the current version, since it was developed to enable their clients to do daily observations using the mobile app version of their solution, then it is expected from the new development that it will have an improved user interface and user experience.

Having the previously stated requirements in mind and the consideration to use the latest technology maintained by the current leading technology company, the Google Flutter is being chosen to be used for this project. This decision is also being considered since the company has done a separate project at the start of the year 2020 which was made successful by using the Flutter framework. Knowing this, it is the company's desire to revamp the current PoultryResult mobile application and evaluate if Google Flutter will be able to satisfy the requirements in mind. However, judging by the current trend of demand for mobile application development within the enterprise community, organizations in 2021 are believed to be considering in migrating their mobile development project towards low-code or no-code tools which enables general business users to easily create new apps or add features without necessarily being a specialist in raw-code development (Chang & Ko, 2017). Therefore, another development approach that is becoming the latest trend in the industry should also be investigated in order to be considered as an alternative solution. Finally, by the end of the project, the newly developed PoultryResult prototype should be able to be tested and evaluated to assess the advantages and the disadvantages of it developed using Google Flutter. Based on this description, the project goals are formulated as:

1. Investigate another mobile app development approach that proposes the potential solution as an alternative.
2. Develop a prototype mobile app using Google Flutter in which the final outcome of the development will be used to replace the current PoultryResult mobile app.
3. Test and evaluate the advantages and the disadvantages of the developed prototype in comparison with the current PoultryResult mobile app.
4. Evaluate the advantages and the disadvantages of the PoultryResult mobile app development using Google Flutter.

2.4. Project Approach and Methods

2.4.1. Scrum – Agile Project Management

The approach used in this project is determined by the characteristics of the project and the duration of the internship contract itself. Since the urgency of the developed solution is to support FarmResult's capability in creating value for their farmer clients, optimum alignment between IT and the growing business needs, Scrum approach based on Agile Methodology has been chosen. This decision is also influenced by the short duration of the internship contract, which is a condition that the Scrum approach was designed to succeed (Srivastava, Bhardwaj, & Saraswat, 2017). Workflow of Scrum consists of close collaboration between the Scrum Team and the Product Owner over continuous iterations of the evolving developed software. The continuous iterations mentioned earlier, is known as the sprint and each sprint usually lasts for 1 to 3 weeks in duration. The task for a sprint is known by the name of a sprint backlog and it is derived from the overall requirements of the product called the product backlog which are determined by the product owner. The aim of the Scrum approach is that by the end of each sprint, the scrum team can deliver a potential shippable product that can be demonstrated to and reviewed by the product owner.

The Scrum Team in this project will consist of myself as the developer which is also responsible for managing the progress of the sprints, while the Product Owner will be the CEO of the company who's also responsible as the company supervisor of this internship assignment. Additional stakeholder involved in this project is the company's senior software developer, who is acting as the supervisor for the technology implementation and can be categorized as one of the Scrum Team in this project. Along with the role definition, product backlog is also necessary in order to set the scope of what should be done in each sprint and it will be further shown in detail in Section 2.5. The sprints however, its planning, execution and evaluation, will be discussed more in Section 2.7 Google Flutter Project Execution.

2.4.2. Mendix Platform

In relation with the first project goal of investigating an alternative approach for mobile app development, the Mendix platform has been chosen. This development platform is regarded as one of the leading low-code solutions recognized by analyst reports such as Gartner and large enterprise companies such as SAP and IBM (Mew & Field, 2018). Mendix platform enables (technical or non-technical) developers to build web applications in a complete package starting from the database model, back end services (business logic along with web services using REST API) as well as the front-end user interfaces. Following the high demand in the industry, Mendix has extended their platform to enable developers to develop mobile application solutions in the same low-code approach. Unlike their web application counterparts where the underlying technology uses common web technologies such as HTML, CSS and JavaScript, the Mendix mobile application are based on React Native which considered as a true native mobile apps development framework based on JavaScript and regarded as one of the most popular cross-platform mobile application development framework in the industry alongside Google Flutter and the others. Apart from this, Mendix has also been considered as the alternative since they offer the possibilities to automatically synchronize data between the mobile app and the back end in an offline-first architecture. However, the details regarding the actual execution, the advantages and the disadvantages of the low-code development experiment using Mendix will be further discussed in Section 2.6 Mendix Prototype.

2.4.3. Usability Testing

Finally, by the end of this project, Usability Testing will be used to evaluate the advantages and the disadvantages of the developed prototype compared to the existing PoultryResult mobile app. This decision is taken under the consideration of the different personas and demographics that the prospective users may possess. Under this dynamic condition, the ability of each user in interacting with the developed app, such as how fast and how easy it is for them to complete specific tasks, will be affected by the usability level of the technology in evaluation. This software testing methodology is also being considered for this project since the target users may expect the developed mobile app can be used as effectively and efficiently as possible without sacrificing the satisfaction of using it (Wicaksono, Firdausy, & Saputra, 2018).

Usability Testing can be performed by involving representative users to a testing session and then give them responsibility to do a set of representative tasks at the developed mobile app. Several metrics that will be captured are the measures of Effectiveness, Efficiency and Satisfaction. Effectiveness discusses the level of accuracy and completeness of the users to complete a certain set of tasks. The completeness level is represented using binary value '1' if the participant completes a task and '0' if they fail after a certain time limit. Then, the sum of this binary value will be divided over the total amount of task that should be done to obtain the Effectiveness value as shown in Formula 1.

$$Effectiveness = \frac{Amount\ of\ Task\ Completed}{Total\ Amount\ of\ Task\ Should\ be\ Done}\ x\ 100\% \quad (1)$$

Efficiency represents the time spent by the user in order to ensure the level of completeness and achievement of the goals which are represented in seconds or minutes. This metric can be obtained by measuring the overall relative efficiency which uses the ratio of the time taken by each participant who has succeeded in completing the task in comparison with the total time taken by all participants which will be represented using percentage as shown in Formula 2.

$$Overall\ Relative\ Efficiency = \frac{\sum_{j=1}^{R}\sum_{i=1}^{S} s_{ij} \times t_{ij}}{\sum_{j=1}^{R}\sum_{i=1}^{S} t_{ij}} \times 100\% \quad (2)$$

Satisfaction discusses the level of convenience and acceptance of the participant as they are using the developed mobile app. Satisfaction will be captured by using standardized satisfaction questionnaires given at the end of the testing session. The result of the questionnaires will be measured using Likert Scale from the scale of 1 to 5 as shown in Figure 1, representing Strongly Disagree to Strongly Agree respectively.

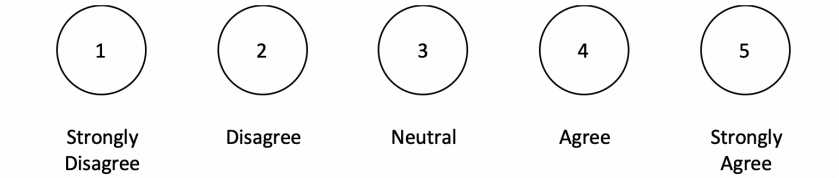

Figure 1 Likert Scale Example

Calculation of the Likert Scale results begins by giving the score weight to each answer, which "Strongly Agree" is given a score of 5, "Agree" is given a score of 4, "Neutral" is given a score of 3, "Disagree" is given a score of 2 and score 1 is given to "Strongly Disagree". Next is to multiply the number of answers from participants who answer on each answer scale point with the answers score weight to obtain the total answered score.

$$Total\ Answered\ Score = (n \times 5) + (n \times 4) + (n \times 3) + (n \times 2) + (n \times 1) \quad (3)$$

Further is to define the maximum score of the scale and then eventually calculating the index score percentage by dividing the total answered score with the maximum score that previously has been defined, ending with multiplying it with 100% as shown in Formula 3 and 4.

$$\text{Index Score } \% = \frac{Total\ Answered\ Score}{Maximum\ Score} \text{ x } 100\% \quad (4)$$

After the percentage of the index score has been obtained then that score can be compared with the defined percentage interval shown in the following Table 1 to interpret the index score meaning, which in this case the level of participants satisfaction in the usability testing context being carried out.

Table 1 Percentage Interval Score Interpretation

Percentage Interval	Score Interpretation
80 to 100%	Strongly Agree
60 to 79,99%	Agree
40 to 59,99%	Neutral
20 to 39,99%	Disagree
0 to 19,99%	Strongly Disagree

2.5. Product Backlog

In this section, the overall requirements of the product known as the product backlog are listed and organized into 3 categories for better requirement visibility and management. These categories consisted of the General Requirements, Animal Daily Observations and House Climate Observations. The following subsections will discuss each of the categories and each of the user stories will be coded to facilitate traceability.

2.5.1. General Requirements

General Requirements contains the user stories that apply to the general features, constraints,

business rules or application flows that are expected of the solution to be developed. In the following Table 2, these requirements will be listed as the user stories coded by the initials of GR (General Requirements), described in detail and decided upon their priority ranging from the scale of 1 to 5.

Table 2 General Requirements Product Backlog

Group	No.	User story	Detail	Priority
General	GR 1.	User (farmer) is able to enter their username and password to login into the application	Only user with registered account is able to access the application	1
	GR 2.	User is able to see and select their farm site location	Possibility of a company controlling multiple farm sites, with different types of farm i.e., Broiler or Layer	1
	GR 3.	User is able to access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	The Animal Daily Observations available for the user to access will only be of the active round	1
	GR 4.	User is able to access and manage House Climate Observations of each house in a round of the selected farm site location	The House Climate Observations of the houses will only be of the active round	3
	GR 5.	User still able to view, edit and delete Animal Daily Observations and House Climate Observations data while not connected to the Internet	- Internet connection inside farm house is mostly poor - The app should support local data storage and synchronize with backend service when connected with Internet	4
	GR 6.	User can include or exclude every measurement through the app	Impose user roles and responsibilities of which observations they can/have access to	5
	GR 7.	User wants to schedule the entry of missing data	Schedule whether it is once per day or once per week	5
	GR 8.	User is able to Logout from the application to terminate access	Enable the user to terminate access and probably to change user	1

2.5.2. Animal Daily Observations

Animal Daily Observations contains the user stories that satisfy one of the urgencies to develop and revamp the mobile version of PoultryResult in the first place. These user stories contain the daily observations of the animals will be gathered on-field by the farmers or field operators and it consists of the mortalities, average weight, water consumptions, feed consumptions, used additives & medications and the eggs productions for layer farms. Along with these, the capabilities to present the overview of the observation data visually into a graph and make use of the device's camera to take pictures are also desired. In the following Table 3, these requirements will be listed as the user stories coded by the initials of DO (Daily Observations), described in detail and decided upon their priority ranging from the scale of 1 to 5.

Table 3 Animal Daily Observations Product Backlog

Group	No.	User story	Detail	Priority
Mortality	DO 1.	User is able to manage daily animal mortality observations of the current round	Animal mortality consist of the number of animals dead, culled along with the remarks	1
	DO 1.1	User is able to see current round animal mortality of the same or another day	Consist of the list of mortality observation showing the amount of culling and death for specific date	1
	DO 1.2	User is able to enter new entry of the chicken flock's mortality observation – maximum 2 observations a day	Animal mortality consists of the amount of culling and death for specific house, round and date	1
	DO 1.3	User is able to edit entry of the chicken flock's mortality observation	Complements new data entry	1
	DO 1.4	User is able to delete entry of the chicken flock's mortality observation	Complements new data entry	1
Feed	DO 2.	User is able to manage daily animal feed observations of the current round	Animal feed consist of the feed type, amount and unit of measurement	1
	DO 2.1	User is able to see current round animal feed of the same or another day	Consist of the number of feed type, amount and unit of measurement	1
	DO 2.2	User is able to enter new entry of the animal feed	New batch entry consists of the number of feed type, amount and unit of measurement	1
	DO 2.3	User is able to edit entry of the animal feed	Complements new batch entry	1
	DO 2.4	User is able to delete entry of the animal feed	Complements new batch entry	1
	DO 2.5	User is able to copy previous entry when animal feed observation for current day is not yet present	Provide the user to do a quick data entry and minimum input effort	2
Weight	DO 3.	User is able to manage daily animal weight observations of the current round	Animal weight consist of the average animal weight in a particular house in current round	1
	DO 3.1	User is able to see current round animal weight of the same or another day	Consist of the average animal weight in a particular house in current round	1
	DO 3.2	User is able to enter new entry of the animal weight observation – once a day	Animal weight consist of the average animal weight in a particular house in current round	1
	DO 3.3	User is able to edit entry of the animal weight observation	Complements new data entry	1
	DO 3.4	User is able to delete entry of the animal weight observation	Complements new data entry	1
Water	DO 4.	User is able to manage daily animal water consumption observations of the current round	Animal water consumption consist of the amount of water	1
	DO 4.1	User is able to see current round animal water consumption of the	Consist of the amount of water consumption	1

		same or another day		
	DO 4.2	User is able to enter new entry of the animal water consumption observation – once a day	New water consumption observation entry consists of the amount of water	1
	DO 4.3	User is able to edit entry of the animal water consumption observation	Complements new batch entry	1
	DO 4.4	User is able to delete entry of the animal water consumption observation	Complements new batch entry	1
	DO 4.5	User is able to copy previous entry when animal water observation for current day is not yet present	Provide the user to do a quick data entry and minimum input effort	2
Additive	DO 5.	User is able to manage daily animal additive consumption observations of the current round	Animal additive consist of the additive article, amount and unit of measurement	1
	DO 5.1	User is able to see current round animal additive consumption of the same or another day	Consist of the number of the additive article, amount and unit of measurement	1
	DO 5.2	User is able to enter new entry of the animal additive consumption observation	New batch entry consists of the additive article, amount and unit of measurement	1
	DO 5.3	User is able to edit entry of the animal additive consumption observation	Complements new batch entry	1
	DO 5.4	User is able to delete entry of the animal additive consumption observation	Complements new batch entry	1
Vaccination	DO 6.	User is able to manage daily animal vaccination observations of the current round	Animal vaccination consist of the vaccination article, amount and unit of measurement	1
	DO 6.1	User is able to see current round animal vaccination of the same or another day	Consist of the number of the vaccination article, amount and unit of measurement	1
	DO 6.2	User is able to enter new entry of the animal vaccination observation	New batch entry consists of the vaccination article, amount and unit of measurement	1
	DO 6.3	User is able to edit entry of the animal vaccination observation	Complements new batch entry	1
	DO 6.4	User is able to delete entry of the animal vaccination observation	Complements new batch entry	1
Eggs	DO 7.	User is able to manage daily egg production observations of the current round – only when user is selecting Layer Farm	Animal egg production consist of the amount of first quality eggs, second quality eggs, ground eggs and the average egg weight	3
	DO 7.1	User is able to see current round egg production of the same or another day	Consist of the amount of first quality eggs, second quality eggs, ground eggs and the average egg weight	3
	DO 7.2	User is able to enter new entry of the egg production observation	New batch entry consists of the amount of first quality eggs, second quality eggs, ground eggs and the average egg weight	3
	DO 7.3	User is able to edit entry of the	Complements new batch entry	3

		egg production observation		
	DO 7.4	User is able to delete entry of the egg production observation	Complements new batch entry	3
Picture & Video	DO 8.1	User is able to upload picture of the chicken	To increase the data quality of the daily observation's record.	2
	DO 8.2	User is able to upload video footage of the situation	To increase the data quality of the daily observation's record.	5
Daily Observations Dashboard	DO 9.	User is able to see all relevant information about the animals in one or two combined graphs	To assess if the realized performance is going well as planned	3
	DO 9.1	User is able to see the animal mortality growth with the reasons	To assess if the realized performance is going well as planned	3
	DO 9.2	User is able to see the animal weight growth in a certain house	To assess if the realized performance is going well as planned	3
	DO 9.3	User is able to see the trend of the feed consumption amount for each feed type	To assess if the realized performance is going well as planned	3
	DO 9.4	User is able to see the trend of the water consumption amount	To assess if the realized performance is going well as planned	3
	DO 9.5	User is able to see the trend of the additive consumption amount for each article	To assess if the realized performance is going well as planned	3
	DO 9.6	User is able to see the trend of the vaccination/medication usage amount for each article	To assess if the realized performance is going well as planned	3
	DO 9.7	User wants to see a target value or plan for certain measures in a graph	To assess if the realized performance is going well as planned	3

2.5.3. House Climate Observations

House Climate Observations discusses the user stories that aim to facilitate visibility and control over the climate data of the poultry houses. As mentioned previously in Section 1.4, data regarding the climate conditions (such as CO2, NH3 and relative humidity) are mostly gathered by utilizing IoT sensors installed inside the houses and stored into the IoT Cloud periodically. However, in case of hardware or network failure, the farmers also expect the ability to perform manual data entry process so that data accuracy and completeness can be improved. Apart from this ability, the representation of the gathered climate data into a graph along with the provision of the KPIs value regarding the rearing performance of the house in the form of ratio value are also expected. In the following Table 4, these requirements will be listed as the user stories coded by the initials of CO (Climate Observations), described in detail and decided upon their priority ranging from the scale of 1 to 5.

