DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION

KAKDE ASHUTOSH CHANDRAKANT

B.Tech. (Agril. Engg.)

MASTER OF TECHNOLOGY

IN

AGRICULTURAL ENGINEERING

(FARM MACHINERY AND POWER ENGINEERING)

DEPARTMENT OF FARM MACHINERY AND POWER ENGINEERING

COLLEGE OF AGRICULTURAL ENGINEERING & TECHNOLOGY,

VASANTRAO NAIK MARATHWADA KRISHI VIDYAPEETH

PARBHANI – 431402 (M.S.) INDIA

2021

DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION

BY

KAKDE ASHUTOSH CHANDRAKANT

B. Tech. (Agril. Engg.)

A thesis submitted to

Vasantrao Naik Marathwada Krishi Vidyapeeth, Parbhani

in partial fulfillment of the requirement for the degree of

MASTER OF TECHNOLOGY

IN

AGRICULTURAL ENGINEERING

(FARM MACHINERY AND POWER ENGINEERING)

DEPARTMENT OF FARM MACHINERY AND POWER ENGINEERING

COLLEGE OF AGRICULTURAL ENGINEERING & TECHNOLOGY,

VASANTRAO NAIK MARATHWADA KRISHI VIDYAPEETH

PARBHANI – 431402 (M.S.) INDIA

2021

CONTENTS

DECLARATION BY THE CANDIDATE

I hereby declare that the thesis entitled, "**DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION**", submitted by me is based on the actual work carried out by me under the guidance and supervision of **Dr. S. N. Solanki**. The extent of information derived from the existing literature have been duly cited and referenced. The existing research work or its any part is not submitted anywhere else for the award of any degree or diploma.

I also hereby declare that no sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is cited and duly referenced. There is no plagiarism; the work presented is original and own work of the researcher. No ideas, process, results or words of other have been presented as researcher's own work.

Place : Parbhani

Date : 16 / 02 /2022

(Kakde Ashutosh Chandrakant)

Reg. No. 2019AE/02M

Dr. S. N. Solanki

Head,

Department of Farm Machinery and Power Engineering,

College of Agricultural Engineering and Technology,

Vasantrao Naik Marathwada Krishi Vidyapeeth,

Parbhani-431402 (M.S.) India.

CERTIFICATE - I

This is to certify that the thesis / dissertation entitled, "**DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION**" submitted by **KAKDE ASHUTOSH CHANDRAKANT, Reg. No. 2019AE/02M** in partial fulfillment of the requirements for the award of the degree of **MASTER OF TECHNOLOGY (Agricultural Engineering)** in the subject of **FARM MACHINERY AND POWER ENGINEERING** submitted to the Vasantrao Naik Marathwada Krishi Vidyapeeth, Parbhani is a record of bonafide research work carried out by his under my guidance and supervision. The thesis or its any part has not previously formed the basis for the award of any degree, diploma or other similar title.

Place: Parbhani

Date: 16 / 02 / 2022

(S. N. Solanki)

Research Guide

CERTIFICATE - II

This is to certify that the thesis entitled, **"DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION"** submitted by **KAKDE ASHUTOSH CHANDRAKANT, Reg. No. 2019AE/02M** in partial fulfillment of the requirements for the award of the degree of **MASTER OF TECHNOLOGY (Agricultural Engineering)** in the subject of **FARM MACHINERY AND POWER ENGINEERING** to the Vasantrao Naik Marathwada Krishi Vidyapeeth, Parbhani has been approved by the Student's Advisory Committee after viva voce examination of the student in collaboration with the External Examiner.

External Examiner
Dr. P. A. Munde

(S. N. Solanki)
Research Guide & Chairman

Advisory Committee Members:

(G. U. Shinde)
Member

(R. T. Ramteke)
Member

(R. V. Chavan)
Member

(S. S. Phulari)
Member

(V. M. Khodke)
Associate Dean & Principal
College of Agril. Engg. & Techology,
VNMKV, Parbhani

Head
Dept. of Farm Machinery & Power Engineering
CAET, VNMKV, Parbhani

PLAGIARISM CLEARANCE CERTIFICATE

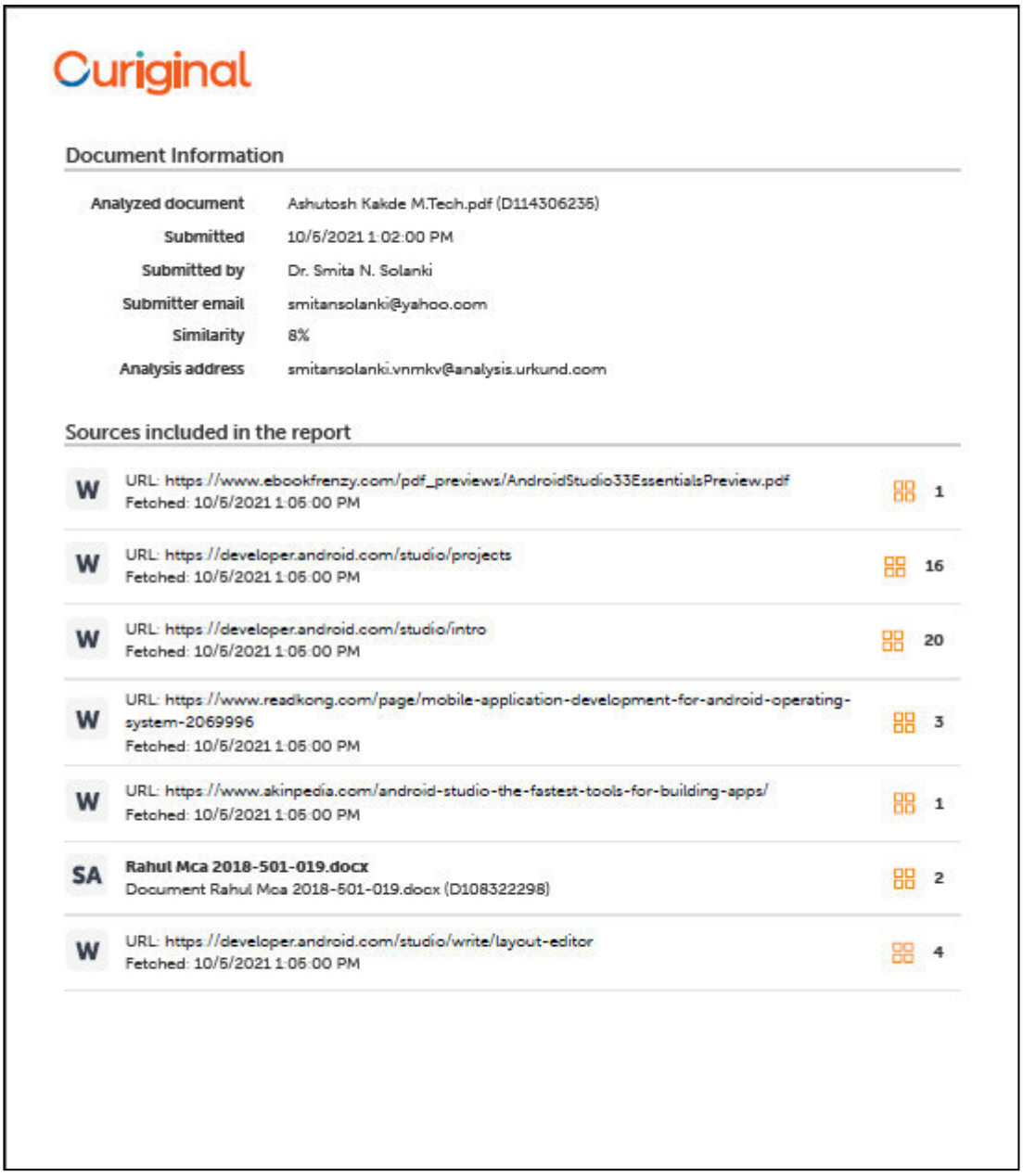
This is to certify that thesis entitled, **"DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION" submitted by KAKDE ASHUTOSH CHANDRAKANT, Reg. No. 2019AE/02M** has been properly examined by URKUND: Anti plagiarism Software. The percentage of similarities found in the thesis is 8%.

