Progress Report

Contents

Problem statement summary

Increasing agricultural yields in a sustainable fashion has become increasingly critical to feed the world's population [1]. Agricultural Extension Services (AES) are by organisations such as local governments and NGOs in developing countries to arm smallholders with agricultural knowledge and new techniques [2]. Out of all forms of AES, SMS-based ones remain the best [3], However, their predominant 'press 1 for x' or 'subscribe to alerts on z' approach limits interaction and confines advice to specific stages of the agricultural cycle in specific locations.

Compounding this, studies are often carried out to optimise message phrasing- resulting in resource wastage and potential confusion among smallholders [4]. Moreover, few SMS-AES harvest the wealth of research and data that farms in developed countries have begun to use as part of the Agriculture-4.0 revolution.

Using a novel approach combing modern Chatbot and Expert System and Adaptive Decision Support System functionality, this project will provide tailored responses world-wide for all farmers for any part of the agricultural cycle, leveraging worldwide data to bring the Agriculture-4.0 experience to smallholders.

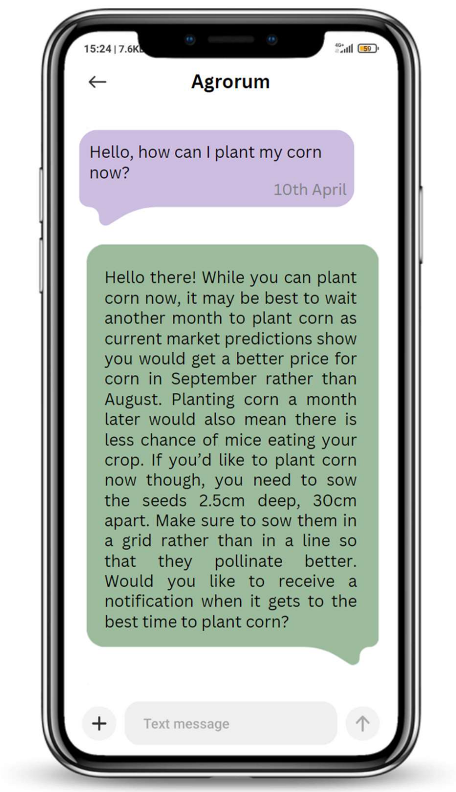


Fig 1. Example input and output of system. Note the use of market price prediction to provide an informed personalised response. Please see end of 'Implementation' section for a more in-depth discussion on sample inputs and outputs.

Glossary

Smallholding A small farm, usually family owned, with an average size of 5 acres, although this varies greatly from country to country. For the purposes of this project, any farm under 50 acres will be considered a smallholding. Smallholdings are also prevalent in developed countries, with the European Union calling them 'the backbone of European farming" [5].

Smallholder A person who owns, or labours in, a smallholding.

Agricultural Extension Service A service which provides advice, knowledge and training to farmers with the aim of educating and empowering them, often supplied by government agencies, although independent groups (such as Precision Development) or NGOs also frequently develop their own. International Development organisations such as the World Bank or Food and Agriculture Organisation (FAO) often provide large amounts of support for Agricultural Extension Services in developing countries. [2]

Agriculture-4.0 Also known as 'digital agriculture'. Often refers to connecting the agricultural world with the IoT, including the use of robotics and AI to perform data-driven management of the farm - key in increasing profitability and sustainability. An example of an Agriculture-4.0 system is using temperature and moisture sensors in the soil, as well as future weather predictions, to irrigate a farm, however modern Agriculture-4.0 tools are usually more complex and interconnected. [6]

Developing Country There is no clear definition of what makes a country a developing country, although a common metric is corresponding a country to its Human Development Index (HDI). This project uses 'developing countries' as a broad term to describe countries which would benefit most from what this project sets out to deliver.

Crop Cycle The process most agricultural harvests go through. This project aims to provide support for the whole cycle.

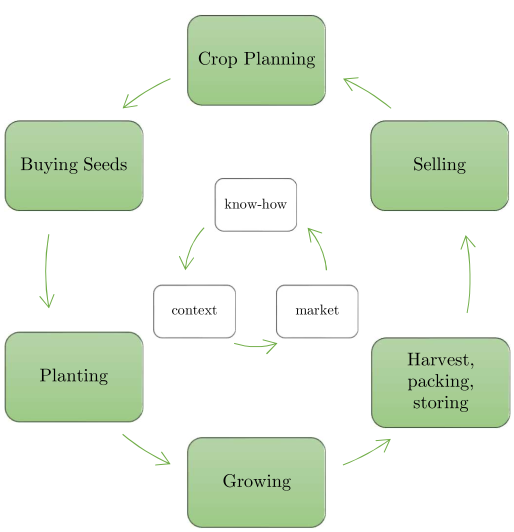


Fig 2. The agricultural crop cycle. Concept taken from [21]

Common Abbreviations

AES Agricultural Extension Service
FAO Food and Agriculture Organisation
ES Expert System
DSS Decision Support System
DM Decision Maker

Notes

Please refer to Appendix A at the end of this document for the Specification Document for this project. Bibliographies for this Progress Report and the Specification have been kept separate.

Introduction, and summary of research

The need for greater agricultural yields

The world's population is growing at an unprecedented rate, with the Food and Agriculture Organisation of the United Nations (FAO) predicting that the population will rise to 9.1 million by 2050 [7]. And yet even now, the agricultural sector struggles to cope [2]. By 2022, 9.2% of the global population were in a state of chronic hunger, and 9 million people still die of starvation each day [8], with food insecurity showing an alarming increase since 2015 [9] despite the UN Sustainable Development Goal 2 (zero hunger by 2030) [10]. To achieve this goal, agricultural yields need to be sustainably increased – especially given that increasing population and global warming decreases available farmland [1] [11]. In fact, if soil erosion in Africa continues, crop yields will be halved in 30-50 years' time if nothing changes [12].

Where can these yields be increased?

In developed countries, commercial farms have reached a state of limiting returns, where heavy investment only sees a minor improvement in already very efficient yields [13]. Smallholdings in developed countries benefit from easy access to technology such as smartphones and the internet, which allows them to access accurate agricultural knowledge at a click of a button [14]. Improving yields here is once again a case of investment into machinery and more resources. Much research and funding has been put into developing 'Agriculture 4.0' to increase yields on farms in developed countries [6], using big data to take the human guesswork out of farming, and allowing 'precision farming' – minimal resources, at the optimal time, for the maximum output, with emphasis on automating as much as possible.

Smallholdings in developing countries, on the other hand contain the greatest potential agricultural yield increase. Best practices and knowledge are difficult to transmit and obtain [15], and having this information alone increases yields [2] [14] [16], explaining the heavy investment into Agricultural Extension Services (AES). An optimal solution would harness the principles of Agriculture-4.0 – using data to inform decision making for optimal results - and AES – providing this information to the smallholders. [17]

How can yields be increased in developing countries?

The transfer of knowledge and agricultural skills in developing countries are either done by word-of-mouth across communities, or through AES. This can take many forms, from Agricultural Extension Workers travelling to communities, or televised weather forecasts and how they may affect crops. There are many different types of AES, and their pros and cons are listed in the section below. Of all AES however, the most promising are SMS-based approaches [18] [19], due to their low-cost, high reach (most farmers have feature phones that support sending and receiving texts) and potential for personalisation, which may reduce the marginalisation of certain groups of people, such as women and youths, in accessing agricultural information [11]. Current SMS-based implementations are either person based (agricultural extension worker sits in a room and responds to messages from farmers like customer support), which are limited by the number of Agricultural Extension Workers [16], or automated. Of those that are automated, they are usually limited to Push systems (Government texting cocoa farmers that a certain pest is coming) or Pull systems (user goes through a series of options to get information they want) [20].

Existing AES [21]

Transmission Type	Drawbacks	Benefits
Radio	People may not have radio Have to tune in at specific time Cannot choose what information to hear Not specific to region	Sometimes there are channels dedicated to agricultural announcements
Newspaper	Cost of purchasing newspaper Not specific to smallholding region Cannot choose what information they need	Reliable source
TV	People may not have TV Have to tune in at specific time Cannot choose what information to hear Not specific to region	Reliable source
SMS	Relies on people having phones with sms-capabilities May not support local dialects May only be Push or Pull	Can be tailored to user Low cost
Voice	Requires phone	Overcomes literacy barriers
Agricultural extension worker	Face to face personalised info	No accountability, may discriminate.