Table 4 House Climate Observations Product Backlog

Group	No.	User story	Detail	Priority
House	CO 1.	User is able to see the measurements of	Measurements data are gathered	3

Climate Dashboard		the company's farm houses	using sensors and desired to be displayed in some overview graphs	
	CO 1.1	User is able to see temperature measurement of the house	Temperature inside and outside of the house	3
	CO 1.2	User is able to see the relative humidity (%) of the house	Relative humidity of the house	3
	CO 1.3	User is able to see the ventilation speed of the house	Ventilation speed of the house	3
	CO 1.4	User is able to see the amount of gas used for heating	Amount of gas used for heating	3
	CO 1.5	User is able to see the reading of CO2 level in ppm unit	The reading of CO2 level in ppm unit	3
	CO 1.6	User is able to see the reading of NH3 level in ppm unit	The reading of NH3 level in ppm unit	3
	CO 1.7	User wants to see a target value or plan for certain measures in a graph	To assess if the observed data complies the norm or standards	3
	CO 1.8	User wants to see some KPIs value regarding the rearing performance of the house	To assess if the realized performance KPIs (in ratio) is going well as planned	3
Climate Data Intervention	CO 2.	User is able to manage manual house climate observation of the current round	At the moment, climate data of CO2, Relative Humidity and NH3 are essential	3
	CO 2.1	User is able to see manual house climate observations	Consist of the level of CO2, Relative Humidity and NH3	3
	CO 2.2	User is able to enter new manual house climate observations	New observation entry consists of CO2, Relative Humidity and NH3	3
	CO 2.3	User is able to edit entry of the manual house climate observations	Complements new observation entry	3
	CO 2.4	User is able to delete entry of the manual house climate observations	Complements new observation entry	3

2.6. Mendix Prototype

As soon as the product backlog has been established, the experiment to develop mobile applications using Mendix can start to be carried out. This defined product backlog will serve as the design and implementation guidelines for the developed prototype using a low-code approach. In order to evaluate the feasibility of this approach, only a selection of user stories from General Requirements and Animal Daily Observations will be implemented.

2.6.1. Design Phase

The development using this approach is started by first determining the user stories to be implemented. The selected user stories based on the product backlog in the previous chapter are shown in the following Table 5.

Table 5 Selected User Stories for Mendix Implementation

Group	No.	User story	Detail
General	GR 1.	User (farmer) is able to enter their username and password to login into the application	Only user with registered account is able to access the application
	GR 2.	User is able to see and select their farm site location	Possibility of a company controlling multiple farm sites, with different types of farm i.e., Broiler or Layer
	GR 3.	User is able to access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	The Animal Daily Observations available for the user to access will only be of the active round
	GR 5.	User still able to view, edit and delete Animal Daily Observations and House Climate Observations data while not connected to the Internet	- Internet connection inside farm house is mostly poor - The app should support local data storage and synchronize with backend service when connected with Internet
	GR 8.	User is able to Logout from the application to terminate access	Enable the user to terminate access and probably to change user
Mortality	DO 1.	User is able to manage daily animal mortality observations of the current round	Animal mortality consist of the number of animals dead, culled along with the remarks
	DO 1.1	User is able to see current round animal mortality of the same or another day	Consist of the list of mortality observation showing the amount of culling and death for specific date
	DO 1.2	User is able to enter new entry of the chicken flock's mortality observation – maximum 2 observations a day	Animal mortality consists of the amount of culling and death for specific house, round and date
	DO 1.3	User is able to edit entry of the chicken flock's mortality observation	Complements new data entry
	DO 1.4	User is able to delete entry of the chicken flock's mortality observation	Complements new data entry
Feed	DO 2.	User is able to manage daily animal feed observations of the current round	Animal feed consist of the feed type, amount and unit of measurement
	DO 2.1	User is able to see current round animal feed of the same or another day	Consist of the number of feed type, amount and unit of measurement
	DO 2.2	User is able to enter new entry of the animal feed	New batch entry consists of the number of feed type, amount and unit of measurement
	DO 2.3	User is able to edit entry of the animal feed	Complements new batch entry
	DO 2.4	User is able to delete entry of the animal feed	Complements new batch entry
Weight	DO 3.	User is able to manage daily animal weight observations of the current round	Animal weight consist of the average animal weight in a particular house in current round
	DO 3.1	User is able to see current round animal weight of the same or another day	Consist of the average animal weight in a particular house in current round
	DO 3.2	User is able to enter new entry of the animal weight observation –once a day	Animal weight consist of the average animal weight in a particular house in current round
	DO 3.3	User is able to edit entry of the animal weight observation	Complements new data entry
	DO 3.4	User is able to delete entry of the animal weight observation	Complements new data entry
Water	DO 4.	User is able to manage daily animal water	Animal water consumption consist of the

		consumption observations of the current round	amount of water
	DO 4.1	User is able to see current round animal water consumption of the same or another day	Consist of the amount of water consumption
	DO 4.2	User is able to enter new entry of the animal water consumption observation – once a day	New water consumption observation entry consists of the amount of water
	DO 4.3	User is able to edit entry of the animal water consumption observation	Complements new batch entry
	DO 4.4	User is able to delete entry of the animal water consumption observation	Complements new batch entry
Additive	DO 5.	User is able to manage daily animal additive consumption observations of the current round	Animal additive consist of the additive article, amount and unit of measurement
	DO 5.1	User is able to see current round animal additive consumption of the same or another day	Consist of the number of the additive article, amount and unit of measurement
	DO 5.2	User is able to enter new entry of the animal additive consumption observation	New batch entry consists of the additive article, amount and unit of measurement
	DO 5.3	User is able to edit entry of the animal additive consumption observation	Complements new batch entry
	DO 5.4	User is able to delete entry of the animal additive consumption observation	Complements new batch entry
Vaccination	DO 6.	User is able to manage daily animal vaccination observations of the current round	Animal vaccination consist of the vaccination article, amount and unit of measurement
	DO 6.1	User is able to see current round animal vaccination of the same or another day	Consist of the number of the vaccination article, amount and unit of measurement
	DO 6.2	User is able to enter new entry of the animal vaccination observation	New batch entry consists of the vaccination article, amount and unit of measurement
	DO 6.3	User is able to edit entry of the animal vaccination observation	Complements new batch entry
	DO 6.4	User is able to delete entry of the animal vaccination observation	Complements new batch entry
Picture & Video	DO 8.1	User is able to upload picture of the chicken	To increase the data quality of the daily observation's record.
Daily Observations Dashboard	DO 9.	User is able to see all relevant information about the animals in one or two combined graphs	To assess if the realized performance is going well as planned

Having already set out the selected user stories as above, the development process can be followed by designing and implementing the data model for data storage. In Mendix, this process can be done in their Domain Model in a drag-n-drop manner. This way the data entities along with their attributes and relations between entities can be implemented as closely as possible with the defined selected user stories earlier, as shown in Figure 2 below. From the domain model, it is shown that the implementation of entities Location, House and Round along with their relations with each other are the realization of user stories of GR 1, GR 2 and GR 3. Moreover, user stories regarding animal daily observations starting from DO 1 until DO 6 are facilitated by the implementations of the corresponding entities surrounding the Round entity.

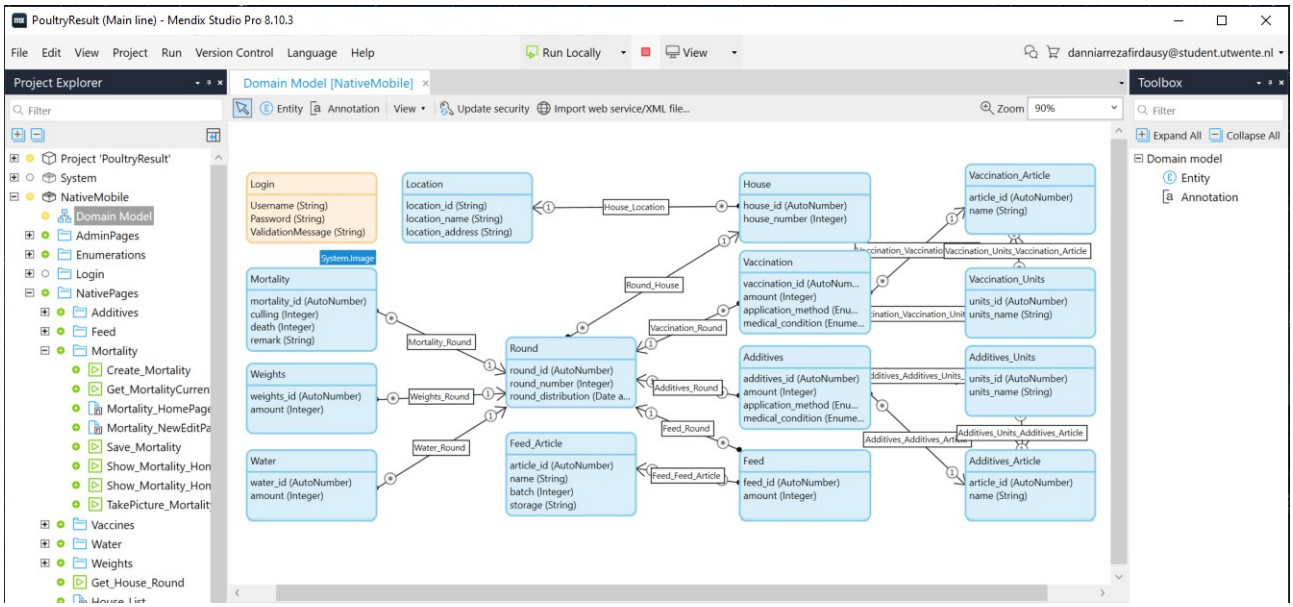

Figure 2 Mendix Prototype Domain Model

2.6.2. Execution Phase

Following the domain model implementation, the phase to execute the application development takes place. In order to start working on the native mobile application, the web version of the application should be developed first. This is to make sure that operating the data in the back end can be performed. Fig 3 shows a glimpse of the process of application's user interface (UI) development using Mendix in relevance towards the low-code approach.

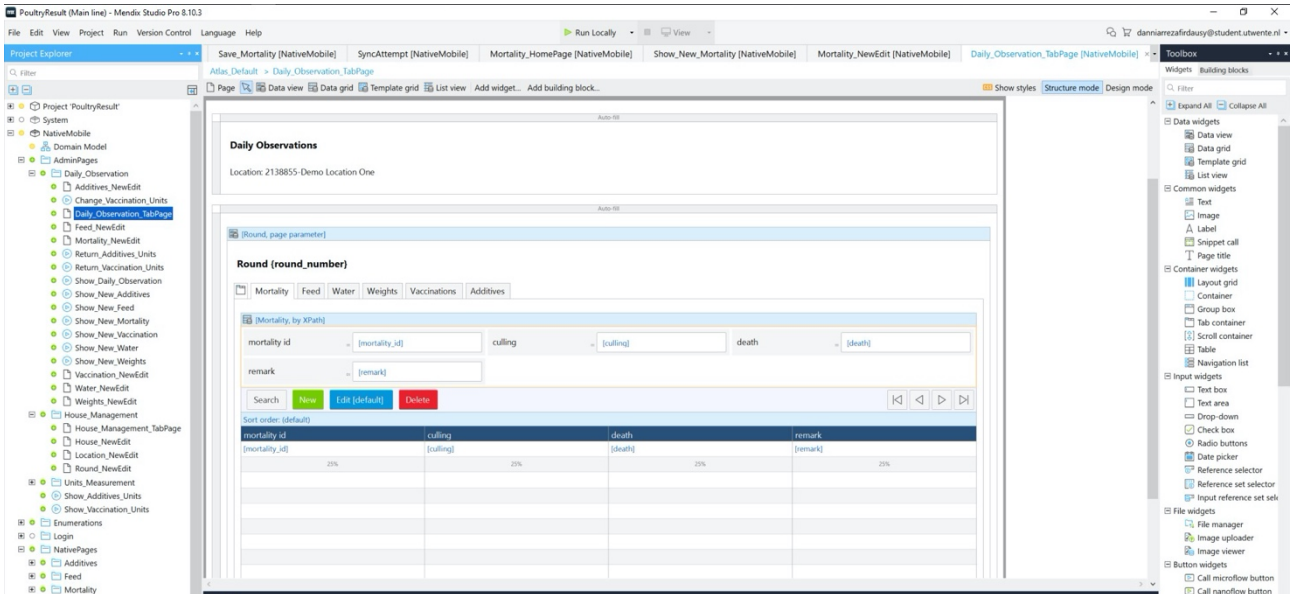

Figure 3 Developing User Interface in Mendix

Most of the UI development processes are being done by selecting the premade view components or known as widgets into the page and developers are required to "code" (called XPath)

only when defining constraints of the data that wanted to be shown in some widgets. The result of the UI implementation, after starting the Mendix server, can be seen in the following Figure 4.

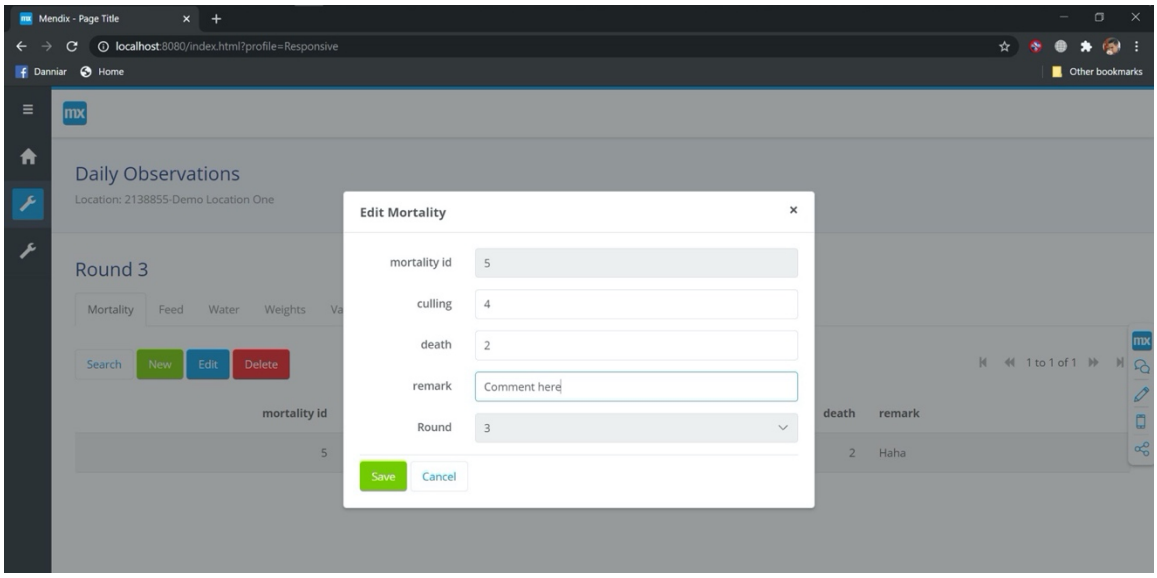

Figure 4 Mendix Web Application User Interface

Business logic, application flow, or functions in conventional raw-code environment, are also realized in a low-code manner by the flow of what they called “Microflow” or “Nanoflow” which adheres to the design principles of Business Process Modelling Notation (BPMN) 2.0. The difference between the two is that the former takes the execution in server side and the latter takes the execution in client side. An example of Nanoflow implementation, as shown in Figure 5, is to save changes for mortality observation data. Figure 5 also reveals that a flow can contain and call another flow, in this case, to call another nanoflow SyncAttempt. It can also be seen that in SyncAttemp, the nanoflow is arranged to perform actions that Mendix has provided for their mobile application development environment, which are to check the internet connection and synchronize the local database in the device with the back-end server as the realization towards the user story GR 5.