No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is duly cited and referenced. The work presented is original and own work of the researcher (i.e. there is no plagiarism). No ideas, process, result or words of other have been presented as researcher's own work.

(S. N. Solanki)

Research Guide

URKUND REPORT



Curigional

Document Information

Analyzed document	Ashutosh Kakde M.Tech.pdf (D114306236)
Submitted	10/6/2021 1:02:00 PM
Submitted by	Dr. Smita N. Solanki
Submitter email	smitansolanki@yahoo.com
Similarity	8%
Analysis address	smitansolanki.vnmkv@analysis.urkund.com

Sources included in the report

W	URL: https://www.ebookfrenzy.com/pdf_previews/AndroidStudio33EssentialsPreview.pdf Fetched: 10/6/2021 1:05:00 PM	1
W	URL: https://developer.android.com/studio/projects Fetched: 10/6/2021 1:05:00 PM	16
W	URL: https://developer.android.com/studio/intro Fetched: 10/6/2021 1:05:00 PM	20
W	URL: https://www.readkong.com/page/mobile-application-development-for-android-operating-system-2069996 Fetched: 10/6/2021 1:05:00 PM	3
W	URL: https://www.akinpedia.com/android-studio-the-fastest-tools-for-building-apps/ Fetched: 10/6/2021 1:05:00 PM	1
SA	**Rahul Mca 2018-501-019.docx** Document Rahul Mca 2018-501-019.docx (D108322298)	2
W	URL: https://developer.android.com/studio/write/layout-editor Fetched: 10/6/2021 1:05:00 PM	4

ACKNOWKEDGEMENT

I think it is a matter of pleasure to glance back and recall the path one travels the days of hard work and perseverance. It is still great at the juncture to recall all the face and spirit in the form of teachers, family, friends, near and dear ones.

I avail this opportunity to acknowledge my sincere, humble, indebtedness and whole-hearted sense of gratitude to my honorable guide Dr. S. N. Solanki, Head, Dept. of Farm Machinery and Power Engineering, College of Agricultural Engineering and Technology, VNMKV, Parbhani who conceived, detailed and shaped the research problem and provided adequate guidance which led to the successful articulation of this thesis. His valuable suggestions, sympathetic behavior and co-operative nature during the course of present investigation would remain encouraging me forever in my life.

I owe high esteemed respect to Dr. U. M. Khodke, Associate Dean and Principal, College of Agricultural Engineering and Technology, VNMKV, Parbhani for providing necessary facilities during the present investigation.

I feel immense pleasure in expressing my humble and deepest sense of gratitude to Dr. R. T. Ramteke, Head, Department of Electrical and Other Energy Sources, College of Agricultural Engineering and Technology, VNMKV, Parbhani and Er. A. A. Waghmare, SRA, AICRP on Utilization of Animal Energy, VNMKV, Parbhani for their expert opinion and valuable suggestions during the project work and providing necessary facilities for this study.

I would like to express my sincere thanks to advisory committee members, Dr. G. U. Shinde, Assistant Professor, Department of Farm Machinery and Power Engineering , College of Agricultural Engineering and Technology, VNMKV, Parbhani, Dr. R. V. Chavan, Associate Professor, Department of Agricultural Economics, College of Agriculture, VNMKV, Parbhani and Dr. S. S. Phulari, Assistant Professor, Department of Computer Science for their useful suggestions, valuable advises and constructive criticism for this manuscript.

I am enormously grateful to Dr. P. A. Munde, Associate Professor, Department of Farm Machinery and Power Engineering, College of Agricultural Engineering and Technology, VNMKV, Parbhani , Dr. P. S. Kapse, Assistant Professor, Department of Extension Education, College of Agriculture, VNMKV, Parbhani, and Prof. D. V. Patil, Assistant Professor Department of Farm Machinery and Power Engineering, CAET, VNMKV, Parbhani for his rendering help and co-operation at various stages of this work.

I would also like to thank the expert who were involved in development of app for this research project Mr. Vishal S. Wadmare. Without their passionate participation and input, the development could not have been successfully conducted.

I avail this opportunity to acknowledge all those who helped & guide me during the course of my research work. This journey would not have been possible without the support of my family. No words are enough to express my love for my father Mr. Chandrakant, mother Mrs. Vidhya, and my uncle Mr. Shirish Chaudhari for their deep love, encouragement and care throughout my life.

I wish to extend my gratitude towards my friend Mr. Sanket Kokane, Krishna Lohakare, Lily, Pragati Dhande, Shalaka Kalmnurikar and Pornima Rathod for providing necessary help and guidance as and when required for the completion of this work. Whenever I approach him, they were always ready to spare their valuable time for the technical discussion. Thank you so much dear. My whole hearted thanks to my all friends, Balaji Jawale, Bhagvan Sangle, Ramkisan Jadhav, Shriram Murtadak, Vijay Mane, Shriniwas Varpe, Chalapati H. V. for their help and support during the entire degree course.

Most importantly, I am grateful for my brothers; Big Bro, Sai, Shanu, Chetu, Omiee, Yash, Khasdar saheb for their unconditional, unequivocal and loving support.

I am grateful to all the authorities of university, all teaching and non- teaching staff and those lovely peoples who are part of my life but remain to mention here for their blessing, co-operation and affection from them.

I ever thank one and all who helped directly or indirectly, visibly and invisibly inspired me the most.

PLACE : Parbhani

DATE : 16 / 02 /2022

(Kakde Ashutosh Chandrakant)

LIST OF TABLES

LIST OF FIGURES

LIST OF SCREENSHOTS

ABBREVIATIONS USED

Abbreviation		Elaboration
%	:	Per cent
/	:	Per
ICT	:	Information and Communication Technology
IT	:	Information Technology
GIS	:	Geographical Information System
CV	:	Coefficient of variation
UI	:	User Interface
IP	:	Image Processing
MCC	:	Mobile Cloud Computing
Engg	:	Engineering
et al	:	All other
KCC	:	Kisan Call Centre
AICRP	:	All India Co-ordinate Research Project
Etc	:	Excetra
SDK	:	Software Development Kit
OS	:	Operating System
iOS	:	iPhone Operating System
Fig.	:	Figure
ART	:	Android Runtime
CPU	:	Central Processing Unit
XML	:	Extensible Markup Language
IDE	:	Integrated Development Environment
i.e.	:	That is
APKs	:	Android Application Packages
AVD	:	Android Virtual Devices
Apps	:	Applications
IDE	:	Integrated Development Environment
JDK	:	Java Development Kit
GUI	:	Graphics User Interface
USB	:	Universal Serial Bus
API	:	Application Programming Interface

JAR	:	Java Archives
HTTP	:	Hypertext Markup Language
VCS	:	Visual Computing System
TV	:	Television
NDK	:	Native Development Kit
OOP	:	Object Oriented Programming
URLs	:	Uniform Resource Locators
RAM	:	Random Access Memory
XP	:	eXPerience
JRE	:	JAVA Runtime Environment
V.N.M.KV.	:	Vasantrao Naik Marathwada Krishi Vidyapeeth

ABSTRACT

THESIS ABSTRACT

1. Title of the thesis	**:**	**DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION**
2. Full name of the candidate	**:**	**Kakde Ashutosh Chandrakant**
3. Full name of the Research Guide	**:**	**Dr. S. N. Solanki**
4. Department	:	Farm Machinery and Power Engineering
5. College/ University	:	Vasantrao Naik Marathwada Krishi Vidyapeeth, Parbhani
6. Degree to be awarded	:	Master of Technology (Agricultural Engineering)

ABSTRACT

Agriculture is one of the significant contributors of Indian economy and play lead role in it. Whereas advancement in agriculture will return larger benefits to the economy. Widespread availability of smartphone and Internet, there is a huge potential for supplying essential information by the means of smartphone and Internet. Android holds share of 82.18% in smartphone industry of India. Sharma Sunidhi et. al., (2008). Android offer several features like having easy UI, affordable and readily available.