Problems with Existing SMS-Based AES

As mentioned previously, non-automated SMS-based AES are limited by the number of available Agricultural workers, and automated ones are limited by their implementation. Furthermore, the UI tends to be clunky and unintuitive- something that has been a great barrier for adoption due to the general mistrust of technology and low technological skill [11].

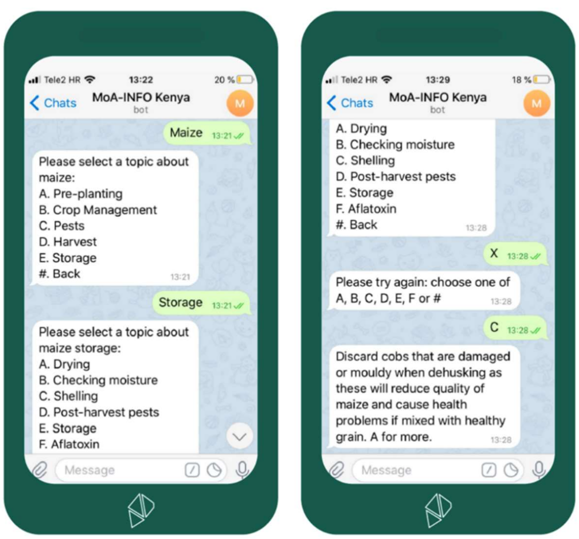


Figure 3: Example of existing SMS-Based AES. This example is taken from Precision Development (PxD) [38]

Current developed AES also tend to focus on a specific group of farmers, in one region, doing one type of farming, for only one part of the crop cycle - very little work has been done to produce an integrated environment [22]. This limits the effectiveness of AES as there is no one place to go for all smallholder knowledge.

Commented [IS1]: Agricultural market place - information is a commodity, competing for best information and selling at price, people can't pay + unreliable information (like medical marketplaces back in galens time) vs one standardised source of information (nhs)

How can the negatives of existing SMS-based AES be tackled?

Existing AES problems	Possible solutions
Lack of existing systems that are both push and pull	Allow the user to ask questions and respond, as well as send notifications [23]
Non-intuitive UI of service	Do not use 'text 1 for option A' kind of format. Allow any kind of natural text from the user, and let the system use NLP to put it into a form the system can understand.
Mistrust of technology	Allow free text, and respond using human-like speech with intelligent suggestions.
Lack of one place for all knowledge	Support whole agricultural cycle, for any crop, anywhere in the world.
Benefits of Agriculture 4.0 not used	Leverage datasets and APIs to provide intelligent and informed responses to the user,
Lack of personalisation	Calculate user characteristics such as literacy level and gender to tailor response and be able to reply in any language.
Not suitable for illiterate smallholders, or smallholders that use a very specific dialect [11]	Include the option for an automated telephone call that includes Interactive Voice Response (IVR) such that the same conversation thar can be held by text can be held over the phone. And factor in research for low resource languages to incorporate models for every single dialect.

Objectives
Given the research conducted, the following objectives have been decided on for the project. All objectives refer to what should be done by the project Presentation.
Must
The system must:
- Allow free text input on the side of the user (not restricting them to press 'A' for 'X').
- Respond in a human-like fashion, rated at least 80% on friendliness.
- Give accurate suggestions at least 90% of the time.
- Score at least 8/10 on ease of use.
- Be able to answer questions on all parts of the agricultural cycle (crop planning, buying seeds, planting, growing, harvest/packing/storing, selling).
- Be able to answer questions about at least five different staple crops which are the most common cash crops globally (maximum global reach).

Commented [IS2]: Which ones? Write about in final report

- Be able to send a triggered notification on at least one aspect (e.g. weather).
- Be able to send a frequency notification on at least one aspect (e.g. reminder every week to check on plants).

Should

The system should:

- Be able to take input in at least 2 different languages.
- Be able to respond in at least 2 different languages.
- Not lose at least 90% of tailoring and friendliness when responding in a different language.
- Give recommendations tailored specifically to a user's country for at least one aspect of the crop cycle for at least one crop.
- Give recommendations tailored specifically to a user's authorship characteristics (such as literacy and age) to improve the chances of a user implementing suggestions for at least one aspect of the crop cycle for at least one crop.

Could

If there is time, the system could:

- Provide responses tailored specifically to a user's region for at least one aspect of the crop cycle for at least one crop.
- Use previous message history of user to determine the user's knowledge of specific areas and tailor level of detail in response to that for at least one aspect of the crop cycle for at least one crop.
- Use ML to not only answer questions, but to also give insights (such as figure 1) for at least one aspect of the crop cycle for at least one crop.

Would

The system could be extended to:

- Provide information specifically tailored to the user's smallholding for at least one aspect of the crop cycle for at least one crop.
- Have an IVR option.
- Decide what notifications to send to user based on past message history for at least one aspect of the crop cycle for at least one crop.
- Recognise words despite spelling mistakes.
- Be able to respond to queries in local dialects.

Will Not

The system will be limited to providing information only about crops, and not involve livestock. The majority of AES focus around crops, and this project aims to extend that research.

Implementation

Overall Project Design

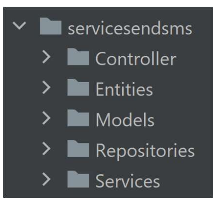
This project is developed in Kotlin due to its conciseness and readability, minimising the need for extensive boilerplate code. Adopting a microservice architecture further enhances the systems flexibility, extensibility, and fault-tolerance compared to a monolithic structure.

This approach is especially useful for incremental development across different services, potentially in distinct branches, while keeping the whole system working, lending itself to trunk-based development (TBD) where small self-contained feature branches are taken from the master branch (the trunk) and merged back in. This strategy facilitates concurrent development on multiple fronts, mitigating the risk of one bug/blocker in one area bringing the entire project to a halt. Additionally, it guarantees there is always one place where the project works (the trunk).

The project is hosted on GitHub, providing the flexibility to work on different machines and ensuring storage beyond a single device. Finally, the entire microservice project is built by Gradle - its efficiency in multi-module projects due to its build speed and customisable dependency selection rules that can be declared once and used project wide makes it perfect for the project structure.

SMS-Service

The SMS-Service has been implemented using GatewayAPI as it supports sending messages world-wide from UK shortcodes, unlike other SMS-Services such as Azure Communications.



The project currently has two microservices dealing with SMS - service-send-sms and service-receive-sms. Each service has 2 main packages associated with it - main, for the source code, and test, for test writing. In main, there are the following folders: Controller, Entities, Repositories, Models and Services. Entities, Models and Repositories are discussed in more depth in the database section. The Controller and Services discussed here may be modified as development progresses to accommodate new features.

The Controller handles incoming requests, processes them by interfacing with other services if necessary, and then returns as a response. In Service-Send-SMS, it looks like the following:

```
Jacqueline Dobreva-Skevington *
@RestController("/rest")
class SendSMSController(
    val smsSendService: SMSSendService,
) : Logging {

    Jacqueline Dobreva-Skevington *
    @PostMapping("/mtsms")
    fun sendSMS(@RequestBody resource: SendMessageDTO): ResponseEntity<ResponseDTO> {
        logger().info("Sending message with ID ${resource.userref}")

        val sms = smsSendService.sendMessage(resource)

        return ResponseEntity.status(HttpStatus.OK).body(sms)
    }

}
```

Figure 4: SMS-Send-Service controller

Here, the controller simply calls the smsSendService, and, for now, always returns a response entity with status OK. Logging functionality has been defined in the ‘common' service.

The SMS send service is defined in the Services folder as such:

```kotlin
@Service
class SMSSendService(
    val userRepository: UserRepository,
    val messageRepository: MessageRepository
) : Logging {
    final val token = "QWg_52T4ToSxD4YPZnwWpb7PT7HiBJx0I-kaHSnNNdrTgp9_XRdzg3gHI51tu2h6"
    val uri = URI.create("https://gatewayapi.com/rest/mtsms?token=$token")

    val restTemplate: RestTemplate by lazy {
        RestTemplate()
    }

    @Transactional
    fun create(message: NewMessageDTO): ResponseDTO? {
        logger().info("Creating a new sms resource.")
        val user = userRepository.findByIdOrNull(
            UUID.fromString(message.userId)) ?: throw Exception("Could not find user ${message.userId}")
        val m = Message(
            message.message,
            LocalDateTime.now(),
            MessageType.OUTGOING,
            MessageStatus.PENDING,
            LanguageCode.EN
        )
        return messageRepository.save(m).let{
            it.user = user
            sendMessage(it.toMessageDTO())
        }
    }

    fun sendMessage(messageDTO: SendMessageDTO): ResponseDTO?{
        return restTemplate.postForEntity(uri, messageDTO, ResponseDTO::class.java).also {
            if(!it.statusCode.is2xxSuccessful) throw Exception("Error ${it.statusCode.value()}")
        }.body
    }
}
```

Figure 5: SMS-Send-Service Service

It creates a message with the user it's sending the message to and saves it to the database. It then attempts to send the message and throws an error if necessary.