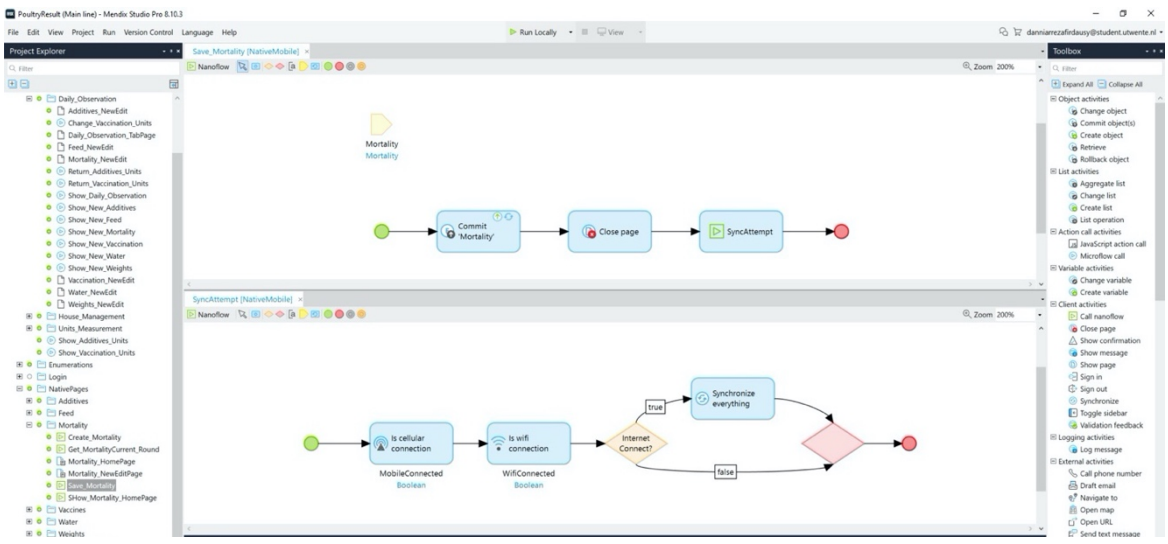

Figure 5 Nanoflow Containing Mobile Application Specific Actions

Having the web application and the data prepared, the mobile application development can be started by creating the required pages for each user story, arranging the provided view widgets into a mobile device sized page and apply nanoflow functions to some of the widgets such as the Save_Mortality nanoflow above. The construction of the mobile app pages using widgets are shown in the following Figure 6, where it consists of structures of the widgets in which it will be compiled into the actual widget in runtime. Furthermore, Figure 6 also shows the realization of the user stories DO 1.1, DO 2.1, DO 3.1, DO 4.1, DO 5.1 and DO 6.1.

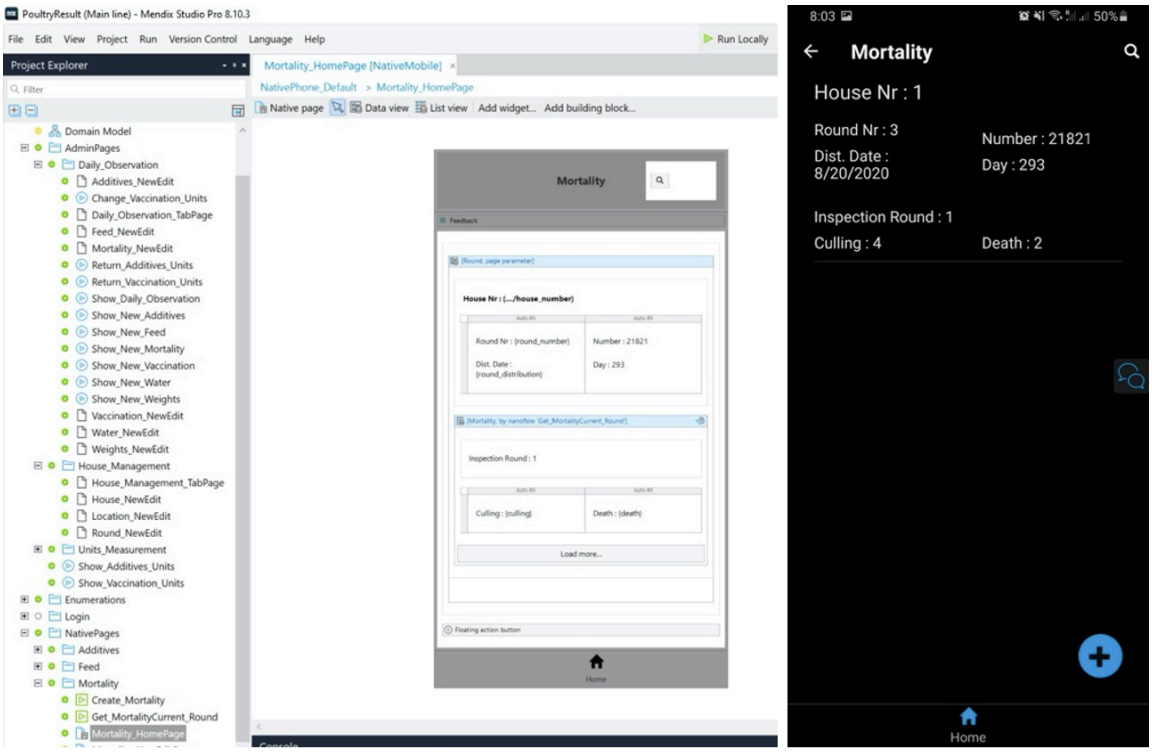

Figure 6 Mendix Mobile Application User Interface Structure and Runtime

Based on the defined user stories, the ability to upload pictures from the device's camera is also desired. In this experiment, this particular user story will be realized as a function provided to the user when they create a new Mortality data. As shown in Figure 7, the structure of the new edit page along with the built-in function to use the device's camera functionality is arranged where the resulting page is revealed in Figure 8. This implementation is addressing the user stories of DO 1.2, DO 1.3, and the other group of new edit animal daily observations along with the requirement of device camera utilization in DO 8.1.

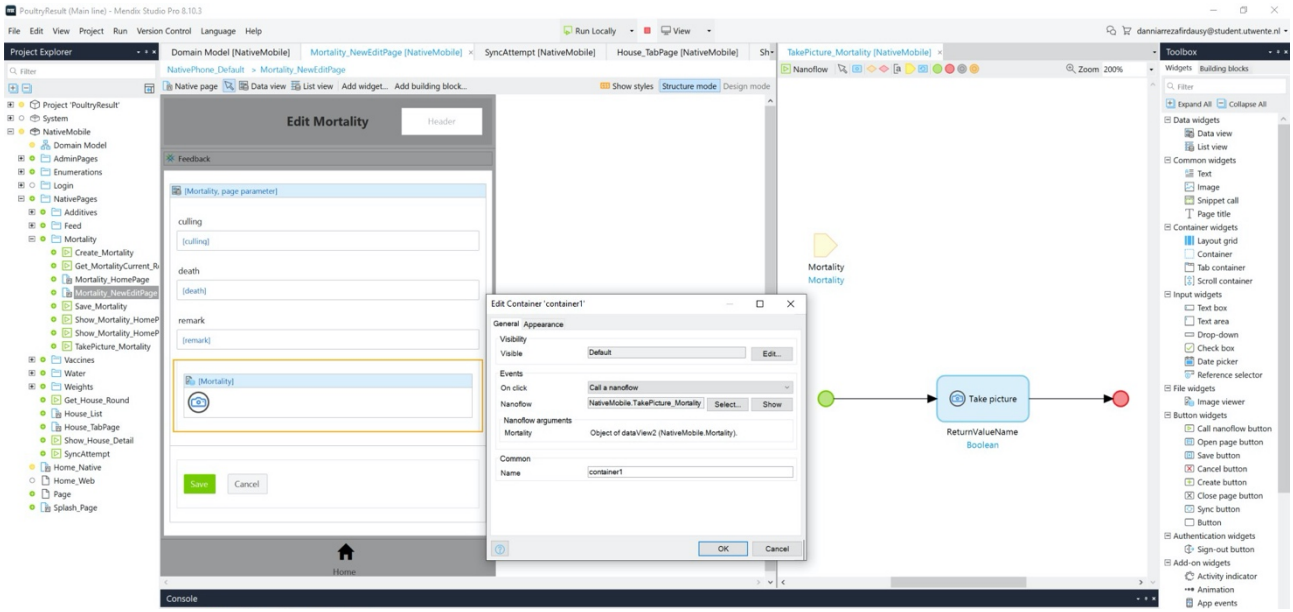

Figure 7 Mendix Building the Mortality New Edit Page

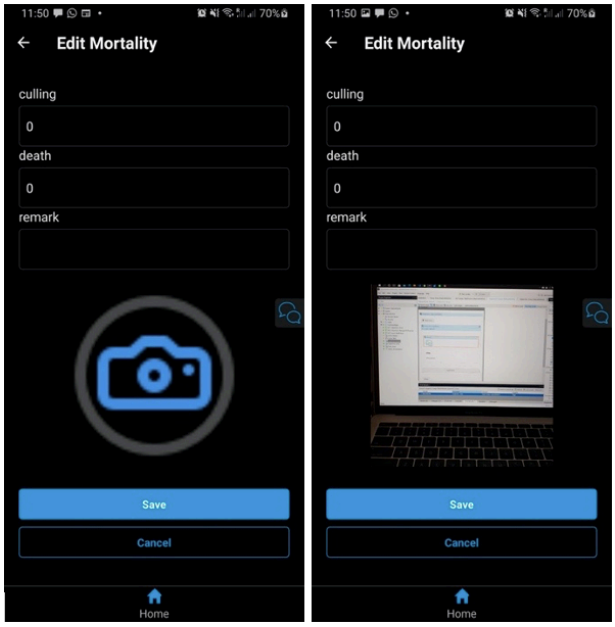

Figure 8 Mendix Resulting Mortality New Edit Page with Camera Function

2.6.3. Evaluation Phase

In order to evaluate the outcome of the experiment, multiple factors are taken into consideration. First, is to evaluate the selected user stories realization. Based on Table 5 and the execution earlier, it is discovered that the user stories related to data deletion from mobile apps in Mendix is not yet realized nor possible. This is due to nanoflow in Mendix has not provided the function to delete an object as visualized in the following Figure 9, unlike the microflow for the web application version which has provided the function that comes as a standard.

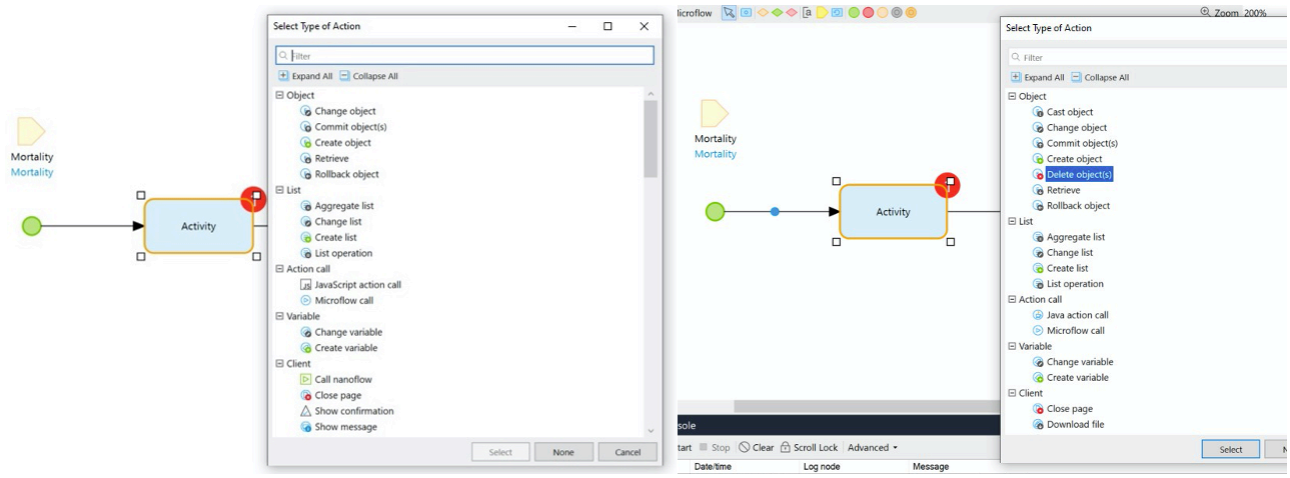

Figure 9 Delete Action in Nanoflow vs Microflow

Furthermore, the user stories also explicitly mentioned that a dashboard containing graphs to visualize the animal information is desired. However, it is discovered that in Mendix this capability as a standard only has been offered in their web application development as shown in Figure 10, whereas in the following Figure 11 shows that it has not been offered as a standard yet in their mobile application development. Based on this result, it can be deduced that at this point of time, the development of mobile application using low-code approach offered by Mendix has not yet provided the flexibility and level of customization that the company desired in relevance to the defined user stories.