The "FARM MECHANIZATION" app can be installed with the apk or share on network. It is developed and tested using emulators which validate its function and reliability. The user interface was developed using JAVA language. The App "FARM MECHANIZATION" was developed on Android Studio and minimum requirement of API to run the app is 5.1 Lollipop version. Tools used to develop the 'Farm Mechanization' app are Android Studio IDE, SQLite, IntelliJ IDEA and Android Device Monitor. From this android app farmer can easily access information of machines for different mode of operations for the cultivation practices of eight major crops.

Keywords: Farm Mechanization, Machines, Mobile app, Android

CHAPTER -I

INTRODUCTION

CHAPTER-I

INTRODUCTION

India is marketed as a ‘Krishipradhan' country. Maharashtra is quite important in terms of agricultural productivity. Mechanization is a crucial component of agricultural progress and worker well-being. Mechanization is the use of machines, automatic devices, new tools, or technologies to better a process or activity. Production, preservation, transportation, manufacturing, packing, distribution, and reselling are all processes in a well-structured agriculture. The term "mechanization" refers to the use and application of tools, machinery and tools to increase the productivity of agriculture labour and land; it may employ human, animal, or motorized power or combination of these. Agriculture Mechanization Systems will aid in increased production and farmers revenue. Reduces drudgery of operation, increase efficiency of operator without jeopardizing health. Helps in timely operation, saving in input, time and reduces cost of operation and increases yield. Thus from all direction helps in doubling income of farmers. With the digitalization of the economy, innovation may play a significant role in boosting mobility. Even our farmers in the countryside have access to digital tools. Technology will also be beneficial to both small and large farmers and resulting in increased output.

The great majority of the Indian growers, including local producers are routinely denied access to information and technical resources that could boost output and result in higher pricing for their crops and goods. We are the world's leading producer of rice, wheat, pulses, and spices, as well as the world's second most successful grower of fruits and vegetables. Access to the information which lead to higher yields and higher prices of the products. It also aim to close the gap between delivery and availability of agricultural supplies and infrastructure. Farmers today obtain a lot of information and facts about farming, including cropping pattern, crop selection, crop practices, climate, organic fertilizer, pesticides, industrial equipment, and so on, from such a variety of sources that are divided into a different location based on their supplier, manufacturers, producers, or importers. Sowing seeds to harvesting of the crops, farmers need to expend lot of blood and sweat. As a result, it is necessary to create a platform in which the necessary information is readily available and accessible to the farmer.

Farm machinery contributes to increased yield, productivity, and revenue growth in agriculture by attaining timeless in farming activities, helps to bringing precision in placement of inputs, lowering available input economic loss, boosting utilization efficiency of expensive inputs, lowering unit cost of harvest, maximizing profitability and efficiency through cost of operations. It also aids in the preservation of food and byproducts based on qualitative and quantitative degradation, as well as enables value addition for extra revenue. The low cost, simple mobility, and user-pleasant appearance of cell phones has everyday been the primary benefit of making them popular among a few of the common human.

Inventions in agriculture technology are not reaching to the farmers, because many of them are uneducated so they are uninformed of location where they might obtain information. Hence, the farmer is at utmost level of being lack of knowledge about the modern techniques. Therefore use of smartphone to overcome this makes comfortable to farmers. Smart phones are not only most effectively used mostly for communication via the telecommunications network, but they also come with a number of handy tools simply referred to as "apps" that may be broadly used by users in their day-to-day lives. There are numerous mobile apps being developed internationally that provide an excellent interface for users for outstanding functions such as e-ticketing reservation, virtual trading agricultural farm products, and mobile professional advisory services relating to agriculture and available at their location.

1.1. Smartphones

Information and communications technology have played a significant influence in farmer's everyday life. ICT (Information and Communication Technology) in agriculture is now a new field that focuses on agricultural growth and rural agriculture development. There is a significant gap between both the management of availability and delivery of agricultural inputs and agricultural infrastructure, which technology tools and emerging apps can help to overcome.

1.2 Information and Communication Technology (ICT)

Cloud technology, integrated IT services platform, online learning and also the growth of smartphones make it easier to disseminate agricultural-related information and data to the farmers. It allows farmers to make efficient land strategic decisions. It

permits the monitoring of soil conditions in combination with weather data in order to better planning of the planting and harvesting seasons. Consequently, Geographical Information Systems (GIS) could also be used for precise information about pests and livestock diseases so that growers can respond appropriately to the amount of risk. Smartphone and cloud digital computing technology can also be used to optimize the usage of fertilizer, seeds and irrigation water.

In agricultural production and development, innovations in technologies is becoming crucial increasingly. Mobile ICT (Information and Communication Technology) in agricultural production gives a more productive and cost-effective technique of extensively working together for exchanging and sharing knowledge. Farmers gain from having access to critical information like as diseases and pests reports, weather forecasts and market pricing. It can also help to increase cooperation between farmers, researchers and extension workers, who may not be able to visit farmers as frequently as both sides parties would want. Effective communications between farmers, extension officers, academics and policymakers is critical to the advancement of agricultural e-commerce.

When planning to the adoption of digital technology, many more difficulties to be considered. The context of the innovation and technology and the end user's requirements are may be the most vital to consideration. There is also no single software or device which will work in every situation in every country. Some areas might easily adopt tablets with smart apps and while other countries might prefer to receive news and information via SMS. Technology does not seem to be difficult.

ICT skills of people in the region must also be recognized. Offering technology will not ensure a long-term project. Investing in user training and developing technological support is critical to making the most use of ICT and assuring the projects long-term success and profitable. Monitoring and assessment evaluation (M&E) groups are also required to analyze the projects continually. Communication among evaluators which enables projects to learn out from experiences and accomplishments of others, allowing the best practices in the field to be utilized. These could be disseminated by media such as television, radio or pamphlets.

1.3 Android Architecture

Android is an free and open source application development framework that allows developers to create extremely sophisticated applications. The Google's android system is a collection of software components and development tools grouped into five categories: Apps, Programming Model, Utilities, Android Runtime, Linux Operating systems, and four core layers. It allows designers to take full advantage of both the devices technology, receive location data and execute hidden services, among other things. Once the app has been published, it can be downloaded from online stores such as Google's app store, Android Market. Specialized comprehensive development platforms, such as Android Studio or Eclipse, are necessary for mobile application development. The app is initially evaluated using emulators, which seem to be software simulations of genuine hardware components, and afterwards field testing and assessment is undertaken. Another important aspect of developing applications is mobile user interface (UI) designing. Context, display, and client input and output control signals mobility are all factors to consider while designing a user interface. Mobile user interfaces are considered front-end, and they depend on database servers to provide access to business services.

Mobile applications are also gaining popularity in the agricultural industry, with the prospect for the further improvement. Many efforts have been made to build mobile apps in the rural development and sustainable agriculture sectors in order to enhance rural livelihoods and farming activities. Singhal et. al., (2011) developed android based solution for Indian agriculture named Krishi Ville. Saha et. al., (2012) developed m-Sahayak. Farm Manager for the management of small farm was developed by Lantzos et. al., in 2013. Hetal Patel et. al., Survey of Android Apps for Agriculture Sector in 2016. Abishek A.G. et. al. studied on Agriculture Marketing Using Web and Mobile Based Technologies in 2016. The scientists are working on a smartphone application for farming techniques, agricultural production, and weather forecasting. As a result, using a smartphone app to share the knowledge, information, ideas, and provide services to farmers may be done more efficiently.