The microservice also has a Utilities folder, containing mappers such as this one:

```kotlin
fun Message.toMessageDTO(): SendMessageDTO{
    return SendMessageDTO(
        message = this.message,
        sender = "447418372559",
        recipients = listOf(
            RecipientDTO(
                this.user!!.phoneNumber.toLong()
            )
        )
    )
}
```

Figure 6: SMS-Send-Service Message to SendMessageDTO mapper utility function

Logging functionality has been defined in the 'common' service as follows:

```
/**
 * Extension that returns a logger object that can be used by any class implementing
 * the [Logging] interface.
 */
Jacqueline Dobreva-Skevington
fun <T : Logging> T.logger(): Logger = getLogger(javaClass)
```

Figure 7: Part 1 of the logging extension

```
/**
 * Marker interface that can be used on classes that want to use the
 * logging extension.
 */
Jacqueline Dobreva-Skevington
interface Logging
```

Figure 8: Part 2 of the logging extension

Sending an example successfully delivered SMS creates the following entry in the message table:

	id (UUID)	language	message	sent_at	status	type	user_id
1	4f975f8d-2dca-4599-b6c7-c7f4…	EN	Test Message	2023-11-22 23:58:47.460688	DELIVERED	OUTGOING	0xEFBBBF30623100000000000000…

Figure 9: Part of message table. Note that the user_id was created arbitrarily just for demonstration purposes.

<u>Model</u>

Choosing Implementation: Decision Support Systems (DSS), Expert Systems (ES) and chatbots

A DSS is a computer program which allows businesses to make decisions more effectively [24]. It does this by looking through large amounts of data with each question and then synthesising it into easily understandable human form [25]. Many organisations use DSS, all the way from clinical DSS' which may help with diagnosis [26] to the investigation of sustainable transport in the metaverse [27]. Its underlying goal is to support the decision maker (DM), usually someone who helped design the DSS as they themselves know the rules behind how decisions should be made [28]. This is done by providing data and models to do with the problem, leaving the overall decision to the DM. Recent research on DSS has proposed Adaptive DSS which adapts to the DMs understanding of the concept and procedures behind the guidance the DSS is giving, and tailors its responses [29]. This is done through storing the user history and thereby extracting their knowledge level.

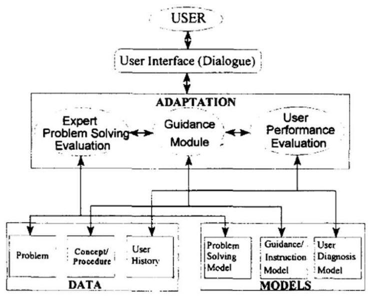


Figure 10: A Adaptive Decision Support model diagram taken from [29]

An ES' goal on the other hand is to provide a user with a decision rather than just seeking to improve the quality of a user's decision. The correctness of this decision should be close to, or better than, that of a real expert. The systems are built to replicate the knowledge and experience of experts (like Agricultural Extension Workers), and so therefore the users themselves usually have no effect on its development [28]. Unfortunately, due to knowledge acquisition bottlenecks to make the rules [3] and the overtly optimistic view of their capabilities at the time (1970s-90s) [29] these systems fell out of use. However, with the rapid advancement in AI, ML and NPL a resurgence of fields with links to ES (such as recommender systems) has come into itself. Furthermore, there has been research done with Explainable AI (XAI) ES - this aims to make the decision making more transparent to the user [30], bringing ES closer to DSS in that regard.

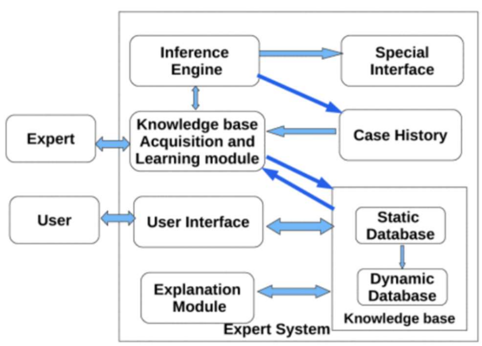


Figure 11: Basic Expert System Model

As this project aims to provide smallholders with answers to their questions while taking into account previous responses and potentially modelling the response to the users' characteristics (see project objectives), the project will use a primarily ES architecture to answer the questions and some aspects of Adaptive DSS research to model the system to the user. Explaining to the user the rationale behind each decision is less important as the primary goal of this project is to provide smallholders with the best possible answers to their question, and this information should only be provided if the user asks for it.

A closed-domain informative generative autonomous chatbot on the other hand is a chatbot with limited topic scope that provides information and generates responses dynamically based on a user's questions. This could provide more conversational responses and adapt to user preferences over time (like adaptive DSS); however, it may provide less accurate responses than an ES [31]. However, the more user-friendly experience that these chatbots can provide is very important in regard to this project as smallholders using technological AES are more likely to implement the suggestions of systems which are more human like.

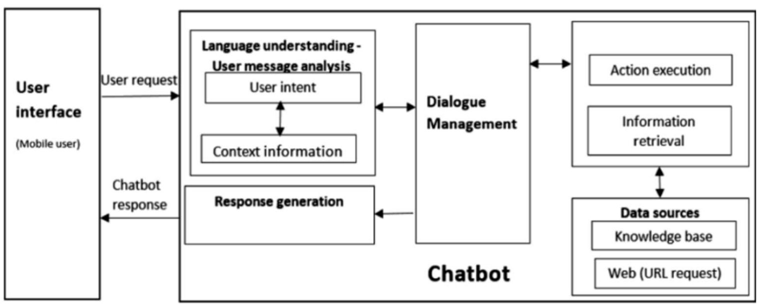


Figure 12: Internal model of a chatbot, taken from [39]

As this project aims to address the shortcomings of existing AES implementations by removing the negatives of more recent technological AES, and factoring in the positive aspects of extensions workers, a good middle ground would be a more conversational expert system that adapts to a user's knowledge level based on previous message history with the user (like adaptive DSS).

Overall, this project's proposed architecture of the model, combining all aspects mentioned above, is as follows:

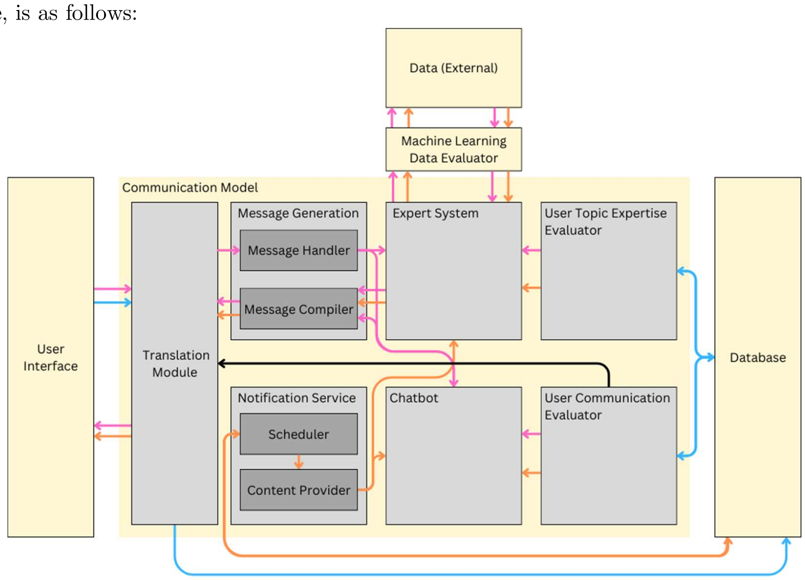


Figure 13: Overall planned architecture of system. Note that some Modules may be less fleshed out than discussion below in accordance with MoSCoW

There are three different asynchronous workflows represented in this diagram, represented by different coloured arrows. First, each module will be briefly discussed, before the three proposed workflows are explained.