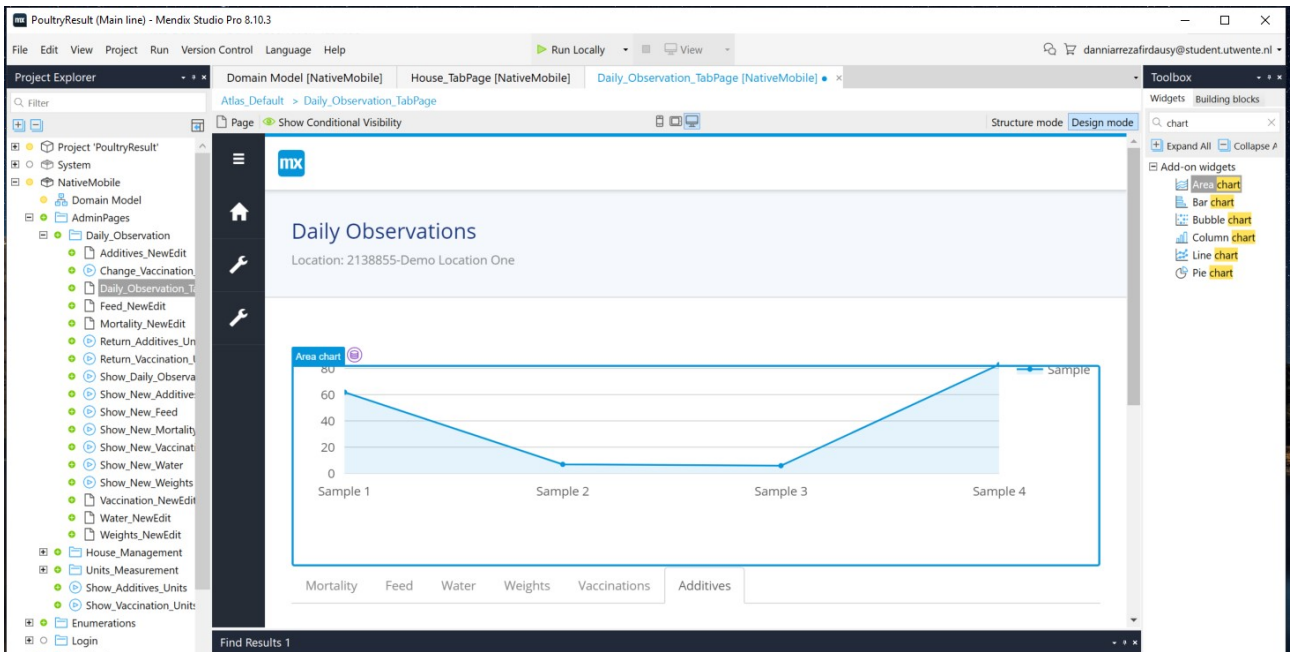

Figure 10 Graphical Charts Comes as a Standard in Mendix Web Application

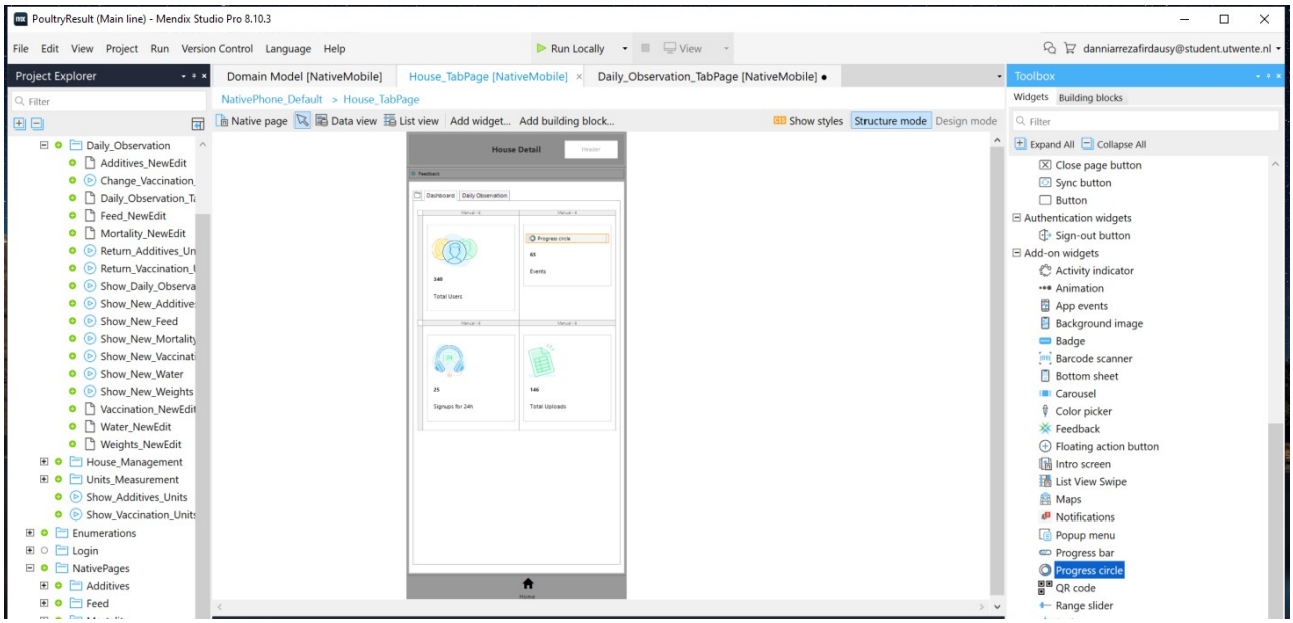

Figure 11 Graphical Charts not yet Offered in Mendix Mobile Application

Secondly, the evaluation towards the adoption of Mendix as FarmResult's development platform takes into consideration of resources that the company needs to allocate. In this case, resources being discussed are consisting of resources of time, effort and capital. It is proven that development using low-code approach by Mendix was fast, as the time needed to develop the mobile application along with the web application earlier only took 3 working days for an experienced Mendix Web Application Developer who have just started working on the Mobile Application version and can be predicted that it will take 6-10 of working days for a complete beginner to deliver the same result by following the developer tutorials provided by Mendix. As time is considered to be an advantage, effort can also be considered as lessened. Since the development processes are using graphical representation of view widgets and BPMN's principles, along with the availability of tutorials provided, developers with business background are being given hands-on ability towards the development. However, this facility come with its price, especially since the platform is being provided by a third-party company so a set of price lists will arise at the end of the development phase. Referring to Mendix's website, their pricing starts from € 1875 / month with 3-year commitment and as the basic plan it is limited up to 50 internal or 500 external users. Considering the big amount of users that FarmResult has projected for their farmer clients in relevance with the budget, along with the disadvantages regarding flexibilities and level of customization, it is decided that this low-code approach platform does not fit the requirements from FarmResult at the moment and thus Google Flutter framework remains as the solution to be adopted.

2.7. Google Flutter Project Execution

In this project execution using Google Flutter framework, the Scrum approach is being followed as the project management guideline. According to Scrum approach, the project will be divided over continuous iterations, in this case will be divided over 5 sprints with the duration for each sprint will be ranging between 1-3 weeks. The details regarding this project timeline is described in the following Table 6. From Table 6, it is shown that Sprint 3 takes the longest time due to the immediate action that was required to clear impediments to the development process, which

will be discussed later in the following sub-section. Along with that, it is also observable that week 6 and 7 is missing from the timeline since the week was being used to perform an experiment using Mendix as detailed in the previous section. In the following subsections, the initial planning, development execution, final user testing and evaluation towards future works will be detailed.

Table 6 Scrum Sprints Project Timeline

Iterations	Weeks	Goal
Sprint 1	Week 2	Requirement Analysis
Sprint 2	Week 3-5	Overall UI Pages Development and Components
Sprint 3	Week 8-10	Back End Development and Integration
Sprint 4	Week 11-12	Value Added Features Implementation
Sprint 5	Week 13-14	Usability Testing & Evaluation

2.7.1. Initial Phase

In order to start working on the initial prototype of the mobile app using Google Flutter, requirements analysis should be performed in the first place. This is done to establish the general scope of the solution to be developed, avoid scope creep and to gain better understanding of the purpose of the solution. Requirement analysis is performed in Sprint 1 and begins by gathering information from the product owner about what the mobile application is expected to be able to do both present time as well as in the future, the infrastructure that supports PoultryResult solution and FarmResult's clients as the potential users of the developed app. Along with this, requirements are also gathered by testing and analyzing the user interface, experience and capabilities of the existing mobile app. From this process, initial user stories such as GR 1, GR 2, GR 3, GR 4, GR 8, DO 1, DO 2, DO 3, DO 4, DO 5 and DO 6 are formalized.

2.7.2. Execution Phase

Having initial user stories set in place, the Flutter mobile app development project can be initialized. The mobile app development in this project is carried out using Android Studio in MacOS, considering Flutter is a cross-platform framework and MacOS is capable of running emulators for both Android and IOS. Project execution started in Sprint 2 and in this sprint, UI development is the main focus without the function to integrate with the back-end server.

The project is initialized by creating the login page that can be used by the user to enter their username and password, although in this sprint, the function to authenticate and logout has not yet been implemented. The work is then continued by creating a page that asks the user to select the farm site location that they want to access. Further, the user is faced with the list of farm houses that exist in the selected farm site location, which if one of the farm houses is selected, then the app will redirect the user to the daily observations page. The visualization of these implementations is shown in the following Figure 12.

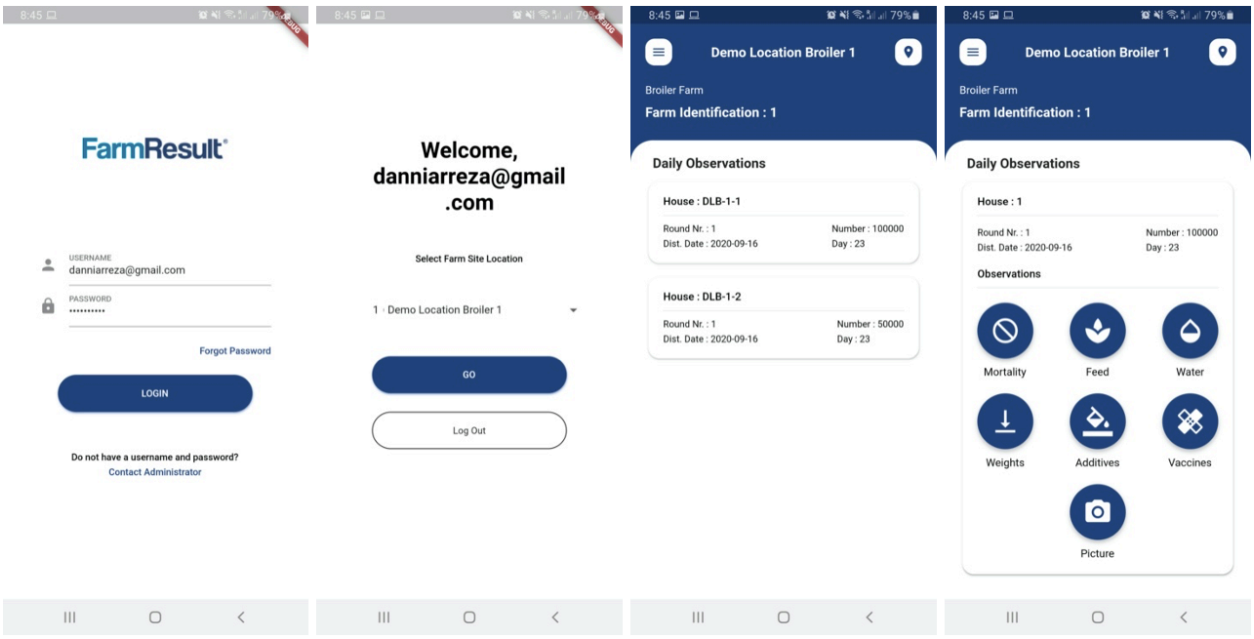

Figure 12 Login, Farm Sites, Farm Houses and Daily Observations Menu Page

The work is further carried on by creating the pages for each grouping of the animal daily observations visible in the previous Figure 12, which consist of Mortality, Feed, Water, Weights, Additives, Vaccines and Picture. Each of these observations will have their own daily entry list page and a page to create a new as well as edit existing data entry as shown in Figure 13. The following Figure 13 is showing the realization towards the user stories of animal daily observations in which the Mortality home and new edit pages are representing the Water and Weights observations whereas the Feed are representing the Additives and Vaccination observations since they hold similar design properties.

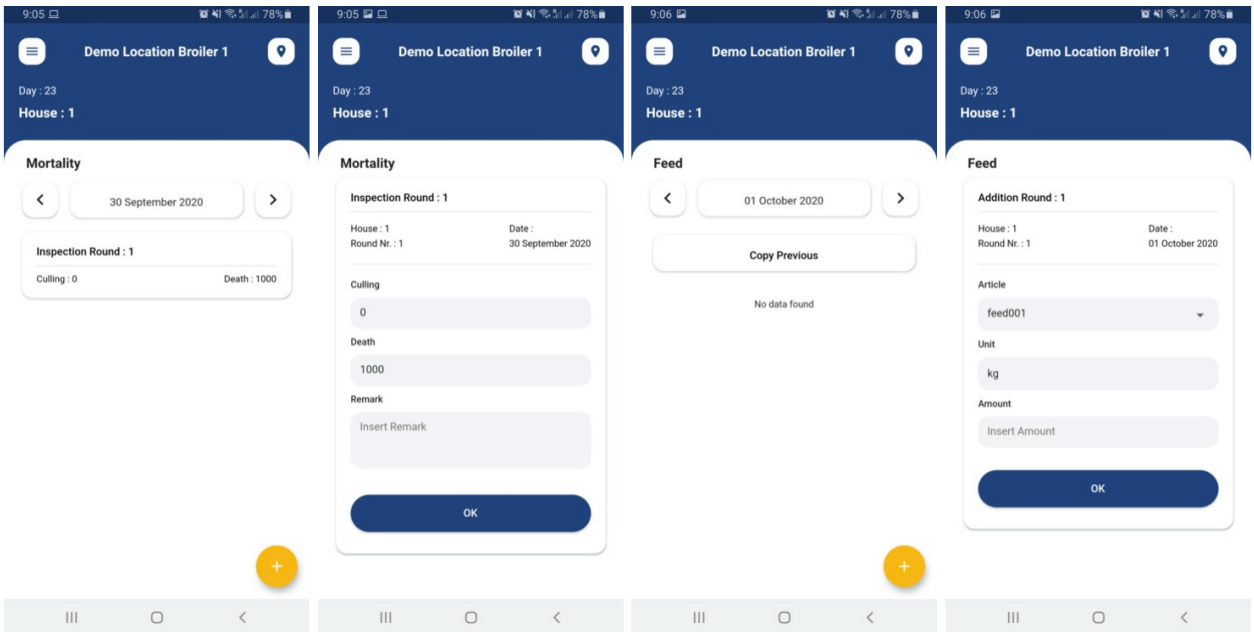

Figure 13 Mortality and Feed Pages

Entering the week 4 of the development, the work is mainly focused towards the implementation of charts for house climate dashboard and daily observations dashboard using charts library available for Flutter framework. The line chart is being used, since these dashboards will present temporal data for the whole round of chicken rearing. Figure 14 shows the dashboards which realize the requirement to provide users the ability to see relevant information about both the animals or the house climates.

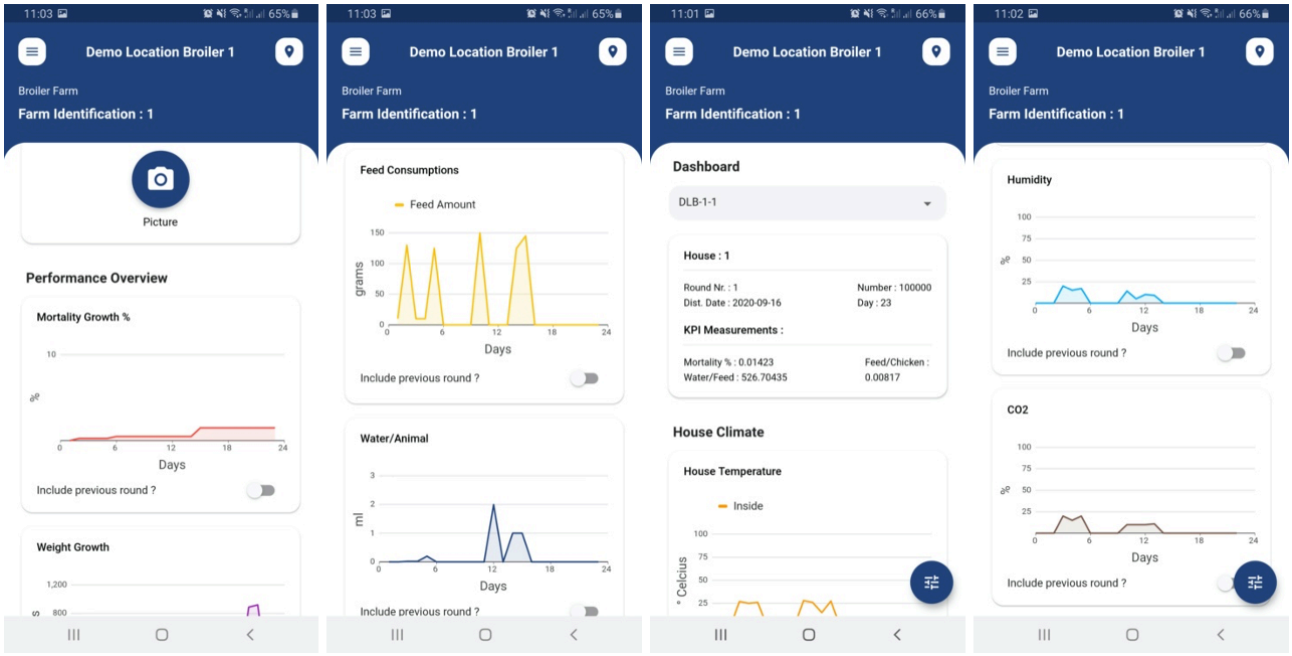

Figure 14 Daily Observations Performance Dashboard (left) and House Climate Dashboard (right)

The work of the Second Sprint is being closed by the implementation of a small addition towards the initial user stories. This addition is concerning the provision of eggs production observation that is only made available if the user is in or selecting a layer farm house. Since the design properties of this requirement are not far different from the rest of the daily observations, then this inclusion can be executed right away in the same sprint. In addition, to finalize the UI implementation in this sprint, a page to utilize the camera functionality is also developed. This camera implementation has only utilized the function to capture images however. Since the finalized business usage of this has not yet been established, the function to capture video is not yet a priority in this execution. Figure 15 below exhibits the implementation of the user stories mentioned earlier and this indicates the end of Sprint 2 and the next Sprint 3 will be discussed in the following paragraph.