This work is focused on the development of the android based mobile app. The app will provide detailed information about the package of operation over different crops with respect to farm machinery, tools and power sources. The

operational mechanism of the app also helps to fill the mechanization gap by accepting modern techniques. Incorporating innovative technology targeted at decreasing pre- and post-harvest losses by appropriate procedures, as well as encouraging value addition.

The major goal of this project is to create a smartphone based solution that aids in farm management, results in increased agricultural yields, and aids in agronomic practices. It can investigate the influence of smartphones on the Indian production sector, with a particular focusing on local and small scale farmers. To satisfy this goal, agricultural crop yield must be doubled from its current level. As we all know, IT is constantly spreading its claws and enhancing efficiency in all fields, and from what we can tell, only farmers and producers appear to be unaffected by it, which is likely the cause of our farmers' backwardness and it will in marginalization. This Android application software is only for farmers who use smart mobile phones and want to get authentic information on the overall performance of all crops.

To design a full-featured Mobile app for farmer's perspective, simple to use and adaptable, with storage involvement, it enables farmers to clarify their concerns and obtain relevant information for any professional to propose. It includes:

1. Multi-functional software to assist farmers in obtaining detailed agricultural information of crops.
2. Mobile app is a multi-lingual services.
3. All modes of learning, including multimedia content, for farmers as well as the unskilled.
4. It can also serve as a question-answering service in offline platform.

Objectives

- Collection and formulation of package of operation for different crops.
- To design and develop android based mobile app for farm mechanization of Maharashtra.
- To test the developed mobile app.

CHAPTER -II

REVIEW OF LITERATURE

CHAPTER- II

REVIEW OF LITERATURE

This chapter deals with a review in relevance of the objectives by the proposed study of research conducted. Developed some android based mobile application for weather forecasting, market selling price, NPK detection, diseases identification etc. This chapter discusses information from published work on android based mobile application

Development of android based mobile application is based on different developing platforms. The following contains a review of the study as well as information about it.

2.1 General

2.2 Application development

2.3 Survey of App

2.4 Testing of App

2.1 General

Oinas-Kukkonen and Kurkela (2003) studied challenges concerning the development of effective mobile applications, both commercially and in terms of user self satisfaction. It illustrates that, despite the numerous constraints of mobile devices, it is important to take into consideration developing them. Mobile technology could become a major driving force in the development of android based mobile applications. Despite the various limitations of smart phones, it is worthwhile to consider for design and developing such applications. Furthermore, the creation of mobile applications should not be about minimizing features and information, but rather about developing new, inventive ways of utilising information technology in location, time, access confidential and user sensitive data. The mobile design concepts of mobility, utility, significance, ease of use, navigation proficiency, and personalisation may assist us in this approach.

Islam Md. Rashedul, and Islam Md. Rofiqul (2010) reported Mobile applications are one of the most important and fast increasing fields in the digital information and communication era. We exhibit the outcome of a smartphone app in the commercial sector. Various statistical information on the history and present status of smartphone apps in the past 60 years ago was 10%, in 2009 18%. Whereas smartphone application fields of utilization are rapidly expanding in Chat, Entertainment, Multimedia, Productivity, Tourism, and Utilities. Mostly used in conversation and entertainment.

Wu Yi and Klarissa Chang (2013) empirical studied and investigated how the simplicity of design effects users' intellectual and affective perceptions of mobile application behaviour intention. This investigates simplicity design issues of mobile applications interaction perspective, two design artifacts. The use of symbol represent in information design with mobile application interactions promotes both objective and subjective evaluations of mobile application behaviours. The usage of location-awareness, on the other hand, only greatly increases the formation of emotive assessment. This could be the location-awareness function clear and simplifies the interface for user's actual processing; it filters out unnecessary but frequently important information for users to interpret.

Madhumathi R, and RadhaKrishnan R. (2016) studied on Auction Application of Amazon Web Services for Agro Product Sales from Growers and Agriculturalists to attain the best value for their items they manufacture or sell. The owner/farmer obtains their best price for their produce. The bid begins with the low bidder, and the amount continues to rise as other competitors increase their bids. Whenever the highest bid-price is met, then there is no additional high bid offer from the buyers, and the bidding is over. The seller establishes an appropriate price called the marginal value, and the bidding is held. As a result of the auction being forcefully ended when the bid falls below the optimum price, this approach offers the seller with a guaranteed price. This web-page not only delivers the greatest compensation for farmers, but it also has many other characteristics that make the application the most simple, dependable, and user friendly.

Mane, and Kulkarni (2019) reviewed to assist the farming community, a smartphone app for agricultural advancement has been developed. Various

agricultural technologies have been created, and the government also provides additional services to farmers in order to increase output. Due to inequitable management, all critical data, plans, and schemes pertaining to farming do not reach farmers on time. To alleviate this challenge, the government must help to enhance agriculture growth, as well as farmers' ignorance of new technologies. This app merely defines the required technique and model for informing farmers regarding new and diversified agricultural information and assisting them in improving agriculture in our country.

2.2 Application Development

Shweta Sharan, *et. al.* (2013) studied that aim of this project was to create a mobile phone-based solution that aids in farm management, leads to increased agricultural yield, and aids in farm care and maintenance. It would also assist them in maintaining a list of suppliers and sellers of various commodities and items sold in the market. The software would also give some information regarding various credit schemes provided by India's major banks. In this app, the credit strategies vary on a regular basis, the newer versions would have the most recent schemes. Receiving market informational updates for various items allows them to make better selections about where to sell the production after gaining market pricing for a range of regional and distant marketplaces.

Prasad, *et. al.* (2013) carried out a research on Information Technology must play a major role in Indian agriculture in order to assist farmers in enhancing production all-over by a Cloud-Based Framework for Farming community on Smartphone. Innovation could be used in two different ways: as a direct resource in agriculture production, like satellite technologies, geographical information system (GIS), agronomy, soil sciences, and also as an indirect resource for uplifting farmers to take information and also have conversations that favorably enhance typical farming practices. Farmers can utilize multi-functional software to find extensive information on crops, infections, and causes. Whether prediction were monitor the climatic conditions without providing any location or position since it will be automatically traces the location by using GPS. Crop Advice and Analysis, to capture the crop pictures, sends them to the server, to undergo through the process. Image Processing (IP) methods and some knowledge-based classifiers operating also on the

server, to identify periodical seasonal diseases. Mobile Cloud Computing (MCC) helps identification of botanical species and detection of diseases using a simple smartphone with a camera–a Remote Vision.

Confalonieri, M. Foi, *et. al.* (2013) designed and presented an app for smartphones that implements two approaches for LAI estimate based on the utilization of sensors and processing speed (power) that are commonly found in most latest smartphones. The initial technique (App-L) estimates the gap proportion at 57.5 (to obtain values that are nearly independent of leaf tilt) from luminance determination above and below the canopy. The second technique (App-G) determines the gap proportion by automatically processing of photos taken under the canopy. The accuracy and linearity are the two approaches utilized in the application were determined using information recorded in a scatter-seeded rice field in northern Italy and performance is compared to those with the LAI-2000 and Accu-PAR ceptometer. App-G performed similarly to LAI-2000 and AccuPAR, while App-L attained the best trueness rating (RMSE = 0.37 m^2) whereas it reseulted the least accurate, necessitating a large number of replication to generate valid estimations.