1. The User Interface is the SMS-Service described at the start of the implementation section. This module is responsible for collecting user input and displaying the systems output to the user and facilitating a push-pull kind of AES.
2. The Translation Module is responsible for translating all incoming messages into English, so the rest of the modules will only have to be able to deal with English input. It is also then responsible for translating the output of the system into the language the user originally queried with, taking as input the systems response, as well as the user communication evaluator, to ensure that the translation is still modelled to the users' characteristics (ensuring calibrating is not lost in translation).
3. The Message Generator includes the Message Handler, which takes the raw stream of characters of the translated message provided by the Translation Module and then transforms it for the right input for the Chatbot and Expert System. The Message Compiler on the other hand, takes the response from the Chatbot and the Expert System, and then blends them together to create a user-friendly, and yet informative response.
4. The Expert System interfaces with the perceived knowledge base (made up of the Machine Learning Data Evaluator and Data) and The User Topic Expertise Evaluator. It is responsible for providing an expert response tailored by the user's knowledge of the associated topics, raw data, and other possible contributing factors.
5. The Chatbot interfaces with the User Communication Evaluator. It is responsible for producing a generic response based on a users' characteristics such as a user's literacy level.
6. The User Topic Expertise Evaluator takes the users message history from the database and calculates the user's expertise in various areas such as markets, pests, and certain types of crops. This allows the Expert System to tailor its level of detail and depth.
7. The User Communication Evaluator takes the users message history from the database and calculates various author characteristics such as literacy level to guide the chatbot with what sort of language it should use to ensure user is more likely to implement suggestions and understands what they are being told to do.
8. Data (External) and Machine Learning Data Evaluator together act as the Knowledge Base of the system. Data (External) will be made up of data that talks about the how, why, when, where and what of smallholding principles, as well as data which may influence the suggestion such as weather and market prices (historic and predicted). The Machine Learning Data Evaluator takes the formatted message from the Expert System (which in turn took it from the Message Handler), and the data from Data (External) and evaluates whether, despite this being the direct correct answer to the question, there could be a better answer.
9. The Notification Service calculates whether a user should receive notifications about a certain topic using the users message history (obtained from the database) based on whether they have asked directly, or whether they usually ask lots of questions about it. The Scheduler then decides whether the notification should be sent to the user at regular intervals (and if so, at what duration), or whether sending a notification should be sent based on a certain trigger. This information is saved back into the database. If

one of these events happen, then the Content Provider will provide a mock message to the Chatbot and Expert System in the form the message compiler to produce the necessary content.

10. The Database stores information about the user and their characteristics, the user's notification settings, and the users message history among other things. A more in-depth discussion about the database will be provided later.

The main workflow is the one denoted by pink arrows on the diagram. Here the message would be passed to the translation module to be translated into English. It will then go through the message handler to be standardised and be passed to the Expert System and Chatbot. The Expert System will output an expert response tailored to the user's knowledge level and the chatbot will provide a natural human response tailored to the users' characteristics. These two will be combined in the correct mix by the message compiler to produce a natural sounding informative response tailored to the user. This will then be re-translated by the translation module and checked a final time to make sure it is still suitable for the users' characteristics post-translation (see black arrow on diagram) and be sent back to the user.

Every time the user sends a message, it is saved in the database under the user's message history post-translation (blue arrows in diagram). Every ten messages, a task will run triggering the user's topic expertise and characteristics to be recalculated.

Finally, the scheduler will run routinely every night to check whether the notification settings need to be modified (orange workflow). The scheduler will also be triggered when a notification event occurs. When this happens the content provider in the notification service will make a mock message in standardised form and send it to the chatbot and expert system. From here on, the same workflow as the pink arrows is followed.

Example flow of first pathway
User: 'Hello, I would like to know about planting corn now' (in Swahili)
Translator: [translates message to English]
Message handler: keywords: {temporal: 10th April 2022, cycle: planting, crop: corn, location: null}
User expertise evaluator: {planting knowledge: high, corn knowledge: medium, location: UK}
Data: JSON file about planting corn in the UK in April
ML: taking predicted market prices in September (when corn will be ready), user should plant corn a month later instead to sell in October where market prices will be higher to make more of a profit.
User expertise evaluator: [knowledge about everything in JSON file including market prices]
Expert system: adds/ removes things as necessary depending on response and expertise evaluator.
Chatbot: [user says 'hello' and 'would like to know'] 'Hello there! Let me see if I can help. This is the information I have managed to find [+ generic information about corn]'
User Communication Evaluator: user has high literacy level so any vocabulary chatbot uses is okay.
Message Compiler: 'Hello there! While you can plant corn now, it may be best to wait another month to plant corn as current market predictions show you would get a better price for corn in September rather than August. Planting corn a month later would also mean there is less chance of mice eating your crop. If you'd like to plant corn now though, you need to sow the

seeds 2.5cm deep, 30cm apart. Make sure to sow them in a grid rather than in a line so that they pollinate better. Would you like to receive a notification when it gets to the best time to plant corn?'
Translator: English -> Swahili.
User Interface: SMS sent to user.

<u>Database Design</u>

The Database has been implemented using Spring Boot, JPA, MySQL and Flyway. Spring Boot follows the convention over configuration principle, thus reducing the amount of boilerplate code needed. Spring Boot also provides tool and features needed for test writing and is perfect for microservice architecture due to its ease of configuration. JPA provides a standard way to map Java/Kotlin objects to relational databases (Object Relational Mapping), simplifying database interactions. Furthermore, Spring Data JPA provides repository interfaces with ready to use CRUD operations, further reducing boilerplate code and increasing readability. Both Spring Boot and JPA are also very well documented and have an active community. The database tables are formulated in MySQL as Spring Boot and JPA allow for straightforward integration. Any Database migrations will be performed using Flyway as it provides a clear versioning system and easy rollbacks.

Below is an overview of the schema design so far. This will likely change to accommodate new features as the project is developed:

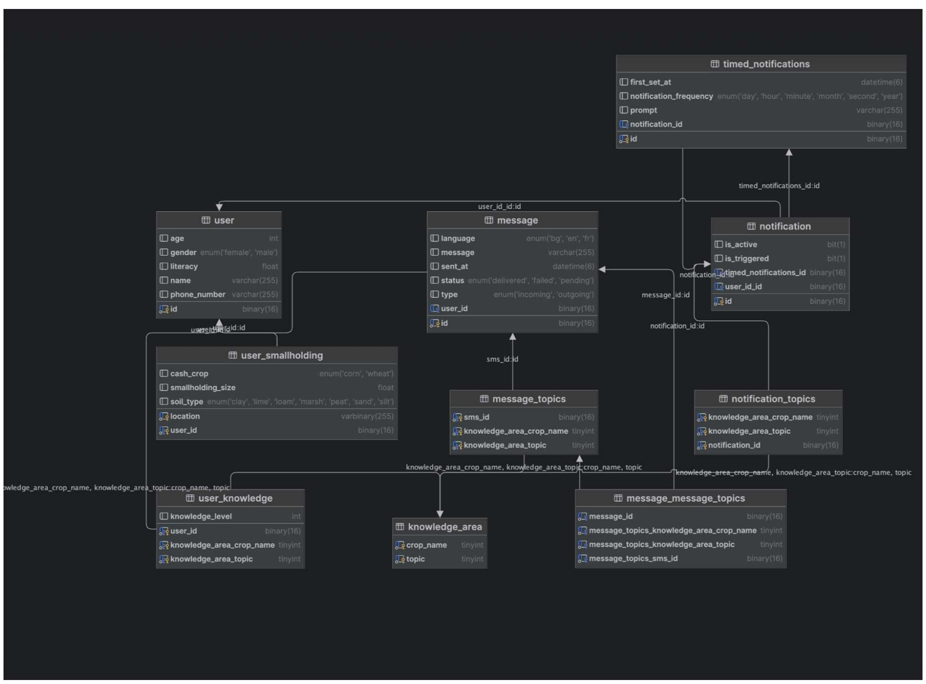


Figure 14: Current Schema structure

To implement this schema, Entities (Object Relationally Mapped to MySQL tables), Repositories (providing functionality for each table such as CRUD operations or data manipulating and retrieving functions such as find message by ID), and any associated Models (such as an Enum corresponding to common message states, or Data Transfer Objects (DTOs)) need to be defined. The entities have been designed so that the tables are in 4th Normal Form. Below is an example of an entity, and it's associated Repository and Models. As the schema, this will likely change by the time of the final submission to accommodate new features.