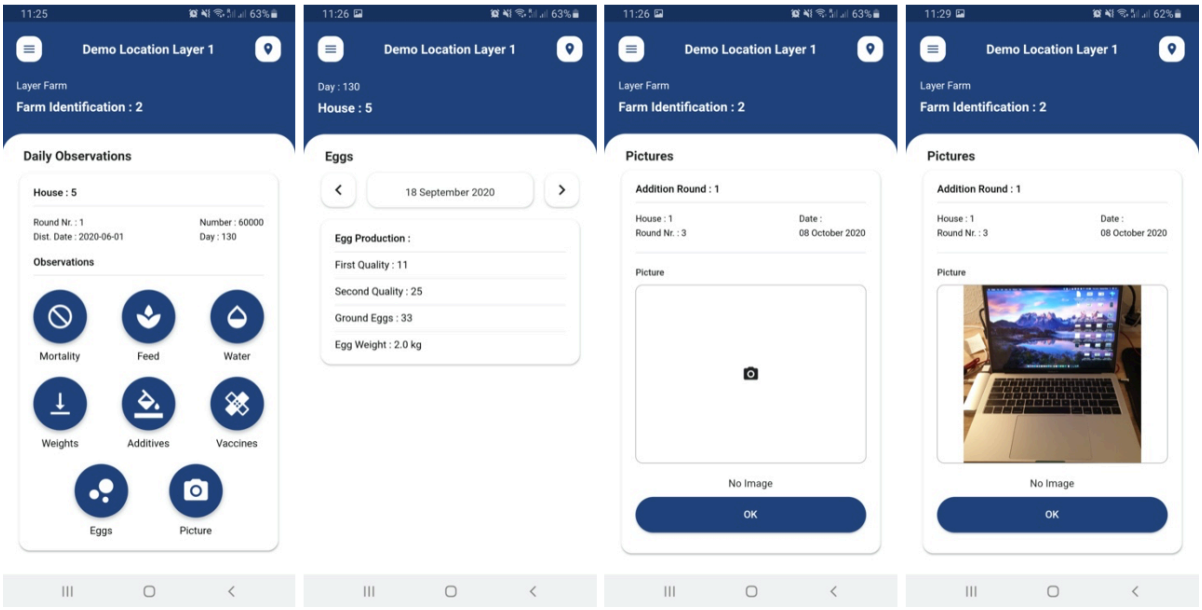

Figure 15 Eggs Observations for Layer Farm and Picture New Edit Page

Sprint 3 is about the integration between the developed front-end with the back-end services. The initial plan was to integrate it with back-end services of the existing PoultryResult. However, since the company was also still working on a major remodeling for the infrastructure along with the back-end services at the time, it is then agreed that for this internship project the back end services will be built from the start using Mendix since it is also able to facilitate a quick development of back-end services using REST API. The database design for the development of back end service using Mendix will align towards the new design that the senior developer in the company has proposed, since there are some additional entities being introduced such as Eggs Production etc. Figure 16 below presents the domain model implemented in Mendix which is different with the one developed earlier in the previous experiment with Mendix Mobile Application as it is on a different project.

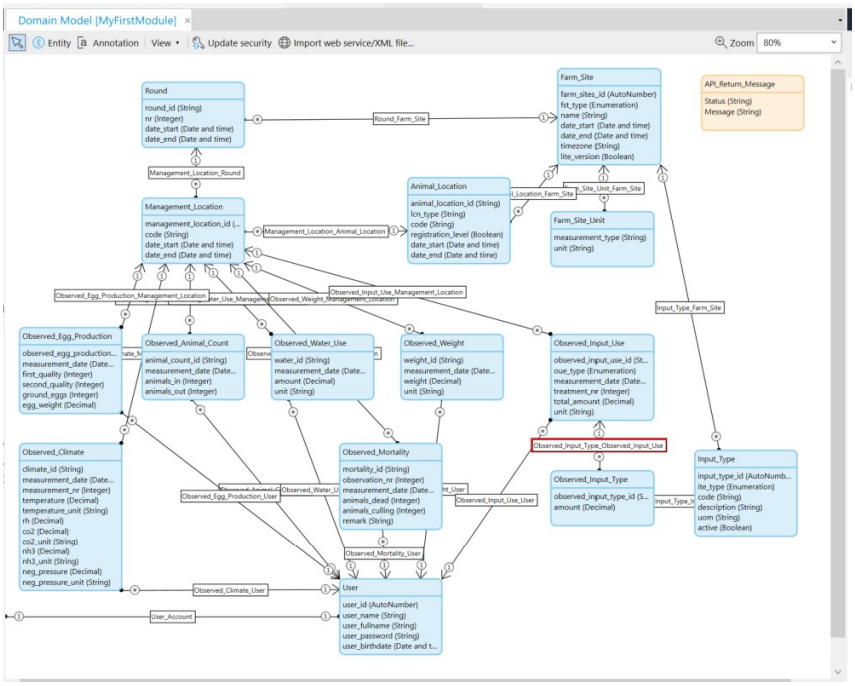

Figure 16 Back-End Service Domain Model

Having the REST API set in place using Mendix, the Flutter mobile app is ready to integrate data from the front with the back-end. Since it is desired that the Flutter PoultryResult will be able to work in an offline condition, local database implementation using SQFlite (SQLite implementation for Flutter) is required to store data locally before sending it to the back-end server. Furthermore, in order to impose a close relationship of data structures between the mobile app and back-end, a similar data structure as Domain Model in Figure 16 with a little bit of adjustment is being used in the local SQFlite. As the mobile app is ready to store data, the first functionality to be integrated is the login authentication and User session. By having user data stored in session after they logged in, storing (animal and house climate) observations data can be indicated that it is performed by the logged in user. Next is to implement integration of the Farm Site, Animal Location, Round and Management Location. Currently, this integration takes place whenever the user accesses the page where they select farm site location and the same method will be used for the rest of the data entities. This means that the previously mentioned user stories such as GR 1, GR 2, GR 3, GR 4, GR 5, GR 8 are fully realized and the pages in Figure 12 are now fully operational.

Since the user is now able to access the daily observations menu with the correct data integration, the next work to perform is to integrate animal daily observations data, including the new eggs production observation. The challenge in this work is that the Feed, Additive and Vaccine observations are being stored into the same data structure which is consisted of tables of Observed Input Use to store the aggregate information of the observation, Observed Input Type to store detailed information of the input use in relation with the input type such as the feed or additive type being used. Approaching the end of the third sprint, some complementary features such as “Copy Previous” for Water and Feed observations are implemented. Along with it, business rules such as limit towards Mortality of only maximum 2 observations a day and only once for Water and Weight observations are also imposed by controlling the appearance of the “+” button as shown in the following Figure 17. In addition, since the ability to delete these observations is also desired and the fact that Flutter framework has already provided a built-in function of swipe to dismiss the card

widgets, this gesture is then implemented in this case. These implementations are then fulfilling the user stories of DO 1, DO 2, DO 3, DO 4, DO 5, DO 6 and DO 7 along with their sub-stories.

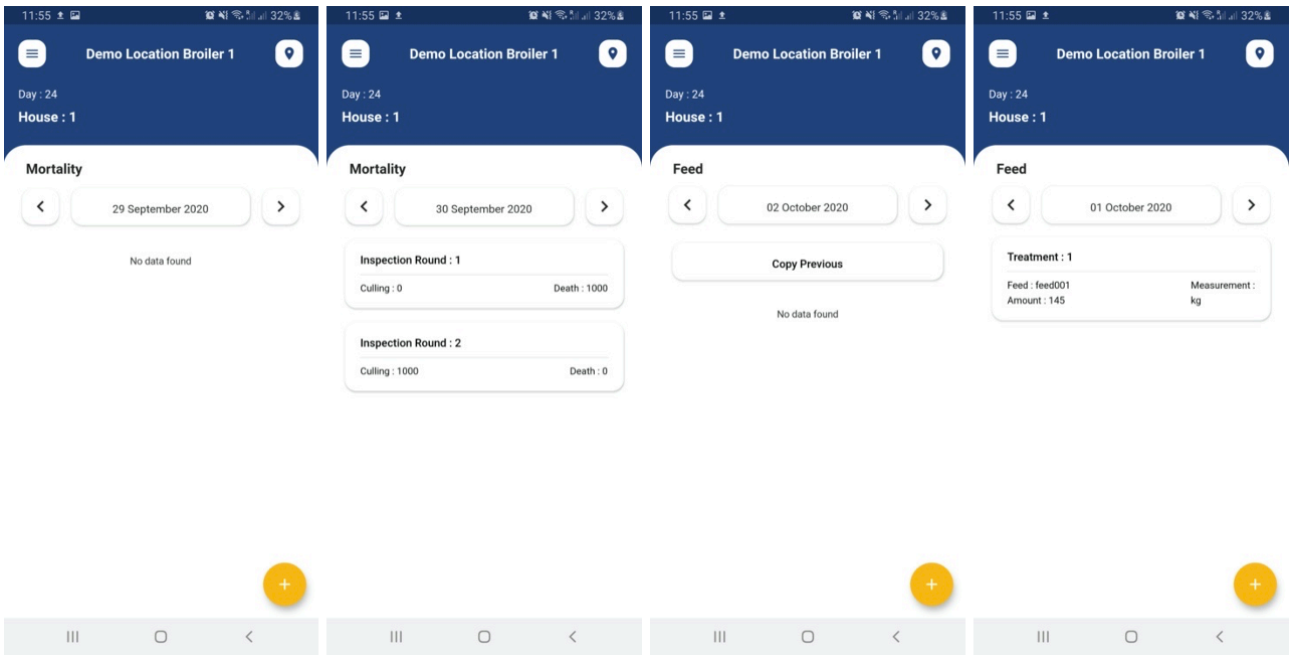

Figure 17 Business Rules of Limit to the Daily Observations

Sprint 4 is allocated to implement additional features that further add value of the developed app in comparison with the previous version. This addition comprises the implementation of house climate manual observation, operationalization of the dashboard charts and the implementation of local push notification. The inclusion of House Climate manual observation is a new user story that formulated by the end of the previous sprint and was derived from the updated business requirement obtained from the latest client feedback. This implementation will share the same design with the animal daily observations such as the same observations list and new edit page. Further, since all observations data entry processes have been provided, dashboard charts are now configured to show temporal data visualization of the animal observation and house climate entries. In addition, some KPIs calculation based on animal daily observation are also put in place above the charts as shown in Figure 14, to provide a more comprehensive information of the current selected house to the user. These implementations can be seen in the following Figure 18 on the left and these realize the user stories CO 1 and CO 2 along with the sub-stories.

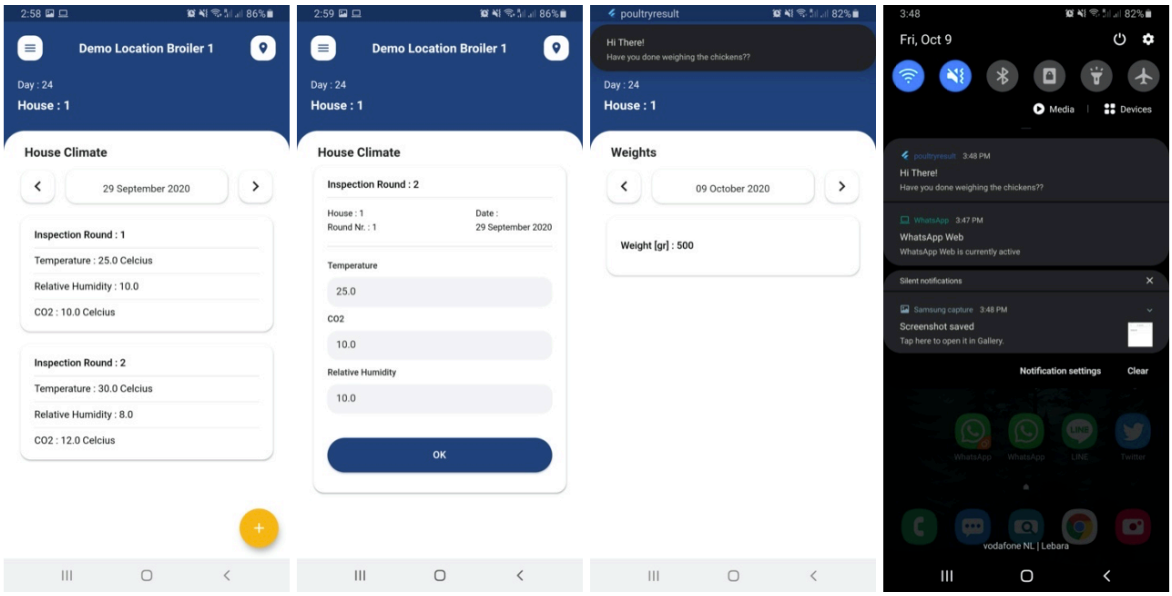

Figure 18 Manual House Climate Observation and Scheduled Local Notification

Moreover, since user story GR 7 has stated that scheduling the entry of missing data is needed, the local notification function is then implemented. This is made possible since Flutter framework has provided the plugin with an easy installation and configuration. This plugin also comes with a relatively complete built-in functionalities such as immediate notification, scheduled (delayed) notification as well as daily, weekly or monthly notification. Taking advantage of this, for current development purposes, Figure 18 on the right shows a scenario of local notification being triggered with a delay of 1 minute after the user creates a new weight observation. It is observable that the notification can still be triggered even though the app is closed and will stay visible in the notification tray until the user dismisses it themselves. This scenario represents the real-world situation where a user would want to get a reminder to perform animal weight observation in the next 7 days if they have not performed it, thus fulfilling the stated user story.

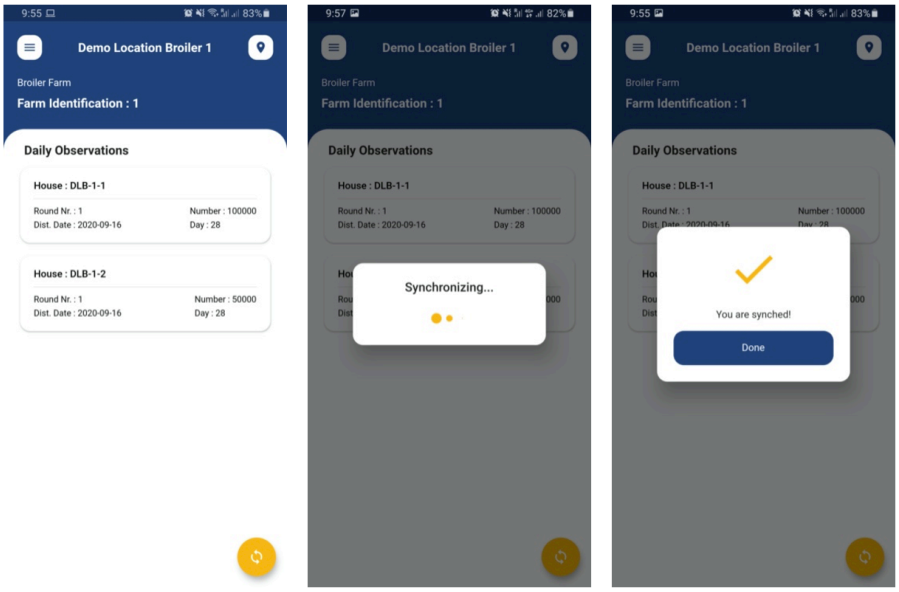

Figure 19 Synchronization Button and Status

Next feature to be delivered is the implementation of data synchronization. The scenario for this case is when the user is performing observations inside the farm house, mostly they will not have a good internet connection and will not be ideal for them to fill in the information later after they got connected again. In this case, the mobile application will detect the device internet connection and if connection is not found, then the prepared data to be sent to the back-end server will be stored first in the local database waiting in a queue for synchronization later following First-In First-Out (FIFO) principle. In order to test this synchronization feature, a 'sync' button is provided in the homepage for daily observation. In the earlier Figure 19 above, the button will present the status of the synchronization process while Figure 20 below shows the FIFO takes place behind the screen with the scenario of creating a new eggs production daily observation then updates the amount of the eggs twice (first quality followed by second quality amount) under offline condition. Beside inserting and updating data in offline-then-sync-later feature, data deletion is also provided incorporated inside the swipe to delete data gesture. Having this implemented, the user story GR 5 to enable data operation while offline is then realized.

```
I/flutter (31601): {status: Success, message: Insert id 14423198196555 is successful.}
I/flutter (31601): {status: Success, message: Update id 14423198196555 is successful.}
I/flutter (31601): {status: Success, message: Update id 14423198196555 is successful.}
```