Uchinuno *et. al.* (2013) developed the knowledge sharing platform for agricultural applications in data overview. This research material concerning agricultural product growers' judgment was gathered from a webpage. Similarly, the sensor put in fields collects cultivation information about the environment. Transmission of agricultural technology through the use of two data collection methods: automatic collection of environmental information by a detector (sensors) and recording of data available by a farmer. Data Sharing Network outlines the concept for skilled farmers' information and reports the experimental results of environmental-data collecting regarding cultivation utilizing specific garden planters. Knowledge sharing platform will not provide robustness of actual data.

Intaravanne, annd Sarun Sumriddetchkajorn (2013) demonstrated a smartphone based rice leaf colorimeter for nitrogen detection in the paddy field. This smartphone contains our Android-based application programme named "BaiKhao," which analyses the colour tone of the rice leaf and estimates the required quantity of Fertilizer N. The rice leaf and white reference material were arranged in such a way that they are both in the field of vision of the smartphone camera. This indicates that

both the reference and the object were examined at the same time. In our field deployable prototype, we evaluate the CV value and link it with the leaf colour level generated from the LCC.

Murakami *et. al.* (2013) proposed and developed a cloud-based cultivation management system. The technology assists many farmhouses in managing agricultural activities in order to achieve cost-effective precision agriculture. Precision agriculture aims to optimise farmland management and necessitates to record agricultural work. Farmers traditionally keep paper records, but it is difficult and time-consuming to examine previous agricultural-work data and regulate the cost of agricultural production. iFarm app is offered for cultivation management system built to enable effective farming management. The system is composed of smartphone apps, Web applications, as well as a cloud server.

Delgado *et. al.* (2015) presented a mobile application that enables farmers and agricultural professionals to address several agricultural jobs using their smartphones. It only takes a few taps in this application to determine the earth–sun relative distance, extraterrestrial solar radiation, incidence radiation on Earth, number of hours of sun, equivalent evaporation, and so on. These parameters were required for various tasks related to water management in agriculture, such as estimating crop water requirement, validating data supplied by agricultural meteorological stations, measuring solar radiation in a location of interest for installation of solar water heating systems, and so on. It has been rebuilt utilising modern trends of mobile application; we have involved new way of acquiring coordinate inputs through the projection of maps using the Google Maps tool, by choosing on the map, including the height given in this coordinate by online service. Unlike the previous version, this programme permitting the user to send the collected data not only via text message, SMS type, but also via other soft-ware apps such as email or social networks.

Aniket Bhave *et. al.* (2014) studied on Mahafarm – An Android-based approach for Remunerative Agriculture Information and Communication Technology (ICT) in agriculture is a new field that focuses on improving agricultural and rural development in India. In the rural area, using innovation and technology is a pivotal factor. The improvement of ICT may be used to provide farmers with accurate and timely relevant information and services, hence facilitating an environment to

remunerative agriculture. This study describes a smartphone based app for farmers that will comprehensively assist them in their farming tasks.

Patel and Thakkar (2014) conducted research on Agricultural Android Application to give entire idea about by using this Proposed System, just by operating the android phone farmers will receive all the information provided by the kiosks. As, Whether Forecasting information, Agricultural crops information and he will get various crops diseases information also farmer want to talk with Agricultural Experts for instant problem solution may added this feature. This proposed system provides user friendly environment to the illiterate farmer of India as System provide information in local language and in voice form. He received all Whether Forecasting, Agricultural crops, Crops diseases information just operation Android Phone Keypad. Farmers can access information 24X7 from anywhere, thus travelling time and waiting time in queue to access kiosk will be save. System is very cost effective as it is freely available.

Karkhile and Ghuge (2015) investigated research to provide Innovative Farming Approach Using an Android Application to Improve Transparency in the Agriculture Commodities marketplace by Procuring Market Price Information, Promoting Collective Purchasing of Inputs, and Collective Selling of Output. Researcher adopted Methodology to provide Techniques for Farmers as Weather Forecast Report, Information about Crops, News and Feeds, Farming Tools and Technology. Developed a smartphone based system that aids in farm management, leads to increased agricultural productivity, and aids in farm perpetuation. Traditional farming needs a significant quantity of workers and a variety of activities. Alternatively, modern farming does not need a large quantity of workers because mobile, equipment, and new technology take care of everything. This mobile application gives real-time weather, news, and market prices in a variety of positioning, with all data offered in local languages.

Sharma *et. al.* (2015) conducted researched on an When a farmer is unclear about which fertilizer to use, which insecticide to use, or when to perform specific farming tasks, he or she should consult with a professional. Farmers will receive reminders about fertilizer, herbicide application, pesticide for diseases, and weather updates if a crop's favorable temperature range is exceeded based on the crop's date of

sowing. eAgro will assist farmers with experts in crop cultivation, pricing, fertilizers, and disease detail technique of cure, as well as ideas on innovative ways for cultivation, utilization bio-fertilizers, and crop cultivation. This will not only display present weather conditions but will also provide weather alerts to prevent future crop damage and it will recommend the best crop for them, along with the necessary fertilizers, pesticides, and herbicides throughout their operations.

Chirmade *et. al.* (2015) investigated research on Agro Supply Chain would be accessible on smartphones, developed for farmers to help them remain on track, avoid problems, organize their cultivation spending, and getting updated with the latest information, government schemes, and strategies relevant to the field of agriculture, as well as sugarcane supplier information. The advisory system will provide customers with real-time and interactive advices on crop guidance and alerts. Plantation, insect, disease, and nutrition notifications will also be delivered. Farmers would also regularly receive pest and disease alerts, as well as market price (MSP) information, to help them making on-farm decisions. It assists the farmer in order to manage his spending and schedule. This application provides farmers with an estimated budget for cultivating sugarcane. Farmer enters information such as land area, soil type, and month of planting. The amount of fertilizer, the amount of water is displayed based on the input harvesting period. Farmers will be able to receive the most recent government policy pertaining to agriculture. It also includes innovative farming strategies and technologies, as well as plantation and upcoming crop producing technologies for better production of crop. The plantation technique allows for the management of quantity of irrigation to be done, as well as the type and amount of fertilizers to be utilized in the plantation. This Android application is a complete bundle for farmers who want to cultivate sugarcane and get good yields through effective management.

Choudhary *et. al.* (2015) developed the system uses an Ardino Uno Microcontroller that is linked to GSM, sensors, and a motor to schedule and manages agricultural gadgets on the Android platform. The temperature sensor was utilized to detects the temperature of the environment, while the capacitive sensor detects the flow of water inside the pipe. This application features a water pump motor on/off switch, a pesticide controller, and the ability for the operator to schedule operations. In this system, the internet is used to link users to the server machine. This server

contains a java programme that sends commands to the microcontroller and transmits farm video to the client. A farmer will be able to manage the motor and pesticide proportions as well as remotely monitor farming practices in the farm due to the simplicity of an Android application. The initiative will enable the irrigation process to be more efficient.

Sopegno *et. al.* (2016) developed a web application for equipments and machinery cost on the basis of information and communications technology (ICT), and now it plays an vital role in corporate processes and provides a strong foundation to manage several daily challenges. Consumers want to be able to access valuable information in real time. To that objective, the goal of this study was to create an easy-to-use smartphone application named "AMACA" (Agricultural Machine App Cost Analysis) for calculating the cost of equipment in various farm operations and making it accessible via a web - based mobile application utilizing a cross-platform approach. Subsequent estimation can be performed by altering the input variables (fuel price, interest rate, field capacity, tractor power, etc.) and comparing the progress on a sensitivity analysis basis. The AMACA app can help you decide whether to buy new equipment/tractors, use your own equipment, or lease on service, as well as choose the least cost-effective farming system.