To continue with the send SMS microservice discussed earlier, here is the current message entity, it's associated repository, one of the enums used in the entity, and the MessageDTO:

```kotlin
Jacqueline Dobreva-Skevington *
@Entity
class Message(

    @Column
    val message: String,

    @Column
    val sentAt: LocalDateTime,

    @Enumerated(value = EnumType.STRING)
    @Column
    val type: MessageType,

    @Enumerated(value = EnumType.STRING)
    @Column
    val status: MessageStatus,

    @Enumerated(value = EnumType.STRING)
    @Column
    val language: LanguageCode,

    @OneToMany
    val messageTopics: List<MessageTopics> = listOf()
    )
{
    @Id
    @GeneratedValue(strategy = GenerationType.UUID)
    lateinit var id: UUID

    @ManyToOne
    var user: User? = null

}
```

Figure 15: Message Entity

```kotlin
@Repository
interface MessageRepository: CrudRepository<Message, UUID>
```

Figure 16: Message Repository

```
enum class MessageType {
    INCOMING,  //sent by user
    OUTGOING,  //sent by system
}
```

Figure 17: MessageType enum

```
class MessageDTO(
    val message_id: UUID,

    val content: String? = null,

    val timestamp: LocalDate? = null,

    val is_sent_by_user: Boolean? = null,

    val status: MessageStatus? = null, //PENDING, FAILED, DELIVERED etc

    val sent_at: LocalDate? = null
)
```

Figure 18: MessageDTO

Datasets

For an overall knowledge base, bulk csv data from https://www.fao.org/faostat/en/#home will be used. This database was chosen compared to others such as https://data.nal.usda.gov/ as it covered the greatest number of countries (245), has data from 1961 to now, and they frequently update their database. However, significant work will need to be done to whittle it down to only the datasets needed, as well as putting it into a usable format for the project.

In terms of other data which allow the project to fine tune its suggestions in accordance with the FAOSTAT data, WeatherAPI will be used for weather data, and more datasets will be found for Field boundaries and soil type based on location. Other statistics such as seed availability, predicted market prices, and pests and diseases can be extrapolated from the FAOSTAT database as a proof of concept. If there is time, individual APIs which will give more accurate predictions may be found for each such statistic.

The greatest challenge with datasets will be finding a way to efficiently extrapolate or make a knowledge base for the ES. More in depth research into this is planned for after submission of this document.

Testing plan

Each microservice will have several integration and unit tests associated with it to make sure all aspects of the controller are working as expected. This will allow for modular development so each microservice can be developed independently of one another, which will make the overall system easier to build. If there is time, a CI/CD pipeline will be set up to ensure that any new code is valid and passes all checks before it is deployed. Overall, once all individual aspects are finished and tied together, some in the field testing will be performed in an allotment setting to verify any suggestions it gives are valid. This may involve asking other members of the allotment to test the system out as well. In this case, ethics consent will be sought if necessary. If there is time, various smallholding groups such as 'the accidental smallholder' may be contacted to see if any members are willing to give the system a try. I will perform linguistic testing of the system.

Test-Driven Development was considered as a possible approach; however, this was discounted as the overhead of writing tests first was judged as too high, and the primary emphasis of the project is to extend specific concepts and theories.

Timetable plan and evaluation of progress against specification

The project began by setting up a JIRA board with brief tickets on overarching tasks that needed to be done within the first term. These became epics which were split up into subtasks. Start dates of tasks were not as stringently recorded as end dates, and minor tasks which took less than a day were not added onto the JIRA board to save time. Similarly, some issues remain 'unassigned' to a user as there is only one developer. Here is the JIRA board for this term:

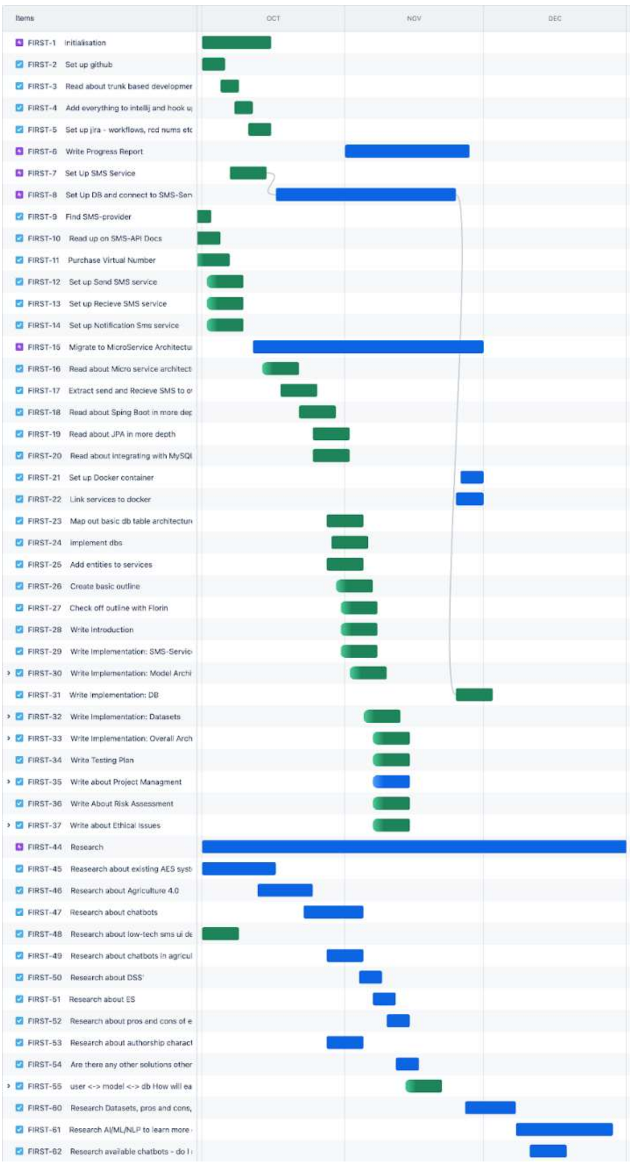


Figure 19: Term 1 JIRA Board

The JIRA board for next term will be created after this progress report is submitted, as this progress report was deemed as having a higher priority. However, the overall plan is to start off with the Translation module and expert system and then move on to the chatbot system. This was decided as these are key features in having something to show in the Presentation. Afterward, the Notification system and User Characteristics evaluators will be completed and the remainder of the time will be spent on the message compiler and the Machine Learning Data Evaluator. Throughout the term, materials for the Presentation will be prepared, and the final report will be written as to not leave too much for the end.

During weeks 1 and 2 of the project, much research was done on SMS-Based AES' in preparation for the specification submission. However soon after submission, the research field Agriculture 4.0 was discovered, where some authors had created decision systems for the crop cycle. It was decided that simply taking this and turning its output into SMS would not be enough for a 3rd year project, and therefore further research into decision systems and authorship characteristics were done. ES and Adaptive DSS looked the most promising in terms of extension, and User Characteristics fed nicely into the 'Adaptive' part of Adaptive DSS. Therefore, it was decided that a suitable 3rd project for a SMS-Based AES would be one which extends existing work on decision making (see 'Model' section) and also incorporates User Characteristics into it, specifically in an agricultural context - a field which, as far as the research currently done can tell, has had little to no research done into it.

Due to this change, more time was spent on research than developing than was planned for Term 1, however, due to the contingency weeks put into timetable for December, there is enough time to catch up in terms of development. Furthermore, the development work was different than specified in weeks 6-10 as the plan set out in the Specification was geared towards the original project. This updated work was specified in Figure 19.

To discuss progress and findings, weekly meetings were held with Dissertation Supervisor. During these discussions it was suggested that less emphasis should be put on testing and more emphasis should be put on having a novel solution. Therefore, some of the term 2 goals in the specification, and the majority of discussion in 'Legal, Social, Ethical and Professional Issues & Considerations' from the specification no longer apply. This additional time has been put to developing a more complex solution.