Figure 20 First-In First-Out Synchronization Queue

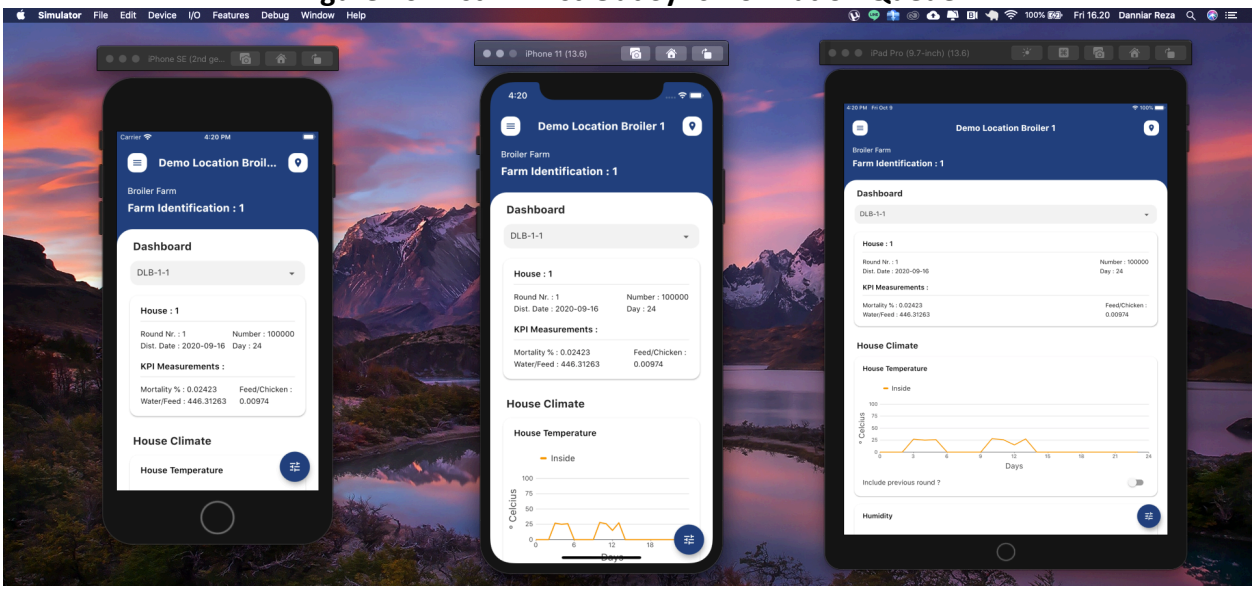

Figure 21 Flutter Cross-Platform Implementation in iOS

By the end of Sprint 4, the assessment towards the cross-platform properties of the Flutter framework is performed. In order to guarantee the quality of the developed apps, one of the many contributing factors is the expected consistency across various platforms (Inukollu, Keshamoni, Kang, & Manikanta, 2014). Figure 21 has shown that running the app in multiple iOS emulators that consist of diverse screen properties from iPhone SE, iPhone 11 and iPad 7 is not showing any issue. Furthermore, similar assessments are also carried out towards Android emulators with different screen properties as represented in Figure 22 below and similar results is also obtained. However, a more elaborate testing with the potential users is required in order to assess if the implemented

design can be used by them effortlessly. This assessment also signals the end of the execution phase, which concludes the second project goal of developing the prototype mobile app using Flutter.

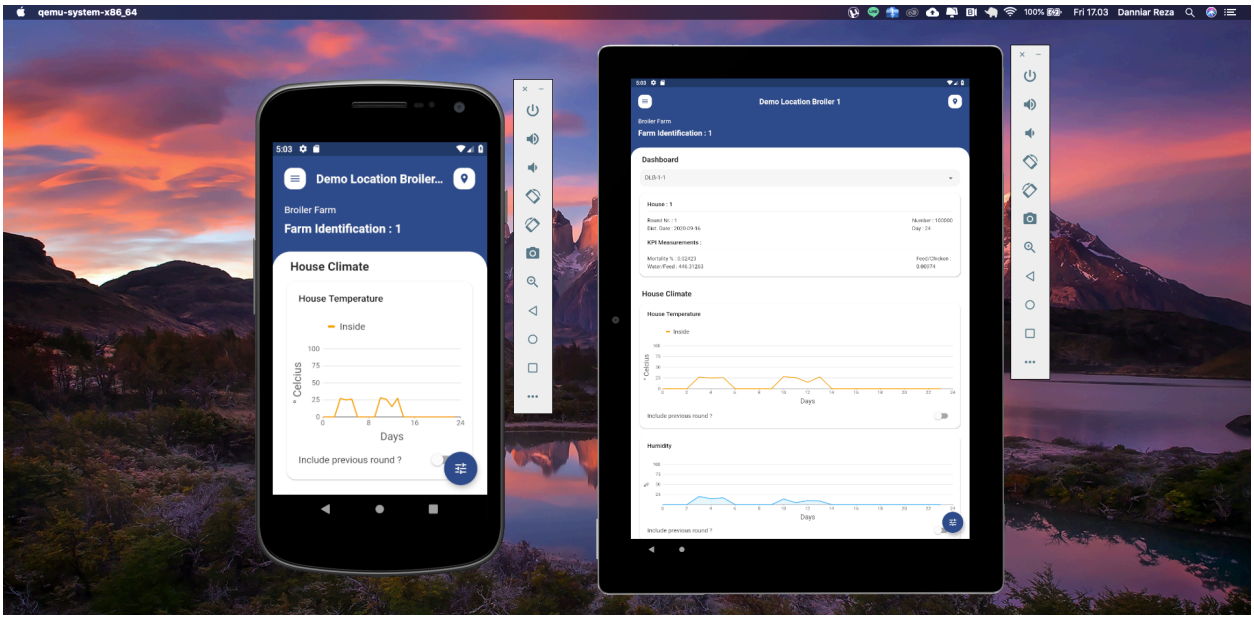

Figure 22 Flutter Cross-Platform Implementation in Android

2.7.3. Testing Phase

The Testing Phase takes place as part of the last sprint prior to the final evaluation. As defined in Section 2.4 Project Approach and Methods, in order to test the developed mobile apps in relevance towards the potential user acceptance, the Usability Testing is preferred. In this test, the participants will be given a set of certain tasks that they need to perform. Later after this testing session, Effectiveness, Efficiency and Satisfaction metrics will be measured. Effectiveness and Efficiency metrics will require the measurements of success result and the time taken from the list of tasks as shown in Table 7 below.

Table 7 List of Usability Testing Tasks

Group	No	User Story	Success	Time
General	GR 2.	See and select their farm site location		
General	GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location		
Mortality	DO 1.1	See current round animal mortality of the same or another day		
	DO 1.2	Enter new entry of the chicken flock's mortality observation		
	DO 1.3	Edit entry of the chicken flock's mortality observation		
	DO 1.4	Delete entry of the chicken flock's mortality observation		
Feed	DO 2.1	See current round animal feed of the same or another day		
	DO 2.2	Enter new entry of the animal feed		
	DO 2.3	Edit entry of the animal feed		
	DO 2.4	Delete entry of the animal feed		
	DO 2.5	Copy previous entry when animal feed observation for current day is not yet present		

Weight	DO 3.1	See current round animal weight of the same or another day		
	DO 3.2	Enter new entry of the animal weight observation		
	DO 3.3	Edit entry of the animal weight observation		
	DO 3.4	Delete entry of the animal weight observation		
Eggs	DO 7.1	See current round egg production of the same or another day		
	DO 7.2	Enter new entry of the egg production observation		
	DO 7.3	Edit entry of the egg production observation		
	DO 7.4	Delete entry of the egg production observation		
General	GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location		
Climate Data Intervention	CO 2.1	See manual house climate observations		
	CO 2.2	Enter new manual house climate observations		
	CO 2.3	Edit entry of the manual house climate observations		
	CO 2.4	Delete entry of the manual house climate observations		

Meanwhile, the Satisfaction will be measured by giving the participants a questionnaire that measures their task level satisfaction which is attached in Table 8. In this questionnaire, the participant can give a checkmark for each task in the response box where they think suitable. Upon the end of the session, the check marks will be counted for each task and will be calculated following the previously defined formula in Section 2.4.3.

Table 8 List of Task Level Satisfaction

No	Task	Response				
		SD	D	N	A	SA
GR 2.	See and select their farm site location					
GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location					
DO 1.1	See current round animal mortality of the same or another day					
DO 1.2	Enter new entry of the chicken flock's mortality observation					
DO 1.3	Edit entry of the chicken flock's mortality observation					
DO 1.4	Delete entry of the chicken flock's mortality observation					
DO 2.1	See current round animal feed of the same or another day					
DO 2.2	Enter new entry of the animal feed					
DO 2.3	Edit entry of the animal feed					
DO 2.4	Delete entry of the animal feed					
DO 2.5	Copy previous entry when animal feed observation for current day is not yet present					
DO 3.1	See current round animal weight of the same or another day					
DO 3.2	Enter new entry of the animal weight observation					
DO 3.3	Edit entry of the animal weight observation					

DO 3.4	Delete entry of the animal weight observation					
DO 7.1	See current round egg production of the same or another day					
DO 7.2	Enter new entry of the egg production observation					
DO 7.3	Edit entry of the egg production observation					
DO 7.4	Delete entry of the egg production observation					
GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location					
CO 2.1	See manual house climate observations					
CO 2.2	Enter new manual house climate observations					
CO 2.3	Edit entry of the manual house climate observations					
CO 2.4	Delete entry of the manual house climate observations					

After the testing session is carried out, an analysis towards the testing result can be started to be performed. From the test, result data in the form of the participants tasks completion and their tasks completion time is obtained. Moreover, the participants' satisfaction in performing these given tasks is also obtained from the questionnaire that was given after they finished all of the tasks. The following Table 9 explains about the task completion and time consumed result of 3 participants, which will be represented in binary value for "S" as the Success indicator and in seconds value for "T" as the Time measurement. These 3 participants represent 3 different user backgrounds consisting of a Technical Implementation Consultant from FarmResult, a business developer from an Indonesian poultry integrator company and the owner of 2 poultry farms in the Netherlands.

Table 9 Task Completion and Time Consumed Results

Group	No	A		B		C	
		S	T	S	T	S	T
General	GR 2.	1	10	1	5	1	10
General	GR 3.	1	10	1	30	1	35
Mortality	DO 1.1	1	7	1	3	1	10
	DO 1.2	1	20	1	17	1	20
	DO 1.3	1	13	1	19	1	15
	DO 1.4	1	14	1	15	0	10
Feed	DO 2.1	1	5	1	12	1	10
	DO 2.2	1	21	1	20	1	25
	DO 2.3	1	13	1	15	1	20
	DO 2.4	1	3	1	3	0	10
	DO 2.5	1	5	1	3	1	5
Weight	DO 3.1	1	7	1	6	1	10
	DO 3.2	1	9	1	5	1	5
	DO 3.3	1	8	1	22	1	6
	DO 3.4	1	2	1	2	0	8
Eggs	DO 7.1	1	14	1	40	0	0

	DO 7.2	1	25	1	22	0	0
	DO 7.3	1	8	1	15	0	0
	DO 7.4	1	2	1	2	0	0
General	GR 4.	1	5	1	5	1	8
Climate Data Intervention	CO 2.1	1	27	1	4	1	10
	CO 2.2	1	34	1	25	0	0
	CO 2.3	1	6	1	10	0	0
	CO 2.4	1	4	1	2	0	0

Meanwhile, data regarding the satisfaction from the participants who have completed all of the task is explained in the following Table 10, which consist of the questions that was given in respect to each of the defined task for this testing purpose and then followed by the count of the response scale ranging from “SD” as Strongly Disagree to “SA” as Strongly Agree.

Table 10 Task Satisfaction Questionnaire

No	Task	Response				
		SD	D	N	A	SA
GR 2.	See and select their farm site location	0	0	1	1	1
GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	0	0	1	1	1
DO 1.1	See current round animal mortality of the same or another day	0	0	0	2	1
DO 1.2	Enter new entry of the chicken flock’s mortality observation	0	0	0	2	1
DO 1.3	Edit entry of the chicken flock’s mortality observation	0	0	0	1	2
DO 1.4	Delete entry of the chicken flock’s mortality observation	0	0	1	0	2
DO 2.1	See current round animal feed of the same or another day	0	0	0	2	1
DO 2.2	Enter new entry of the animal feed	0	0	0	2	1
DO 2.3	Edit entry of the animal feed	0	0	0	1	2
DO 2.4	Delete entry of the animal feed	0	0	1	0	2
DO 2.5	Copy previous entry when animal feed observation for current day is not yet present	0	0	0	1	2
DO 3.1	See current round animal weight of the same or another day	0	0	0	2	1
DO 3.2	Enter new entry of the animal weight observation	0	0	0	2	1
DO 3.3	Edit entry of the animal weight observation	0	0	0	1	2
DO 3.4	Delete entry of the animal weight observation	0	0	1	0	2
DO 7.1	See current round egg production of the same or another day	0	0	0	3	0
DO 7.2	Enter new entry of the egg production observation	0	0	0	2	1
DO 7.3	Edit entry of the egg production observation	0	0	0	1	2
DO 7.4	Delete entry of the egg production observation	0	0	0	1	2

GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location	0	0	1	1	1
CO 2.1	See manual house climate observations	0	0	1	1	1
CO 2.2	Enter new manual house climate observations	0	0	1	1	1
CO 2.3	Edit entry of the manual house climate observations	0	0	1	0	2
CO 2.4	Delete entry of the manual house climate observations	0	0	1	0	2

Since the data has been gathered, the first analysis can be performed to obtain the effectiveness measurement. Based on the task completion result from Table 9, it is obtained that the effectiveness rate reaches 86%. This result can be considered as above the average compared to the average effectiveness rate of 78% from an analysis performed on almost 1200 usability test tasks (Sauro, 2011). This task completion results also influenced by the fact that the last participant did not perform all of the requested task, due to his tendency to open a discussion about a set of features immediately after performing a set of particular tasks before finishing all of the task. This calculation is further detailed in the following Figure 23.

$$Effectiveness = \frac{Amount\ of\ Task\ Completed}{Total\ Amount\ of\ Task\ Should\ be\ Done} = \frac{24+24+14}{24+24+24} = 86\%$$

SUM =(SUM(C3:C26)+SUM(F3:F26)+SUM(I3:I26))/(COUNT(C3:C26)*COUNT(A31:A33))

	A	B	C	D	E	F	G	H	I	J	K
1	Group	No	A			B			C		
2			S	T	S*T	S	T	S*T	S	T	S*T
3	General	GR 2.	1	10	10	1	5	5	1	10	10
4	General	GR 3.	1	10	10	1	30	30	1	35	35
5	Mortality	DO 1.1	1	7	7	1	3	3	1	10	10
6		DO 1.2	1	20	20	1	17	17	1	20	20
7		DO 1.3	1	13	13	1	19	19	1	15	15
8		DO 1.4	1	14	14	1	15	15	0	10	0
9	Feed	DO 2.1	1	5	5	1	12	12	1	10	10
10		DO 2.2	1	21	21	1	20	20	1	25	25
11		DO 2.3	1	13	13	1	15	15	1	20	20
12		DO 2.4	1	3	3	1	3	3	0	10	0
13		DO 2.5	1	5	5	1	3	3	1	5	5
14	Weight	DO 3.1	1	7	7	1	6	6	1	10	10
15		DO 3.2	1	9	9	1	5	5	1	5	5
16		DO 3.3	1	8	8	1	22	22	1	6	6
17		DO 3.4	1	2	2	1	2	2	0	8	0
18	Eggs	DO 7.1	1	14	14	1	40	40	0	45	0
19		DO 7.2	1	25	25	1	22	22	0	10	0
20		DO 7.3	1	8	8	1	15	15	0	5	0
21		DO 7.4	1	2	2	1	2	2	0	6	0
22	General	GR 4.	1	5	5	1	5	5	1	8	8
23	Climate Data Intervention	CO 2.1	1	27	27	1	4	4	1	10	10
24		CO 2.2	1	34	34	1	25	25	0	5	0
25		CO 2.3	1	6	6	1	10	10	0	4	0
26		CO 2.4	1	4	4	1	2	2	0	6	0
27	Sums		24	272	272	24	302	302	14	298	189
28											