Srija, and Geetha Chanda (2016) developed AgroNutri App for crop specific fertiliser application to planted area. The application of the necessary quantity of nutrients at the appropriate time and percentage would help to increase farm productivity while also protecting soil fertility and microbial biodiversity. AgroNutri is an Android application that aids in the delivery of crop-specific fertilizer quantities. The objective is to measure the quantity of NPK fertilizer to be given which is based on the blanket recommendation crop of interest. This application based on the crop chosen by the farmer and used as input, giving suggestions to the farmers of NPK Fertilizer as well as the quantity of Urea, SSP, and MOP to be applied.

Gaikwad and Gawade (2017) they proposed an android app to overcome the problem which addresses the issues of famers, benefits them with additional guidance regarding soil type, fertilizers, pesticide, irrigation requirements etc. The proposed system is "Secure light-weight Android application for Farmer based on Rating and reviews" (SAFR) provides security to the payment as well as data encryption (AES)

using different security based algorithm. Modern Farming Technique utilizes SAFR to increase integrity in the agricultural commodities market place by giving market pricing information, promoting collective purchasing of inputs, and selling of goods. This also provide solution and suggestion to the farmers problem and additional guidance in area of interest.

Wedpathak (2017) conducted research on due to bacterial infections and a lack of data resources, a substantial amount of crop was being damaged in the farm. This would effectively assist farmers in selling their produce on the worldwide market and earning a significant profit. Using the Horticulture and Android concepts, it built a "Farmer Helping Service" software for the farmers with detailed information on fruits, vegetables, and flowers in audio format. This technology can give data using an Android mobile-phone from everywhere and at any moment, without the need of an internet connection, and at no expense. Gujarat farmers will benefit greatly from it because they'll be able to gain information in Gujarati simply by putting a number from their mobile keypad.

P. Tamil Selvi and Balasubramaniam (2019) carried out research on the use of a mobile application also keeps them informed regarding weather forecasts for agricultural input applications such as fertilizer and pesticides that may be affected by unforeseen natural disasters as conveyed by blocks namely Thogaimalai and Kulithalai were selected randomly in Karur District. The researcher found that farmers got more awareness about mobile apps. Though the mobile app they were getting agricultural and allied information on time. Many applications are being used for various types of functionality related to farming operations such as cropping data, pesticides, fertiliser, seed, crop selling, irrigation data, crop yield prediction, weather updates, and information about effective farming techniques. According to the findings of this study, most of the listed benefits should be bundled into a single app that is available in the farmer's native language and is simple to use.

Kavitha and Girish (2019) researched that the Utilization Barrier of Tremendously worsen the dark silicon, that has been addressed by applying GREENDROID innovation through the application of C-cores to android, where C-core lower energy usage for important regions through selected de pipelining. Conserving corps will allow for the transformation of dark silicon into energy savings

and increased parallel processing under energy - constrained. The amount of C-Cores used per application can be limited, and the selected depipelining approach can be used to decrease the strain of executing relatively unusual code by decreasing registers and clock transitions. According to future plans, the number of C-Cores can be decreased while performance improves. I Cache and D Cache can be integrated to make the work cycle more faster.

Patil (2019) evaluated and researchers made classification of the application in the form of Weather forecasting explored the information of the weather for upcoming 2-3 days of a selected location, humidity of the selected district on the present day, sunrise, sunset, pressure. Market Rates displays the whole list of Vegetables and Fruits prices accessible on the market. Government Schemes gives in-depth information as well as procedures for various programs. Our method will anticipate future pricing based on market situations for the ongoing week and the prior one or two weeks.Farmers who use mobile phones to acquire real-time updates on vegetable and fruit prices in each and every market in India would be able to sell their produces at the correct prices. Our app has a function in which all farming-related alerts from the government are included, and farmers receive detailed information about various policies. The prices of vegetables and fruits are predicted based on market conditions.

Chaurasia *et. al.* (2020) designed and developed of crop advisory application for seed spices on android based platform. Mobile app SSADV developed for seed spices crop advisory to farmer. It has more features as like cultivation practices, crop economics and value addition. India has wide variety of spices but unable to guide a crop advisor for spices crop from this app SSADV guide the crop advisor of spices crop for the cultivation practices to be undertaken. This app need not internet facility, provide crop advisory of spices for farmers which was based on climatic condition, growing condition, resources available and locality, Easy to handle, and user cannot go to next step without providing selected parameters.

2.3 Survey of App

Kokate and Singh (2013) carried out study on use of mobile technology for giving agricultural information and advice to smallholder farmers across the country has shown a circulation of a diversity of messages, decreasing the cost and availability

of data at the correct moment. Because a platform of technology, subject matter and ground level institutions has been created with an structure for information up-gradation and delivery as per location specific requirements increasing their production and income can make a significant involvement to lowering poverty and starvation. There has been attempts in technology diffusion through ICT, but the mobile applications that have just been launched have the potential to revolutionize the information reach to resource-poor small farmers in real time. The creation of content for various clientele groups in various languages is a difficulty, but voice messages provide an easy choice for distribution and interpretation by consumers in the case of most smartphones. Some handsets may be limited in their ability to send text messages in many languages. The cost of audio messages is high, which may be reduced as technological improvements.

Ansari and Pandey (2013) carried out a survey in order to examine the capability and use of smartphones among farmers. The studied sample consisted of 180 farmers chosen at random from the kisan mela (farmers fair) at Pantnagar University. According to the research findings, 83.34 per cent of farmers had a smartphone for more than two years, and 72.33 per cent gained agricultural information from ‘fellow farmers.' All of the respondents polled stated they would like to register if Pantnagar University launched a mobile-based consulting service, but just 45% said they would be ready to pay a minimal price of 100 Rs per month. Farmers identified the following locations for the kinds of services expected through such a mobile phone service: disease detection and control measures (27.7 per cent), fertilizer application dose, method and time (27.77 per cent), harvesting time (25 per cent), marketing (22.23 per cent), and sowing time (19.45 per cent). The study demonstrates the necessity for agricultural advising services.

Karetsos Sotiris and Constantina Costopoulou (2014) studied on developing an app for Farmers using m-government in agriculture use their smartphones to perform a variety of work-related tasks such as sending/receiving email, weather updates, news, and markets. Agriculture Management Information Apps can produce computerized maps of farms to maintain a record of crop growth activities such as fertilizing, planting, and harvesting. Agriculture Information Resource Apps: These apps are mostly used as search tools or as a tool that aids in the identification of species. Farmers can also determine crop maturity by examining current and historical

growing degree days data for their farm's location. Farmers can use Agriculture News Apps to gain access to a website that provides agricultural management news, markets, weather, various notifications, farm business blogs, articles, and radio (e.g. AGWeb). Farmers may access local agricultural news, grain and livestock markets, weather predictions, and blogs. Weather Apps for use with mobile-phone weather apps. Some information apps provided by government entities were included in the M-government App. They can also compare current crop disease resistance ratings, infection symptoms, map diseases, and share disease photos with others (Crop Diseases).

Reddy et. al (2015) carried out a survey on Plant Disease Identification and Prevention Using Android Application of the suggested method on an android application for farmers. Image processing is often used to identify leaf-based plant disease detection systems. Farmers may use the online marketplace option to acquire and sell goods while also increasing their profit margins. Market rate guide, this feature assists users in gathering information about market prices in various markets. The weather report system's functionality plays an important role in assisting users in making decisions about water management, pesticide distribution, and fertilizer supplies for farmers.

Patel and Patel (2016) carried out a survey of Android-based mobile applications in many areas of agriculture and categories them depending on their purpose and category, where the app may have several functionalities. They discovered that each app has a unique set of characteristics. The FarmManager app was created exclusively for small marginal farmers. The Agro Mobile application is solely good for recognizing botanical species and detecting diseases. The Krishi Ville app provides updates on agricultural products, weather forecasts, and agricultural news. Farmers that desire to grow sugarcane will benefit from the Agriculture Supply Chain Management software. The Planning, Regulating, and Monitoring of Agricultural Device app is only capable of controlling the motor and pesticide proportions, as well as observing farming operations and irrigation. The E-agro app is used to identify leaf diseases. Horticulture information regarding flowers, fruits, and vegetables is provided through Farmers Helping Services.