Risk assessment

Risk 2 (project not novel) was realised during this term, but as delt with as described in the section above. Risk 3 (access to GAIs) is also no longer applicable as a different model will be used internally. It is worth mentioning that the laptop used for development of this project refused to turn on for a week during term 1 (issue with docker consuming too many resources), but during this development on the project was still able to continue as the git repository was simply cloned on a different laptop (mitigation strategy of Risk 4 – personal computer corruption).

Overall, the updated Risk table has been created overleaf, with a column detailing how the severity and likelihood scores have changed and why.

Risk	Likelihood (/10)	Severity (/10)	Overall Risk (/100)	Reason for change/being added, if applicable	Mitigation
Dataset with information needed does not exist	3	7	21	The likelihood score has been decreased as over 1000 datasets have been found on agricultural knowledge and many APIs exist for other factors. The likelihood is still not zero as the final datasets haven't been selected yet, and during implementation, it may be discovered that the project requires more things than were accounted for.	Ensure Plan A and Plan B for each dataset for each specific type, which are similar enough to be able to switch between them fairly easily. Make implementation of modules as generic as possible.
Project is not novel or complex enough, or there has been significant existing research coming to same conclusions	7	9	63	Despite much research being done, Agriculture is an incredibly broad and deep research topic. It is entirely possible that a different area that covers what the project is attempting to do has been missed (see section above where AES and Agriculture 4.0 are mentioned), or that the right paper in a specific subtopic has not been found as the wrong search terms are being used. At writing the specification, the breadth of research was not known, hence the likelihood score	Keep thinking of possible novel features and leave enough time at the end to implement if necessary.

				has been adjusted upwards. Furthermore, the severity score has also been increased, as there is only 1 Term left, so less time to course correct.	
Personal computer corruption	3	3	9	This risk has been weighed down as everything is kept on GitHub, interesting research papers are printed off and kept in a binder, and permission has been obtained from work to use work laptop for project if necessary. Furthermore, all bulk writing work is done on Google Docs before it is migrated for formatting.	Keep pushing commits and printing out research. Keep using Google Docs for written pieces and start also backing them up by committing a copy alongside code to GitHub.
Database corruption	4	8	32	Severity has been decreased as less emphasis on training data.	Ensure nightly backups of data are done to remote location
Model does not work as expected	5	9	45	Likelihood has been decreased as model has been well fleshed out however severity has been increased as not much time left.	Same as Risk 2

Legal issues

This project does not have any legal issues, and ethics consent will be sought if external persons are involved in testing, if necessary.

Bibliography

[1] F. S. a. W. Wu, "“Soil and Crop Management Strategies to Ensure Higher Crop Productivity within Sustainable Environments,”," *Sustainability,* vol. 11, no. 5, p. 1485, Mar 2019.

[2] G. F. Jock R. Anderson, "Chapter 44 Agricultural Extension," *Handbook of Agricultural Economics,* vol. 3, pp. 2343-2378, 2007.

[3] P. G. G. J. T. D. P. Dario H. Galeon, "SMS-Based ICT Tool for Knowledge Sharing in Agriculture," *International Journal on Advanced Science Engineering Information Technology,* 2019.

[4] Precision Development, "precisiondev," PxD, [Online]. Available: https://precisiondev.org/our-learnings/learning/. [Accessed 10 10 2023].

[5] A. Gioia, "accesstolandeu," Eco Ruralis, April 2017. [Online]. Available: https://www.accesstoland.eu/IMG/pdf/comparative_analysis_of_small_farms_in_europe.pdf. [Accessed 2023 11 05].

[6] A. Suprem, N. Mahalik and K. Kim, "A review on application of technology systems, standards and interfaces for agriculture and food sector," *Comput. Stand. Interfaces,* vol. 35, pp. 355-364, 2013.

[7] FAO, "How to feed the world by 2050," in *high level expert forum*, 2009.

[8] D. Beasley, "World Food Programme," EFP, 24 September 2021. [Online]. Available: https://www.wfp.org/news/world-wealth-9-million-people-die-every-year-hunger-wfp-chief-tells-food-system-summit. [Accessed 12 10 2023].

[9] Food and Agriculture Organisation of the United Nations, International Fund for Agricultural Development, United Nations Children's Fund, UN World Food Programme, World Health Organisation, "The State of Food Security and Nutrition in the World," 2022.

[10] United Nations, "United Nations," [Online]. Available: https://www.un.org/sustainabledevelopment/hunger/. [Accessed 2023 10 10].

[11] C. M. B. M. H. J. M. J. H.-M. M. A. D. M. E. H. Enock Siankwilimba, "Bioecosystems towards sustainable agricultural extension delivery : effects of various factors," *Environment, Development and Sustainability,* 2023.

[12] N. W. M. P. W. B. A. M. E. L. Peter M. Kopittke, "Soil and the intensification of agriculture for global food security,," vol. 132, 7 August 2019.

[13] D. C. R.A. Fischer, "Issues for cropping and agricultural science in the next 20 years," *Field Crops Researcg,* vol. 222, pp. 121-142, 1 June 2018.

[14] C. K. I. A. Georgios Kountios, "Can the adoption of ICT and Advisory Services Be Considered as a Tool of Competitive Advantage in Agricultural Holdings? A Literature Review," *Agronomy,* vol. 13, p. 530, 12 February 2023.

[15] S. S. Baba, "Effectiveness of Knowledge Management Approaches on Agricultural Extension Services Delivery in Selected Districts of Northern Region," 2018.

[16] T. D. P. B. R. R. B. Angella Namyenya, "E-diary: a digital tool for strengthening accountability in agricultural extension," *Information Technology for Development,* vol. 28, no. 2, pp. 319-345, 2022.

[17] T. A. B. Mostafa Kamal, "Mobile Applications Empowering Smallholder Farmers: A Review of the Impact on Agricultural Development," *International Journal of Social Analytics,* pp. 36-50.

[18] A. P. H. S. G. D. G. Nicoletta Giulivi, "Getting the message out: Information and communication technologies and agricultural extension," *American Journal of Agricultural Economics,* 2022.

[19] P. G. G. J. T. D. P. Dario H. Galeon, "SMS-Based ICT Tool for Knowledge Sharing in Agriculture," *International Journal on Advanced Science Engineering Information Technology,* vol. 9, no. 1, pp. 342-349, 2019.

[20] E. C. Y. M. A. e. a. Isgren, "Are agricultural extension systems ready to scale up ecological intensification in East Africa? A literature review with particular attention to the Push-Pull Technology (PPT)," *Food Security,* vol. 15, p. 1399–1420, 2023.

[21] J. C. Aker, "Dial "A" for agriculture: a review of information and communicationtechnologies for agricultural extension in developing countries," *Agricultural Economics,* 2011.

[22] N. C. Eli-Chukwu, "Applications of Artificial Intelligence in Agriculture: A Review," *Engineering Technology & Applied Science Research,* vol. 9, no. 4, pp. 4377-4383, 2019.

[23] M. S. Bhandarkar, "A Case Study of Virtual Assistant and Effective Implementation for Assisting the Farmers," *Catalyst Research,* vol. 23, no. 2, pp. 434-447, August 2023.

[24] D. Power, "The Global Development Research Center," [Online]. Available: https://www.gdrc.org/decision/dss-types.html. [Accessed 1 11 2023].

[25] "Talking HealthTech," 15 10 2020. [Online]. Available: https://www.talkinghealthtech.com/glossary/decision-support-system. [Accessed 1 11 2023].

[26] M. Fairlie, "Business.com," 21 Feburary 2023. [Online]. Available: https://www.business.com/articles/decision-support-systems-dss-applications-and-uses/. [Accessed 1 11 2023].

[27] A. R. M. I. G. P. R. D. P. a. E. Ö. M. Deveci, "A Decision Support System for Assessing and Prioritizing Sustainable Urban Transportation in Metaverse," *Transactions on Fuzzy Systems,* vol. 31, no. 2, pp. 475-484, 2023.

[28] F. N. Ford, "Decision Support Systems and Expert Systems: A comparison," *Information & Managment,* vol. 8, no. 1, pp. 21-26, 1985.

[29] M. A. P. S. V. Bijan Fazlollahi, "Adaptive decision support systems," *Decision Support Systems,* vol. 20, no. 4, pp. 297-315, 1997.

[30] A. N. a. S. C. M. Ravi, "A Comparative Review of Expert Systems, Recommender Systems, and Explainable AI," *IEEE 7th International conference for Convergence in Technology (I2CT),* pp. 1-8, 2022.