29/30	No	Respondent	Task Completed	Realized Task Completion Time	Expected Task Completion Time
31	1	A	24	272	272
32	2	B	24	302	302
33	3	C	14	189	298
34	Effectiveness		A33))	Overall Relative Efficiency	88%

Figure 23 Effectiveness Metric Calculation

Meanwhile, the second analysis to be done is the efficiency measurement, which is based on the participants tasks completion time. From Table 9, the efficiency rate can be obtained by first multiplying each participant tasks completion time (in seconds) with the task completion value (success value between 0 or 1) to gain their consumed time of successful attempt, sum all of each participants results and then finished by dividing it with the sum of the tasks completion time to obtain the overall relative efficiency (ORE). The calculation in Figure 24 below shows that the efficiency rate from the test is 88%, which is still considered as a high efficiency result considering the issue that some of the tasks were not being performed well by the last participant. Based on this issue, a remark was noted from the last participant stating that future design should remove/reduce extra steps (such as button click to navigate between pages) to go to a certain task and the navigation between pages should be made as direct as possible. However, the validity of this remark should be further investigated in the future to assess if later re-design will eventually improve current result.

$$Overall\ Relative\ Efficiency = \frac{272 + 302 + 189}{272 + 302 + 298} = 88\%$$

SUM = (SUM(E3:E26) + SUM(H3:H26) + SUM(K3:K26)) / (SUM(D3:D26) + SUM(G3:G26) + SUM(J3:J26))

	A	B	C	D	E	F	G	H	I	J	K
1	Group	No	A			B			C		
2			S	T	S*T	S	T	S*T	S	T	S*T
3	General	GR 2.	1	10	10	1	5	5	1	10	10
4	General	GR 3.	1	10	10	1	30	30	1	35	35
5	Mortality	DO 1.1	1	7	7	1	3	3	1	10	10
6		DO 1.2	1	20	20	1	17	17	1	20	20
7		DO 1.3	1	13	13	1	19	19	1	15	15
8		DO 1.4	1	14	14	1	15	15	0	10	0
9	Feed	DO 2.1	1	5	5	1	12	12	1	10	10
10		DO 2.2	1	21	21	1	20	20	1	25	25
11		DO 2.3	1	13	13	1	15	15	1	20	20
12		DO 2.4	1	3	3	1	3	3	0	10	0
13		DO 2.5	1	5	5	1	3	3	1	5	5
14	Weight	DO 3.1	1	7	7	1	6	6	1	10	10
15		DO 3.2	1	9	9	1	5	5	1	5	5
16		DO 3.3	1	8	8	1	22	22	1	6	6
17		DO 3.4	1	2	2	1	2	2	0	8	0
18	Eggs	DO 7.1	1	14	14	1	40	40	0	45	0
19		DO 7.2	1	25	25	1	22	22	0	10	0
20		DO 7.3	1	8	8	1	15	15	0	5	0
21		DO 7.4	1	2	2	1	2	2	0	6	0
22	General	GR 4.	1	5	5	1	5	5	1	8	8
23	Climate Data Intervention	CO 2.1	1	27	27	1	4	4	1	10	10
24		CO 2.2	1	34	34	1	25	25	0	5	0
25		CO 2.3	1	6	6	1	10	10	0	4	0
26		CO 2.4	1	4	4	1	2	2	0	6	0
27	Sums		24	272	272	24	302	302	14	298	189
28											

29/30 No	Respondent	Task Completed	Realized Task Completion Time	Expected Task Completion Time
31	1	A	24	272	272
32	2	B	24	302	302
33	3	C	14	189	298
34	Effectiveness		86%	Overall Relative Efficiency	SUM(D3:D26) + SUM(G3:G26) + SUM(J3:J26))

Figure 24 Efficiency Metric Calculation

Lastly, the satisfaction measurement is obtained by calculating the task satisfaction questionnaire result using index score percentage calculation. From the questionnaire on Table 10, it is obtained the total score from the participants' answered scale points as shown in the calculation

below. After the total score is obtained, then index score percentage can be calculated by dividing it with the maximum score. The detailed calculation for the satisfaction metric is shown in Figure below, which shows the satisfaction result of 87%. This result of the following index score percentage calculation is then compared with the percentage interval as explained earlier in Table 1 and it can be concluded that the participants Strongly Agree that this new version is easy to use.

$$\text{Index Score } \% = \frac{(0*1)+(0*2)+(10*3)+(28*4)+(34*5)}{(24\ tasks*3\ participants*5)} = \frac{312}{360} = 87\%$$

SUM = C29 / C30

	A	B	C	D	E	F	G
1	No	Task	Response				
2			SD	D	N	A	SA
3	GR 2.	See and select their farm site location	0	0	1	1	1
4	GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	0	0	1	1	1
5	DO 1.1	See current round animal mortality of the same or another day	0	0	0	2	1
6	DO 1.2	Enter new entry of the chicken flock's mortality observation	0	0	0	2	1
7	DO 1.3	Edit entry of the chicken flock's mortality observation	0	0	0	1	2
8	DO 1.4	Delete entry of the chicken flock's mortality observation	0	0	1	0	2
9	DO 2.1	See current round animal feed of the same or another day	0	0	0	2	1
10	DO 2.2	Enter new entry of the animal feed	0	0	0	2	1
11	DO 2.3	Edit entry of the animal feed	0	0	0	1	2
12	DO 2.4	Delete entry of the animal feed	0	0	1	0	2
13	DO 2.5	Copy previous entry when animal feed observation for current day is not yet present	0	0	0	1	2
14	DO 3.1	See current round animal weight of the same or another day	0	0	0	2	1
15	DO 3.2	Enter new entry of the animal weight observation	0	0	0	2	1
16	DO 3.3	Edit entry of the animal weight observation	0	0	0	1	2
17	DO 3.4	Delete entry of the animal weight observation	0	0	1	0	2
18	DO 7.1	See current round egg production of the same or another day	0	0	0	3	0
19	DO 7.2	Enter new entry of the egg production observation	0	0	0	2	1
20	DO 7.3	Edit entry of the egg production observation	0	0	0	1	2
21	DO 7.4	Delete entry of the egg production observation	0	0	0	1	2
22	GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location	0	0	1	1	1
23	CO 2.1	See manual house climate observations	0	0	1	1	1
24	CO 2.2	Enter new manual house climate observations	0	0	1	1	1
25	CO 2.3	Edit entry of the manual house climate observations	0	0	1	0	2
26	CO 2.4	Delete entry of the manual house climate observations	0	0	1	0	2
27	Sum Each Participant		0	0	10	28	34
28	Multiply		0	0	30	112	170
29	Sum All Participants		312				
30	Maximum		360				
31	Satisfaction		= C29 / C30				

Figure 25 Satisfaction Metric Calculation

This result proves that the developed prototype has satisfied the business requirements based on the user perspective, and thus, has addressed the third project goal which is to test and evaluate the new prototype of PoultryResult. However, not all of the newly introduced features such as camera and offline synchronization functionalities have been included and measured in this usability test session, since these functions still need further development and alpha testing performed internally.

2.7.4. Evaluation Phase

Since the mobile application has been developed according to the limited time frame and has been tested in the previous section, this evaluation phase discusses the output of this development project in respect towards addressing the fourth project goal. In order to evaluate the advantages and the disadvantages of developing the PoultryResult mobile app using Google Flutter in this

project, an examination in terms of the realization of the user stories is being taken into consideration. Table 11 below shows user stories which are either not yet fully fulfilled or deemed valuable to be discussed and further developed. The discussion may also bring the feedback obtained from the previous testing.

Table 11 User Stories in Discussion

Group	No.	User story	Discussion & Future Improvements
General	GR 2.	User is able to see and select their farm site location	- Re-consider the placement of the button to switch farm site location, a good idea would be to also include it in the side menu along with the "Dashboard" and "Daily Observations". Otherwise, the user will spend a lot of time trying to figure out how to switch farm site location.
	GR 5.	User still able to view, edit and delete Animal Daily Observations and House Climate Observations data while not connected to the Internet	- Flutter supports the implementation of local database SQFlite, which is successfully implemented to store the impending data changes. However, multiple complex testing cases should be considered and addressed such as when different users are submitting changes into the same data with different connectivity status. - Flutter supports function to check internet connection, which is currently implemented to determine whether the mobile app should submit data directly to the back-end or store it into the queue first. However, further real-world situation where internet is connected with a low-to-none internet speed should also be considered. The solution would be to make a ping function or to install another package made available for Flutter to check the internet speed.
	GR 6.	User can include or exclude every measurement through the app	- This user story is not yet implemented due to the delay in the development of back end API. This can be addressed by developing an API in the back end to check and grant user roles to every user which can be used by the mobile app to determine the inclusion or exclusion.
	GR 7.	User wants to schedule the entry of missing data	- This user story has been fulfilled by the implementation of local notification provided by one of the Flutter's library. Further works would be to establish a more formalized business rules or standard operational procedure on the scheduling.
Weight	DO 3.2	User is able to enter new entry of the animal weight observation	- Since decimal value is being used in this measurement and have a potential of fault related with decimal points, a further development and test cases should be considered.
	DO 3.3	User is able to edit entry of the animal weight observation	- Since decimal value is being used in this measurement and have a potential of fault related with decimal points, a further development and test cases should be considered.
Water	DO 4.2	User is able to enter new entry of the animal water consumption observation	- Since decimal value is being used in this measurement and have a potential of fault related with decimal points, a further development and test cases should be considered.
	DO 4.3	User is able to edit entry of the animal water consumption observation	- Since decimal value is being used in this measurement and have a potential of fault related with decimal points, a further development and test cases should be considered.
Picture &	DO 8.1	User is able to	- Camera function has been provided and demonstrated. However,

Video		upload picture of the chicken	the function is still limited to take picture or upload picture from the gallery and has not provided the means to upload to server yet. Improvement should be made in the future such as to convert the image into base64 string format then send it to back-end API. - Provide remark field as the user upload picture to the back-end API.
	DO 8.2	User is able to upload video footage of the situation	- Camera function has been provided only to take picture and has not further extended into implementing video recording function. Even though the package has been made available for Flutter, it needs further development and testing in the future.
Daily Observations Dashboard	DO 9.	User is able to see all relevant information about the animals in one or two combined graphs	- Nice to be able to see daily observations data in a tabular fashion, weekly/multiple days at once especially for Additives and Vaccinations. - Focus of the graph should be set on a certain period of time, feed/water on the last 5-7 days, mortality and weight on the last 14 days. - On top of the menu page for daily observation, there should be a summary/KPIs of the values such as: Mortality total %, Daily average growth, Total feed per broiler.

Based on the discussion in Table 11, it can be summarized that most of the defined user stories can be realized by Google Flutter with some key notes. First note is about the User Interface and User Experience. This is in regards towards the placement of some UI components, the variety of data displayed to the user along with the way they are being presented. Second note is regarding the further development of the offline synchronization in respect with the business rule. Third note is to pay further attention in the white-box aspects of the delivered mobile app. Fourth note is to conduct further development and testing of the picture and video capture. Lastly, as this PoultryResult mobile app delivery was intended to only fulfills prototyping purpose initially, the code development was not fully structured to follows the industry's current best practice such as not adhering to a certain design pattern which would be beneficial for a long-term maintainability and reusability.

2.8. Discussion and Conclusion

This project has proved that the PoultryResult mobile app developed using Google Flutter introduces multiple advantages. First is the advantage of developing mobile apps using a cross-platform framework. This enables the deployment of the new version that requires access into the device's hardware or platform specific into both Android and iOS platform possible. This aspect is important due to the stated requirements such as the camera and local notification functionalities. Another benefit from this aspect is that the provided capabilities to store data into the device's local database and internet connection, enabling the business valuable offline synchronization feature. The second advantage is regarding the user interface and user experience of the delivered mobile app. Due to the huge selection of built-in view components and extension packages available for Flutter framework, building the user interface and specialized functionalities has been substantially accelerated. This user interface advantage also introduces a new set of value-added functionalities to the newly developed version of PoultryResult, such as the implementation of measurements dashboard and KPIs.

Furthermore, the usability testing has provided results stating that this new PoultryResult

prototype has reached the effectiveness rate of 86%, higher than the common average effectiveness rate of 78%. Overall relative efficiency has given the result of 88%, which is still considered as a high efficiency rate despite that not all tasks were successfully being performed. Meanwhile, the analysis towards the questionnaire given to the participants shows that their satisfaction in using this new version of the app is reaching 87%, which can be deduced that they are Strongly Agree about the ease of use of this new version of PoultryResult. This usability testing session can be improved by involving more participants to test this new app version. However, due to the limited time frame of this project in respect with the internship duration, this testing can only involve 3 participants so far.

Finally, in comparison with the current industry's leading low-code development approach such as Mendix, this development of PoultryResult mobile app has proved that using Google Flutter enables FarmResult to implement new functionalities and serve more features to their clients in a more flexible way. Moreover, since this framework is an open-source and free to use mobile software development kit, the constraint of budget and user limit such in the case of Mendix is no longer present. The drawback is that this approach is still relying on raw-code development where programming and technical environment between the front- and back-end will mostly be different. Thus, the success of future developments will partially depend on the availability and capability of the human resource.

However, due to the comprehensive and easy to understand documentation along with the low learning curve of the framework contributed by the fact that it is only using a single programming language, the previous concern will be substantially diminished. All things considered, Google Flutter is a suitable development approach for the company's PoultryResult mobile app solution and further development of this project using Google Flutter is worth continuing in the future.

3. Reflections

3.1. Introduction

In this chapter, reflections on my learning experiences during my internship at FarmResult will be elaborated. The main discussion in this part is focusing towards what I have learned from the working experience in the company as an intern. Moreover, gaps between the practice in academic settings and real-world environments, in respect towards its contribution to my study of software engineering, is also worth to be discussed in this report. Finally, the future implications for my personal development will also be expressed at the end of this chapter.

3.2. Learning Outcomes and Contributions Towards Study

I believe that what I have learned so far from the academic setting has always been limited by the boundary of ideal assumptions and previously unexpected outcomes may reveal when they are being applied to the practice in a real-world environment. To put it into more sense, this perspective is brought into the discussion towards the adoption of Agile development methodology and the practice of usability testing especially in the context of this internship assignment. My previous knowledge and experience in requirement engineering as well as software development are also put into test in this real-world professional environment. In relevance with software development in the academic setting, the low-code development approach that I experienced from a class in the university can finally be used in a real-world setting.

As mentioned in the Project Approach, the Agile development methodology being followed in this project is the Scrum approach which is influenced by the nature of this internship assignment. Even though this project has defined all of the user stories in the product backlog, deciding over their priorities level and the actual implementation (or to-be implemented) in practice is still not as easy as it seems. As a reflection of my past software development experience, in my previous bachelor thesis about software engineering, the Waterfall methodology was being used where prioritization is being done in the manner of phases and every requirement is being considered vital to the success of the overall project. However, in Agile methodology prioritization is being done in the manner of iterations where requirements are sorted based on their business urgency. Having used the conservative approach of Waterfall, switching to Agile means that prioritization decisions based on estimating the balance between business urgency and the team's implementation capability is considered as the vital point. Since in this internship assignment I am the one who is responsible with the planning as well as the development itself, the prioritizing decision became heavily influenced by my own decision with the consideration of my own capability and the readiness of the company in facilitating the infrastructure and back-end APIs. This makes the whole development process progress quicker despite the challenge in the prioritization previously mentioned. Encountering this practical lesson hands-on by myself, I consider this as an invaluable experience where expected benefit from theoretical knowledge obtained from classes (and literatures) can finally be perceived.

Furthermore, through one of my courses in M-BIT in University of Twente, I have received an experience in using a low-code development platform from Mendix. From the course, the goal of the assignment was to integrate business processes between multiple working groups in the class. However, the benefit from that assignment was finally felt out when the company's CEO assigned me with the additional task of investigating the low-code approach of developing mobile apps.

Having a hold on developing web applications using that platform, I was able to immediately propose Mendix as the tool to investigate the possibilities for this project. This advantage has benefitted towards accelerating the pace of the investigation product delivery because if I had not possessed the experience using Mendix previously then it would take more time to learn and deliver the result.