Abishek *et. al.* (2016) evaluated an application that serves as a platform for the actual flow of agricultural commodities from farms to customers or retailers. It may have the goal of Agricultural Marketing on the internet as Digitalization, i.e. e-commerce, offering it more priority. As the objectives are getting the details about the products from farmers/consumers, analyzing the products obtained, recognizing the multiple criteria of examination required for it, achieving the quality expectations of a customer with the support of agricultural experts, getting the right worth of the verified product, and posting it in the application. The products were forwarded to agricultural experts. The specialists were fully unbiased and will assess the quality of the product delivered. The products were evaluated based on specific criteria such as quality, quantity, and pricing. Ratings were assigned to things based on their quality. Cereals and pulses, seeds, spices, vegetables and fruits, fertilizers and fodder are only a few of the essential agricultural items that are widely grown.

The products were produced by the Farmers and then it given directly to the hands of the customers with the help of our application. They achieved their main goal through this application, which was to boost farmers' profit margins and ensure that they receive a fair price for their work.

Belakeri *et. al.* (2017) surveyed on technology transfer and human resource development, the essential mechanism in the farming development process. It had been mentioned that the application of Information and Communication Technology (ICT) devices has the potential to revolutionize the economy of livestock, agriculture, and rural craftsmen in India. Mobile phones will transform farming extension with their remarkable uses and will be a huge help to the extension system's human resources.

Baktha (2017) conducted research to eliminate all the difficulties encountered in mobile application development, as well as the methods and rules for creating a successful smartphone application. The primary findings of this study were that the number of applications is rapidly increasing, competition for successful app creation is high, developers face problems in mobile app development, steps must be taken to assure successful application development, and more downloads, more users, and fewer bugs indicate successful application development. Developers should keep an open mind and be well-informed about current technology, requirements, and events

in the smartphone application sector. Creating fresh and unique apps has the potential to generate revenue.

Giri and Navven Prasad (2017) carried out a study on work awareness focuses on Indian farmers since it targets the key issues of receiving market updates for various goods. Farmers, on the other hand, create and use their own apps for specific purposes. According to their functionalities, each of these apps has a particular usage. Many applications were being used for many types of functionality related to farming operations such as cropping information, pesticides, fertilizer, seed, crop sale, irrigation information, crop production estimation, weather information, and information about optimal farming techniques. Many of the application discovered were stagnant. It is suggested to use dynamic application instead static application.

Gyanappa *et. al.* (2018) conducted a survey of different mobile apps. In this there were three different apps surveyed. All the three apps showed similar line of action of working of app and features in the app is also similar to it. Among three of them the features of this apps are News and feeds section, Weather forecasting, Market price, Fertilizer and Pesticides dealer information, Information about crops, Farming Tools and Technology and Government of India Scheme for Farmers.

Barh and Balakrishnan (2018) conducted research on Agri-information dissemination whereas Mobile application one of such technology that can be utilized directly in agricultural growth. Although this channel of information dissemination is in juvenile phase but it's advantages can be seen in near future. The strategies for expansion of application based information require expulsion of obstacles like better modest handsets, compatible smart phones, multilingual platform, subsidizes internet packs, regular trainings and awareness amongst farmers. Smartphones are an example of overcoming obstacles by bridging the rural digital divide, offering financial benefits, and creating opportunities for social mobilisation through enhanced communication.

Sharma *et. al.* (2018) conducted a survey on overview of Mobile Android Agriculture Applications was created with a specific goal in mind and to provide the necessary functionalities. Cropping information, market prices, online buying for farmers, weather prediction, and daily agriculture updates are one of the functions. Market applications include information about market rates and provide access to

available SME (Small and Medium Enterprises) in the user's region based on their location. Information-based apps were knowledge-based applications that contain information about crops and their types suitable to different soil types, as well as the best techniques to use to increase productivity. This section also contains applications that provide information about the most recent government programmes and policies that can assist farmers. Weather applications contain data on the weather predictions for a specific location that has been supplied by the consumer or discovered by GPS. Advisory applications allow users to contact labs, scientists, or the Kisan Call Centre (KCC) nearby their area. Management-based technologies make it easier for users to govern small or medium-sized farms.

2.4 Testing of App

Muccini *et. al.* (2012) investigated research on testing of mobile applications despite of that it is becoming increasingly popular, it is still unclear whether they require any specific testing approach for its testing and evaluation. This project seeks to examine new research directions in mobile application assessment mechanization by responding to three research questions: (RQ1) Are mobile apps distinct from traditional ones, necessitating the development of new and specialized testing techniques? (RQ2) What are the new difficulties and research areas in testing android application? (RQ3) What role does automation serve in testing android apps? The issues by examining the current state of the art in smartphone application development and testing and sharing our perspective have been responded.

Nimbalkar (2013) carried out a study on Mobile application Testing is process of evaluating applications designed for mobile devices. It is the process of testing an app for functionality, usability, and performance problems. Mobile app testing differs from application software testing, in that addition to typical operational and UI criteria, hardware, screen resolution, platform, connection challenges, and other aspects were also analyzed. Functionality, usability, and reliability are all aspects of smartphone application testing. The trend of using smartphone emulator testing is substitute of testing technique as traditional web and desktop software. We must develop a new strategy and procedure, which is going to take into account to know actually what is the mobile or smartphone world, what it constitutes of and the adjustments. It calls for our traditional testing patterns and techniques.

Méndez-Porras *et. al.* (2015) carried out study on Mobile application assessment that takes advantage of cloud computing's characteristics, such as parallel and distributed platforms, virtualization, and software services. Cloud computing enables application instances to operated separately on a range of operating systems and to connect with others through web services. By minimizing the cost of purchased equipment, research sites could share real devices. The implementation of parallel test cases can help to reduce testing period. However, few systems are currently taking advantage of cloud computing for testing. There were numerous suggested tools for mobile application testing, but several of tools are still not available for download on the web. Other tools can be downloadable, but we believe they are difficult and tricky to use and have inadequate user instructions. Some technologies are very focused and only serve to test a few characteristics of the apps.

CHAPTER -III

MATERIALS AND METHODS

CHAPTER-III

MATERIAL AND METHODS

This chapter focuses on the designing and methods of development of android based mobile application, testing of developed application, checking the compatibility after successful designing. The app was designed for put forward package of machinery for crop cultivation practices of eight crops. The collection and formulation of databases of eight crops have been undertaken at Department of Farm Machinery and Power Engineering and AICRP on Utilization of Animal Energy, VNMKV, Parbhani.

In this chapter, discussed concerning mobile apps, techniques, resources, approaches, and mobile app designing. For the proposed research study, many kinds of software applications tools were utilized to create the application. It includes Android, Software Development Kit, Android Studio, Programming Language, JAVA and JAVA Script, SQLite tools.

1. Farmers should have smartphone with them.
2. Farmers should be well equipped with the smartphone mobile application.
3. Application will provide crop details.
4. It will also provide package of operation

Developing Android based mobile application for the package of operation to be taken consideration for the selected crops. The followings crops were selected as the major crops in this region for design of mobile application.

Selected crops

1. Cotton
2. Soyabean
3. Turmeric
4. Maize
5. Sorghum
6. Sugarcane
7. Groundnut
8. Pulses

Package of operation for above selected crops were taken into consideration in following sequence

1. Land Preparation
2. Sowing/ Raised Bed Sowing
3. Intercultivation
4. Ridging & Furrow making/ Reopening of furrow
5. Spraying Weedicide/ Insecticide
6. Harvesting
7. Threshing
8. Crop Residue Management

3.1 Diverse software packages and technologies

By definition, software is a collection of computer programmes, methods and documentation which perform a variety of functions on a computer system. At its most fundamental, computing software is made up of programming languages, which is made up of a collection of binary values numeric indicators that indicate processor instructions. In a specified sequence, microprocessor instructions affect the state of computer hardware system. In a nutshell, software program is the language that a system communicates.