[31] T.-C. H.-L. H. G.-J. H. a. M.-S. C. Hsu, "Effects of Incorporating an Expert Decision-Making Mechanism into Chatbots on Students' Achievement, Enjoyment, and Anxiety.," *Educational Technology & Society,* vol. 26, no. 1, pp. 218-31, 2023.

[32] R. M. Briones, "The Wage Gap Between Male," in *Outside Looking in : Gendered Perspectives in Work and Education*, Philippine Institute for Development Studies, 2019.

[33] nited Nations, "Sustainable Development Goals," [Online]. Available: https://www.un.org/sustainabledevelopment/hunger/.

[34] J. C. A. Christopher G. Harris, "Designing ICT for Agriculture (ICT4A) Innovations for Smallholder Farmers: The Case of Uganda," *Proceedings of the XIX International Conference on Human Computer Interaction,* pp. 1-9, 2018.

[35] J. F. M. V. B. N. L. M. Zhaoyu Zhai, "Decision support systems for agriculture 4.0: Survey and challenges," *Computers And Electronics in Agriculture,* vol. 170, 2020.

[36] R. G. Tony Dwi Susanto, "User acceptance of SMS-based e-government services: Differences between adopters and non-adopters," *Government Information Quarterly,* 2013.

[38] Precision Development, "precisiondev," PxD, [Online]. Available: https://precisiondev.org/pxds-user-communications-platform-paddy/. [Accessed 10 10 2023].

[39] X. Huang, "Chatbot: Design, Architecture, and Applications," University of Pennsylvania, 2021.

Appendix A (Original Specification)

<u>Enhancing Agricultural Extension Services for Small-Scale Farms in Low Income Countries : Exploring A Unified SMS approach</u>

Problem Statement/ Gap in Existing Provision

The problem of feeding the world's growing population in a sustainable fashion has been a longstanding problem. The Food and Agriculture Organisation of the United Nations (FAO) predicts that by 2050 the population will have risen to 9.1 billion people [7] yet agricultural productivity and yields in Low and Middle Income Countries (LMICs) have begun to lag behind that of Higher Income Countries (HICs) [21]. Focusing on improving agricultural yields, there has been heavy investment in agricultural extension services (AESs), however there are still significant issues with the ones implemented, including leaving women, who produce 60-80% of food crops, behind. By implementing a system with a more balanced user base, this project hopes to go some way to reducing the 20-30% gender pay gap in farming in MLICs, which is estimated could lift more than 1 in 7 people out of malnourishment. [32]

Furthermore, there is a growing concern to ensure that any efforts to bolster yields are not only profit driven, but also ensure sustainable food production that maintains ecosystems and is resilient to climate change – something that few initiatives, both in LMICs and HICs, do. [33]

SMS-based AESs have been advocated as a way forward in providing knowledge, agricultural education and access to new technologies to smallholders in LMICs [3], by minimising the weaknesses of other AESs, and exploiting the fact that most smallholders have access to feature phones [34]. However, existing SMS-based AESs frequently have a lot of issues, leading to them having little to no impact [11].

Finally there has been a huge corpus of research and data that have been produced in the western hemisphere, specifically to do with 'Agriculture 4.0' [35] – connecting farms up to the IoT and using the vast array of data being collected to inform farmers what the best decision to make is through an Adaptive Decision Support System (Adaptive DSS). However farms that have felt the benefit of this research have usually been industrial farms in HICs.

Objectives

The main aim of the project will be using the data and techniques from agriculture 4.0 research, but tailoring it for LMIC smallholders so they can benefit too.

Issues with existing SMS-based AESs will also be addressed through creating a global wide service implementing GAI. This will ensure that the system is both Push-Pull, allowing them to request any information they need, and will also use global data to better inform it's responses instead of only targeting a specific crop in a specific region.

Therefore, tying the above two together, the system ultimately should be an Adaptive DSS GAI chatbot which would have the ability to provide tailored, up-to-date information to users in MLICs.

Example conversation topics could include:

- When to plant certain crops
- Weather forecasts
- Market prices and economy information

More concretely, the chatbot should begin by asking the user their location to be able to provide tailored advice (by getting the smallholders soil type, topology, estimated rainfall, future market price prediction etc). It will provide a full year's coverage for the user – another issue with most existing AESs – by providing information from what seeds to sow, to market prices. The system will then maintain a log of the messages with the smallholder to tailor the conversation to provide the best user experience.

Some other features that the system could implement could address the most common barriers to SMS-based AESs. For example, the AI could also gradually ‘learn' the user's literacy level and find the optimal way to convey information, minimising confusion and the chance that they will not choose to implement the suggestion/opt-in notifications. This would also minimise the digital literacy divide and ensure that people are not left behind. Another example is that the system could adapt to the gender and culture of the user by using, for example, local dialect phrases to further improve the acceptance of the system [36].

Methods & Methodology

The project will be using an agile development strategy to allow maximum flexibility in feature scoping and provide frequent milestones to ensure the project moves at a satisfactory pace. GitHub will be used, combined with a trunk branching strategy, to allow for automatic deployments and testing of working code while the product is being developed. The project could also utilise CI/CD to ensure a smooth pipeline from production to deployment.

In terms of implementation, the project will be leveraging multiple APIs such as DSSAT/FruchtFolge for data on crops etc and an algorithm will be made to make the data suitable to smallholder farms. Different APIs will be used to provide chat-bot and NLP functionality. Researching which APIs the system will use exactly is part of the Week 5 targets (see timetable below).

Kotlin will be used as the primary language for this project, as it is a succinct language to write in, with the ability to quickly draft a backend, connect to cloud messaging services, train AI models, and leverage chatbot generation APIs.

Due to the nature of the application, testing with real target users is highly unlikely, however it may be possible to get some non-technical survey participants to try the application and ensure it is clear and easy to use. Nonetheless, emails will be sent to the life sciences department in Warwick, as well as various NGO's/charities, to see whether at least someone who knows the target audience well could give the system a test.

To test the technical aspects, it will be possible to contact local allotment organisations and smallholding organisations within the UK to ask whether anyone is willing to test using the application. A series of closed questions to verify the suggestions the app is making and their usefulness will be formatted. Testing that suggestions given by the chatbot are alright in multiple different countries may be tested by advertising to such groups outside the UK, as well as asking family members in different countries.

Timetable

Week 1	Formalise Project idea with supervisor and do general research on smallholdings and farming techniques.
Week 2	Write Specification and continue initial research.
Week 3	Perform a literature review of current SMS-based AESs.
Week 4	Perform a literature review of chatbots specifically to do with adaptive DSS chatbots for agriculture. Revise specification with more specific goals and send to supervisor. Create JIRA board and break down steps.
Week5	Research how to implement a GAI, find data sources for model, get ethical consent for surveys.
Week 6	Get a DCS server. Train chatbot with generic knowledge base.
Week 7	Develop ML algorithm for tailored predictions.
Week 8	Write Progress Report.
Week 9	Continue developing algorithm.
Week 10	Test prototype, re-evaluate next steps.
December holidays	Holiday, Working Full-time, Seeing Family, Overflow catch up for project.
Term 2 generic overview	Personalise chatbot by using sentiment analysis. Develop sending notifications. Develop survey and test with people. Develop literacy algorithm. Implement multi-lingual support. Final test, prepare presentation. Write final report.

Resources & Risks

The project will require:

- A server, for processing of messages, response generation, and AI training. This will be obtained from DCS
- Connections to a cloud-based SMS platform for communications in production deployments, such as Azure.
- Data sources from which to pull data about locations, as well as for real-time data to provide to the end user
- Multiple brick phones to test different scenarios, as well as a smartphone to test that it works there as well.

Risk	Likelihood (/10)	Severity (/10)	Overall Risk (/100)	Mitigation
Dataset with information needed doesn't exist	7	7	49	Ensure a wide variety of datasets are included, so that if one is not useful, there is minimal impact on the project
Project is not complex or novel enough	5	5	25	Leave enough time at end to add more functionality if required
Ready made trainable GAIs such as openAI go offline permanently	5	10	50	Research backup option to openAI in case it goes offline
Personal Computer corruption	7	9	63	Ensure frequent commits are made to GitHub to back up code
Someone comes up with the same solution while I'm doing this	5	8	40	Leave enough time at end to add more functionality if required
Database corruption	4	10	40	Ensure nightly backups of data are performed to a remote location
Machine learning models do not work as expected	8	8	64	Leave enough time at end to improve machine learning model

Legal, Social, Ethical and Professional Issues & Considerations

As mentioned in the methodology, testing with the target base will not be possible for the scope of this project, however surveys will be carried out on some non-technical and technical users to ensure the efficacy of the product. Consents need to be obtained for the collection and use of this data, and they need to be informed what the data will be used for.