Finally, Usability Testing also plays an integral part in this internship assignment, since it is also meant to address one of the project goals. Similar to the earlier discussion about development methods, Usability Testing was also being used in my previous bachelor thesis in order to measure how the user perceived the developed mobile application. From the previous experience in conducting the test, the participants consisted of peers from the same study program where potential bias was higher and critical feedback as well as thorough "try-to-break-the-thing" attempts were low. However, in this project, the participants consisted of people with a more diverse characteristics and professional background. This distinction has raised the validity and the perceived impact of the test result, which has further proven the result of both the developed mobile app along with the usability test, at least for the practical level, and can be proposed again for future projects (e.g. comparing this prototype design with potential update for the production version in the future). Overall, this internship assignment has allowed me to extend my experience in software engineering and further diminish the gaps between academic setting and industrial practice.

3.3. Future Implications

I have been aspiring to work in The Netherlands after I finish my master study and this opportunity has given me insights about the working environment in the country, thus has become the stepping stone for me to further strengthen my initial intentions. Another implication is that, from this opportunity, I have gained the knowledge about the future plan that the company FarmResult is trying to achieve in order to deliver higher quality IT solutions to their clients in the industry. This has convinced myself that it is essential for me to further enhance my personal skill sets and capabilities in order for me to gain another opportunity to participate in the future projects. Since the plan will need a thorough expertise in the field of both software engineering and data science, I believe that upon finishing my study I can contribute my experience in software development I gained so far along with the knowledge about data science that I gained from my specialization in the master study of Business IT to the industry.

Lastly, I believe that both this internship and the assignment is only just a start for something bigger. This internship is a stimulant to jumpstart my career journey towards greater opportunities in professional and international business environments. Furthermore, I also believe that the outcome of this assignment can be further developed to incorporate functionalities that have a more positive impact to the business and deserves to be delivered to both the existing as well as the potential clients. Because after all, this assignment has been initiated from the very beginning with the aim of bringing and enhancing the business value of both the company FarmResult and their clients.

4. References

Payne, R. (2019). *Beginning App Development with Flutter: Create Cross-Platform Mobile Apps.* Dallas, USA: Apress, Berkeley, CA.

Chang, Y.-H., & Ko, C.-B. (2017). A Study on the Design of Low-Code and No Code Platform for Mobile Application Development. *International Journal of Advanced Smart Convergence Vol.6*, 50-55.

Distante, D., Pedone, P., Rossi, G., & Cafora, G. (2007). Model-Driven Development of Web Applications with UWA, MVC and JavaServer Faces . *ICWE 2007* (pp. 457–472). Berlin, Heidelberg: Springer.

Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017). SCRUM Model for Agile Methodology. *International Conference on Computing, Communication and Automation (ICCCA2017)* (pp. 864-869). Greater Noida, India: IEEE.

Wicaksono, S. A., Firdausy, D. R., & Saputra, M. C. (2018). Usability Testing on Android Application of Infrastructure and Facility Reporting Management Information System. *Journal of Information Technology and Computer Science*, 184-193.

Mew, L., & Field, D. (2018). A Case Study on Using the Mendix Low Code Platform to Support a Project Management Course. *EDSIG* (pp. 1-11). Norfolk, VA: ISCAP (Information Systems & Computing Academic Professionals).

Inukollu, V. N., Keshamoni, D. D., Kang, T., & Manikanta, I. (2014). Factors Influencing Quality of Mobile Apps: Role of Mobile App Development Life Cycle. *International Journal of Software Engineering & Applications (IJSEA)*, 15-34.

Sauro, J. (2011, March 21). *WHAT IS A GOOD TASK-COMPLETION RATE?* Retrieved from MeasuringU: https://measuringu.com/task-completion/

5. Appendix

5.1. Usability Testing FarmResult B.V.

Usability Testing – Effectiveness & Efficiency

Article : PoultryResult - Mobile App
Company : FarmResult
Participant : Joey Lee Bakker
Role : Technical Implementation Consultant

Task Level Success & Time
Please insert the success and time consumed in the provided space.

Group	No	User Story	Success	Time
General	GR 2.	See and select their farm site location	1	5
General	GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	1	30s
Mortality	DO 1.1	See current round animal mortality of the same or another day	1	3s
	DO 1.2	Enter new entry of the chicken flock's mortality observation	1	17s
	DO 1.3	Edit entry of the chicken flock's mortality observation	1	19s
	DO 1.4	Delete entry of the chicken flock's mortality observation	1	15s
Feed	DO 2.1	See current round animal feed of the same or another day	1	12s
	DO 2.2	Enter new entry of the animal feed	1	20s
	DO 2.3	Edit entry of the animal feed	1	15s
	DO 2.4	Delete entry of the animal feed	1	3s
	DO 2.5	Copy previous entry when animal feed observation for current day is not yet present	1	3s
Weight	DO 3.1	See current round animal weight of the same or another day	1	6s
	DO 3.2	Enter new entry of the animal weight observation	1	5s
	DO 3.3	Edit entry of the animal weight observation	1	22s
	DO 3.4	Delete entry of the animal weight observation	1	2s
Eggs	DO 7.1	See current round egg production of the same or another day	1	40s
	DO 7.2	Enter new entry of the egg production observation	1	22s
	DO 7.3	Edit entry of the egg production observation	1	15s
	DO 7.4	Delete entry of the egg production observation	1	2s
General	GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location	1	5s
Climate Data Intervention	CO 2.1	See manual house climate observations	1	4s
	CO 2.2	Enter new manual house climate observations	1	25s
	CO 2.3	Edit entry of the manual house climate observations	1	10s
	CO 2.4	Delete entry of the manual house climate observations	1	2s

Remarks:

- Re-consider the placement of the button to switch farm site location, a good idea would be to also include it in the side menu along with the "Dashboard" and "Daily Observations". Otherwise, the user will spend a lot of time trying to figure out how to switch farm site location.
- In the future, it's best to also consider white-box test to avoid false input to the input forms.

Usability Testing – Satisfaction

Article : PoultryResult - Mobile App
Company : FarmResult
Participant : Joey Lee Bakker
Role : Technical Implementation Consultant

Task Level Satisfaction

Please give check mark [✓] in the response box which you think is appropriate.

No	Task	Response				
		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
GR 2.	See and select their farm site location					✓
GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location					✓
DO 1.1	See current round animal mortality of the same or another day					✓
DO 1.2	Enter new entry of the chicken flock's mortality observation					✓
DO 1.3	Edit entry of the chicken flock's mortality observation					✓
DO 1.4	Delete entry of the chicken flock's mortality observation					✓
DO 2.1	See current round animal feed of the same or another day					✓
DO 2.2	Enter new entry of the animal feed					✓
DO 2.3	Edit entry of the animal feed					✓
DO 2.4	Delete entry of the animal feed					✓
DO 2.5	Copy previous entry when animal feed observation for current day is not yet present					✓
DO 3.1	See current round animal weight of the same or another day					✓
DO 3.2	Enter new entry of the animal weight observation					✓
DO 3.3	Edit entry of the animal weight observation					✓
DO 3.4	Delete entry of the animal weight observation					✓
DO 7.1	See current round egg production of the same or another day				✓	
DO 7.2	Enter new entry of the egg production observation					✓
DO 7.3	Edit entry of the egg production observation					✓
DO 7.4	Delete entry of the egg					✓

	production observation					
GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location					✓
CO 2.1	See manual house climate observations					✓
CO 2.2	Enter new manual house climate observations					✓
CO 2.3	Edit entry of the manual house climate observations					✓
CO 2.4	Delete entry of the manual house climate observations					✓

Wierden, 20 October 2020

Joey Lee Bakker
Technical Implementation Consultant

5.2. Usability Testing PT Sreeya Sewu Indonesia

Usability Testing – Effectiveness & Efficiency

Article : PoultryResult - Mobile App
Company : PT Sierad Produce / Sreeya Sewu Indonesia
Participant : Melvin Winata
Role : Business Development Manager

Task Level Success & Time
Please insert the success and time consumed in the provided space.

Group	No	User Story	Success	Time
General	GR 2.	See and select their farm site location	1	10s
General	GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	1	10s
Mortality	DO 1.1	See current round animal mortality of the same or another day	1	7s
	DO 1.2	Enter new entry of the chicken flock's mortality observation	1	20s
	DO 1.3	Edit entry of the chicken flock's mortality observation	1	13s
	DO 1.4	Delete entry of the chicken flock's mortality observation	1	14s
Feed	DO 2.1	See current round animal feed of the same or another day	1	5s
	DO 2.2	Enter new entry of the animal feed	1	21s
	DO 2.3	Edit entry of the animal feed	1	13s
	DO 2.4	Delete entry of the animal feed	1	3s
	DO 2.5	Copy previous entry when animal feed observation for current day is not yet present	1	5s
Weight	DO 3.1	See current round animal weight of the same or another day	1	7s
	DO 3.2	Enter new entry of the animal weight observation	1	9s
	DO 3.3	Edit entry of the animal weight observation	1	8s
	DO 3.4	Delete entry of the animal weight observation	1	2s
Eggs	DO 7.1	See current round egg production of the same or another day	1	14s
	DO 7.2	Enter new entry of the egg production observation	1	25s
	DO 7.3	Edit entry of the egg production observation	1	8s
	DO 7.4	Delete entry of the egg production observation	1	2s
General	GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location	1	5s
Climate Data Intervention	CO 2.1	See manual house climate observations	1	27s
	CO 2.2	Enter new manual house climate observations	1	34s
	CO 2.3	Edit entry of the manual house climate observations	1	6s
	CO 2.4	Delete entry of the manual house climate observations	1	4s

Remarks:
- Already covers basic functionalities
- See daily observations, nice to be able to see it in a tabular fashion, weekly/multiple days at once
- Add remark to the camera feature

Usability Testing – Satisfaction

Article : PoultryResult - Mobile App
Company : PT Sierad Produce / Sreeya Sewu Indonesia
Participant : Melvin Winata
Role : Business Development Manager

Task Level Satisfaction

Please give check mark [✓] in the response box which you think is appropriate.

No	Task	Response				
		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
GR 2.	See and select their farm site location				✓	
GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location				✓	
DO 1.1	See current round animal mortality of the same or another day				✓	
DO 1.2	Enter new entry of the chicken flock's mortality observation				✓	
DO 1.3	Edit entry of the chicken flock's mortality observation					✓
DO 1.4	Delete entry of the chicken flock's mortality observation					✓
DO 2.1	See current round animal feed of the same or another day				✓	
DO 2.2	Enter new entry of the animal feed				✓	
DO 2.3	Edit entry of the animal feed					✓
DO 2.4	Delete entry of the animal feed					✓
DO 2.5	Copy previous entry when animal feed observation for current day is not yet present					✓
DO 3.1	See current round animal weight of the same or another day				✓	
DO 3.2	Enter new entry of the animal weight observation				✓	
DO 3.3	Edit entry of the animal weight observation					✓
DO 3.4	Delete entry of the animal weight observation					✓
DO 7.1	See current round egg production of the same or another day				✓	
DO 7.2	Enter new entry of the egg production observation				✓	
DO 7.3	Edit entry of the egg production observation					✓
DO 7.4	Delete entry of the egg					✓

	production observation					
GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location				✓	
CO 2.1	See manual house climate observations				✓	
CO 2.2	Enter new manual house climate observations				✓	
CO 2.3	Edit entry of the manual house climate observations					✓
CO 2.4	Delete entry of the manual house climate observations					✓

Indonesia, 20 October 2020

Melvin Winata
Business Development Manager

5.3. Usability Testing Schoot Uiterkamp

Usability Testing – Effectiveness & Efficiency

Article : PoultryResult - Mobile App
Company : Schoot Uiterkamp
Participant : Mr. Jeroen Schoot Uiterkamp
Role : Owner

Task Level Success & Time
Please insert the success and time consumed in the provided space.

Group	No	User Story	Success	Time
General	GR 2.	See and select their farm site location	1	10
General	GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location	1	35
Mortality	DO 1.1	See current round animal mortality of the same or another day	1	10
	DO 1.2	Enter new entry of the chicken flock's mortality observation	1	20
	DO 1.3	Edit entry of the chicken flock's mortality observation	1	15
	DO 1.4	Delete entry of the chicken flock's mortality observation	0	10
Feed	DO 2.1	See current round animal feed of the same or another day	1	10
	DO 2.2	Enter new entry of the animal feed	1	25
	DO 2.3	Edit entry of the animal feed	1	20
	DO 2.4	Delete entry of the animal feed	0	10
	DO 2.5	Copy previous entry when animal feed observation for current day is not yet present	1	5
Weight	DO 3.1	See current round animal weight of the same or another day	1	10
	DO 3.2	Enter new entry of the animal weight observation	1	5
	DO 3.3	Edit entry of the animal weight observation	1	6
	DO 3.4	Delete entry of the animal weight observation	0	8
Eggs	DO 7.1	See current round egg production of the same or another day	0	0
	DO 7.2	Enter new entry of the egg production observation	0	0
	DO 7.3	Edit entry of the egg production observation	0	0
	DO 7.4	Delete entry of the egg production observation	0	0
General	GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location	1	8
Climate Data Intervention	CO 2.1	See manual house climate observations	1	10
	CO 2.2	Enter new manual house climate observations	0	0
	CO 2.3	Edit entry of the manual house climate observations	0	0
	CO 2.4	Delete entry of the manual house climate observations	0	0

Remarks:

Overall
The app should focus on daily observations and the things a farmer would like to know when he/she is inside a shed. Above the icons for all daily observations, place the temperature, RH at that moment in the shed. Remove extra steps (button click) to go to a task. Make it as simple as possible.

Focus timeframe
When the app is for use inside the shed, focus should be on current data. The timeframe should be set for:

- Feed/Water on last 5-7 days
- Mortality and Weight on last 14 days

Mortality, Growth, Feed

On top of the entry screen for adding daily observation, there should be a summary of the value.

- Mortality: Total %, Total in numbers
- Growth: daily average growth
- Feed: total feed per broiler

Table list instead of graphs

for:

- Additives
- Vaccinations/Medication

Date	Substance	Amount	UoM
12-01-2020	Vitamine C	10	kg
03-02-2020	Vitamine comp	15	kg

Usability Testing – Satisfaction

Article : PoultryResult - Mobile App
Company : Schoot Uiterkamp
Participant : Mr. Jeroen Schoot Uiterkamp
Role : Owner

Task Level Satisfaction

Please give check mark [✓] in the response box which you think is appropriate.

No	Task	Response				
		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
GR 2.	See and select their farm site location			✓		
GR 3.	Access and manage Animal Daily Observations of the chicken flocks in each house in a round of the selected farm site location			✓		
DO 1.1	See current round animal mortality of the same or another day				✓	
DO 1.2	Enter new entry of the chicken flock's mortality observation				✓	
DO 1.3	Edit entry of the chicken flock's mortality observation				✓	
DO 1.4	Delete entry of the chicken flock's mortality observation			✓		
DO 2.1	See current round animal feed of the same or another day				✓	
DO 2.2	Enter new entry of the animal feed				✓	
DO 2.3	Edit entry of the animal feed				✓	
DO 2.4	Delete entry of the animal feed			✓		
DO 2.5	Copy previous entry when animal feed observation for current day is not yet present				✓	
DO 3.1	See current round animal weight of the same or another day				✓	
DO 3.2	Enter new entry of the animal weight observation				✓	
DO 3.3	Edit entry of the animal weight observation				✓	
DO 3.4	Delete entry of the animal weight observation			✓		
DO 7.1	See current round egg production of the same or another day				✓	
DO 7.2	Enter new entry of the egg production observation				✓	
DO 7.3	Edit entry of the egg production observation				✓	
DO 7.4	Delete entry of the egg				✓	

	production observation					
GR 4.	Access and manage House Climate Observations of each house in a round of the selected farm site location			✓		
CO 2.1	See manual house climate observations			✓		
CO 2.2	Enter new manual house climate observations			✓		
CO 2.3	Edit entry of the manual house climate observations			✓		
CO 2.4	Delete entry of the manual house climate observations			✓		

Netherlands, 28 October 2020

Jeroen Schoot Uiterkamp
Owner