The biggest consideration is, of course, the danger of providing the wrong information to smallholders if this product is released to them. This could risk severely upsetting livelihoods. To mitigate against this, a message will be displayed to all users advertising that the project is an experimental one and that its suggestions should not be used.

Another factor that needs to be considered is getting consent from the users to store their data and use it to train the model better. They also need to be provided with an easy way to delete all the data stored about themselves as required by GDPR.

Bibliography

[1] F. S. a. W. Wu, ""Soil and Crop Management Strategies to Ensure Higher Crop Productivity within Sustainable Environments,"," *Sustainability,* vol. 11, no. 5, p. 1485, Mar 2019.

[2] G. F. Jock R. Anderson, "Chapter 44 Agricultural Extension," *Handbook of Agricultural Economics,* vol. 3, pp. 2343-2378, 2007.

[3] P. G. G. J. T. D. P. Dario H. Galeon, "SMS-Based ICT Tool for Knowledge Sharing in Agriculture," *International Journal on Advanced Science Engineering Information Technology,* 2019.

[4] Precision Development, "precisiondev," PxD, [Online]. Available: https://precisiondev.org/our-learnings/learning/. [Accessed 10 10 2023].

[5] A. Gioia, "accesstolandeu," Eco Ruralis, April 2017. [Online]. Available: https://www.accesstoland.eu/IMG/pdf/comparative_analysis_of_small_farms_in_europe.pdf. [Accessed 2023 11 05].

[6] A. Suprem, N. Mahalik and K. Kim, "A review on application of technology systems, standards and interfaces for agriculture and food sector," *Comput. Stand. Interfaces,* vol. 35, pp. 355-364, 2013.

[7] FAO, "How to feed the world by 2050," in *high level expert forum*, 2009.

[8] D. Beasley, "World Food Programme," EFP, 24 September 2021. [Online]. Available: https://www.wfp.org/news/world-wealth-9-million-people-die-every-year-hunger-wfp-chief-tells-food-system-summit. [Accessed 12 10 2023].

[9] Food and Agriculture Organisation of the United Nations, International Fund for Agricultural Development, United Nations Children's Fund, UN World Food Programme, World Health Organisation, "The State of Food Security and Nutrition in the World," 2022.

[10] United Nations, "United Nations," [Online]. Available: https://www.un.org/sustainabledevelopment/hunger/. [Accessed 2023 10 10].

[11] C. M. B. M. H. J. M. J. H.-M. M. A. D. M. E. H. Enock Siankwilimba, "Bioecosystems towards sustainable agricultural extension delivery : effects of various factors," *Environment, Development and Sustainability,* 2023.

[12] N. W. M. P. W. B. A. M. E. L. Peter M. Kopittke, "Soil and the intensification of agriculture for global food security,," vol. 132, 7 August 2019.

[13] D. C. R.A. Fischer, "Issues for cropping and agricultural science in the next 20 years," *Field Crops Researcg,* vol. 222, pp. 121-142, 1 June 2018.

[14] C. K. I. A. Georgios Kountios, "Can the adoption of ICT and Advisory Services Be Considered as a Tool of Competitive Advantage in Agricultural Holdings? A Literature Review," *Agronomy,* vol. 13, p. 530, 12 February 2023.

[15] S. S. Baba, "Effectiveness of Knowledge Management Approaches on Agricultural Extension Services Delivery in Selected Districts of Northern Region," 2018.

[16] T. D. P. B. R. R. B. Angella Namyenya, "E-diary: a digital tool for strengthening accountability in agricultural extension," *Information Technology for Development,* vol. 28, no. 2, pp. 319-345, 2022.

[17] T. A. B. Mostafa Kamal, "Mobile Applications Empowering Smallholder Farmers: A Review of the Impact on Agricultural Development," *International Journal of Social Analytics,* pp. 36-50.

[18] A. P. H. S. G. D. G. Nicoletta Giulivi, "Getting the message out: Information and communication technologies and agricultural extension," *American Journal of Agricultural Economics,* 2022.

[19] P. G. G. J. T. D. P. Dario H. Galeon, "SMS-Based ICT Tool for Knowledge Sharing in Agriculture," *International Journal on Advanced Science Engineering Information Technology,* vol. 9, no. 1, pp. 342-349, 2019.

[20] E. C. Y. M. A. e. a. Isgren, "Are agricultural extension systems ready to scale up ecological intensification in East Africa? A literature review with particular attention to the Push-Pull Technology (PPT)," *Food Security,* vol. 15, p. 1399–1420, 2023.

[21] J. C. Aker, "Dial "A" for agriculture: a review of information and communicationtechnologies for agricultural extension in developing countries," *Agricultural Economics,* 2011.

[22] N. C. Eli-Chukwu, "Applications of Artificial Intelligence in Agriculture: A Review," *Engineering Technology & Applied Science Research,* vol. 9, no. 4, pp. 4377-4383, 2019.

[23] M. S. Bhandarkar, "A Case Study of Virtual Assistant and Effective Implementation for Assisting the Farmers," *Catalyst Research,* vol. 23, no. 2, pp. 434-447, August 2023.

[24] D. Power, "The Global Development Research Center," [Online]. Available: https://www.gdrc.org/decision/dss-types.html. [Accessed 1 11 2023].

[25] "Talking HealthTech," 15 10 2020. [Online]. Available: https://www.talkinghealthtech.com/glossary/decision-support-system. [Accessed 1 11 2023].

[26] M. Fairlie, "Business.com," 21 Feburary 2023. [Online]. Available: https://www.business.com/articles/decision-support-systems-dss-applications-and-uses/. [Accessed 1 11 2023].

[27] A. R. M. I. G. P. R. D. P. a. E. Ö. M. Deveci, "A Decision Support System for Assessing and Prioritizing Sustainable Urban Transportation in Metaverse," *Transactions on Fuzzy Systems,* vol. 31, no. 2, pp. 475-484, 2023.

[28] F. N. Ford, "Decision Support Systems and Expert Systems: A comparison," *Information & Managment,* vol. 8, no. 1, pp. 21-26, 1985.

[29] M. A. P. S. V. Bijan Fazlollahi, "Adaptive decision support systems," *Decision Support Systems,* vol. 20, no. 4, pp. 297-315, 1997.

[30] A. N. a. S. C. M. Ravi, "A Comparative Review of Expert Systems, Recommender Systems, and Explainable AI," *IEEE 7th International conference for Convergence in Technology (I2CT),* pp. 1-8, 2022.

[31] T.-C. H.-L. H. G.-J. H. a. M.-S. C. Hsu, "Effects of Incorporating an Expert Decision-Making Mechanism into Chatbots on Students' Achievement, Enjoyment, and Anxiety.," *Educational Technology & Society,* vol. 26, no. 1, pp. 218-31, 2023.

[32] R. M. Briones, "The Wage Gap Between Male," in *Outside Looking in : Gendered Perspectives in Work and Education*, Philippine Institute for Development Studies, 2019.

[33] nited Nations, "Sustainable Development Goals," [Online]. Available: https://www.un.org/sustainabledevelopment/hunger/.

[34] J. C. A. Christopher G. Harris, "Designing ICT for Agriculture (ICT4A) Innovations for Smallholder Farmers: The Case of Uganda," *Proceedings of the XIX International Conference on Human Computer Interaction,* pp. 1-9, 2018.

[35] J. F. M. V. B. N. L. M. Zhaoyu Zhai, "Decision support systems for agriculture 4.0: Survey and challenges," *Computers And Electronics in Agriculture,* vol. 170, 2020.

[36] R. G. Tony Dwi Susanto, "User acceptance of SMS-based e-government services: Differences between adopters and non-adopters," *Government Information Quarterly,* 2013.

[38] Precision Development, "precisiondev," PxD, [Online]. Available: https://precisiondev.org/pxds-user-communications-platform-paddy/. [Accessed 10 10 2023].

[39] X. Huang, "Chatbot: Design, Architecture, and Applications," University of Pennsylvania, 2021.