There are various forms of computer software, including:

3.1.1 Programming software:

This software exists in the form of resources that aid a developer in the creation of computer application programmes. Text editors, processors, and interpreters are tools that assist programmers in teaching a software system. To write the software, coding of JAVA and various programming languages are used.

3.1.2 System software:

It aids with the functioning of computer hardware and system software. Operating systems, peripheral devices, data centers, windowing systems and tools are examples of system software. System software aids application programmers in extracting away from a computer's hardware, storage and other underlying issues. The programme is primarily implemented using Windows 7.

3.1.3 Application software:

It allows the end-users customers to execute certain tasks. Application software includes business software, libraries, and instructional software. In this section, An Android mobile application is created. As a result, the user must have an Android smartphone. Only then user be able to use the android app.

3.1.4 Malware:

Malware is a general term that refers to every harmful programme that poses a serious threat to computer systems. Malware includes adware, spyware, viruses, worms, Trojan horses, and scareware.

3.1.5 Adware:

Adware are software that allows advertising to be broadcast and downloaded on a device. Adware is created by hackers as a revenue-generating tool.

3.1.6 Inventory management software:

This type of application software assists a company in monitoring its products and components in terms of both quality and quantity. The inventory control activities of a warehouse include internal warehousing movements and safekeeping. Inventories software assists businesses in organizing inventory and streamlining the distribution of items throughout the organization.

3.1.7 Utility software:

Utility software, often defined as a service routine, aids in the control of computer equipment and software applications. It only has a limited set of functions. Utility software commonly includes disc defragments, system utilities, and viruses' scanners.

3.1.8 Data backup and recovery software:

The best data backup and restore software goes beyond simply duplicating data files. This software frequently accommodates the needs of the user in terms of determining when and what should be backed up. Backup and restore software preserves the appropriate file organisation and allows for the restoration of secondary storage.

3.2 Platform Used for Mobile App

There are many operating systems being used on several smartphones manufactured by different companies. This operating system includes iOS for Apple, from the Symbian system of operating to Windows over the trend of Android now a day used by all smartphone company. It is also an free and open - source platform with several significant applications and functionalities in mobile application development. Android is a Linux-based operating system (OS) with a Core JAVA language. The Android Software Development Kit (SDK) contains many of the technologies required to develop and create Android applications.

The development of a mobile app necessitates understanding of Android's architecture. The programming is mostly written in the JAVA programming and compiled to binary code within the Android Studio build environment. Whenever software is installed on such a device, the Android Runtime (ART) compiles the bytecode to domestic style utilized mostly by CPU.

3.3 Project structure

Every projects in Android Studio has one or more categories or module, each with its own set of source code as well as resource files. Different type of modules are:

- Android app modules
- Modules for the library
- Modules for Google App Engine

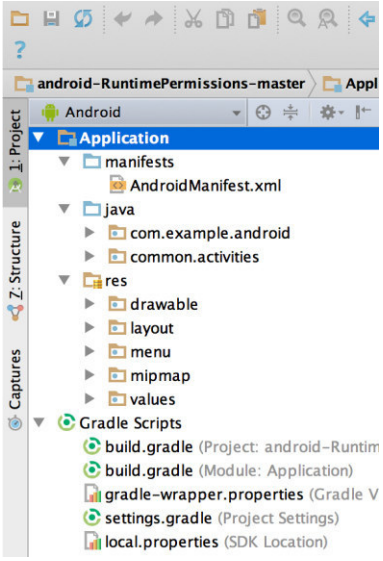


Fig 3.1: Project files in Android view

As shown in Figure 3.1, Android Studio needs to display source code in the Android project interpretation by insolvency. This display is grouped by sections to enable rapid access to the important source assets of the application.

Under Gradle Scripting, able to see all of the built files, and every app package has the following folders:

- **manifests:** This folder contains AndroidManifest.xml file.
- **java:** This directory contains all Java source code files, as well as JUnit test code.
- **res:** All non-code elements, such as XML templates, UI texts, and bitmap pictures, are stored in this directory.

Select Project from the Program drop down (in figure 3.2, it's Android) to examine the real file plan of the system.

Users can also change the way the project files are displayed to focused and target specific parts of developing apps. Accessing the designs of Issues view of your design, for example, exposes links to any recognized code and language problems, including a missing XML component ending tag in a layout file.

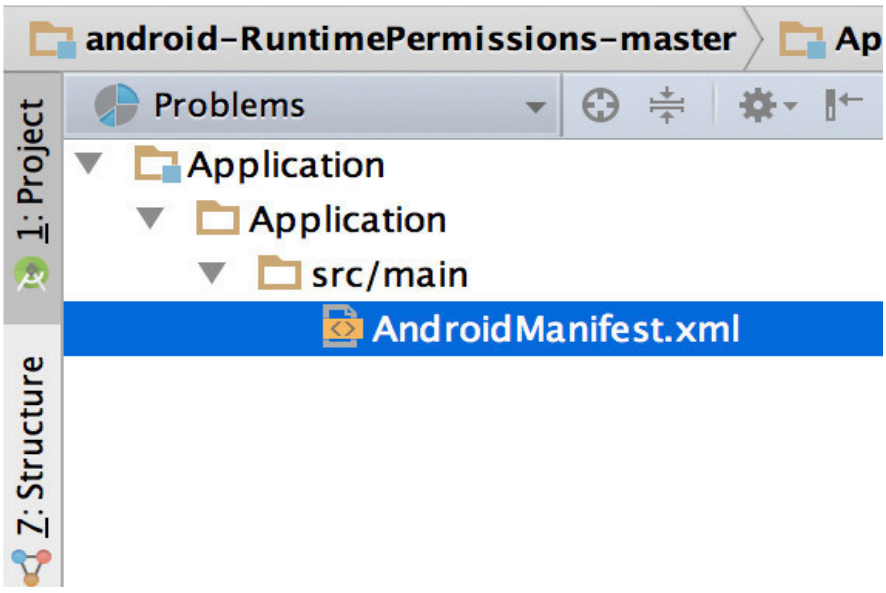


Fig 3.2: The project files of error view, showing a layout file

3.4 User Interface (UI) The Android Studio main window is made up of several logical areas identified in figure 3.3.

1. The **toolbar** performs varieties of task, including executing program code of an application and launching Android tools.

2. The **navigation bar** assists in navigating developed project and open files for modification. It shows a more streamlined version of the layout in the Project screen.

3. The **editor window** is created and modified in the program window. The editor are able to change window based on the prevailing file type. When examining a pattern of file, for example, the editor shows the Layout Editor.

4. The **tool window bar** wraps around the exterior of the IDE window and has buttons for expanding and collapsing specific tool windows.

5. The **tool windows** provide access to specialized activities such as project management, search, and configuration management, among others. Tool window get expanded and collapsed.

6. The **status bar** shows the current state of application and the IDE, as well as any cautions or notifications.

By suppressing or rearranging toolbars and tool windows, users can free up more display in the preview pane. Most IDE functions can also be accessed using shortcut keys.

Exploring of source code, database systems, commands, user interface components, and so on at any moment by double-clicking the Shift key or tapping the magnifying glass in the corner of top right hand of the Android Studio window.

3.4.1 Tool windows

As work start, Android Studio detects context and displays relevant utility screens. The most frequently used utility windows are pinned to the tool window bar at the program window's boundaries by default. Click the window icon on the lower left-hand corner of the Android Studio window to reveal or conceal the whole tool window bar.

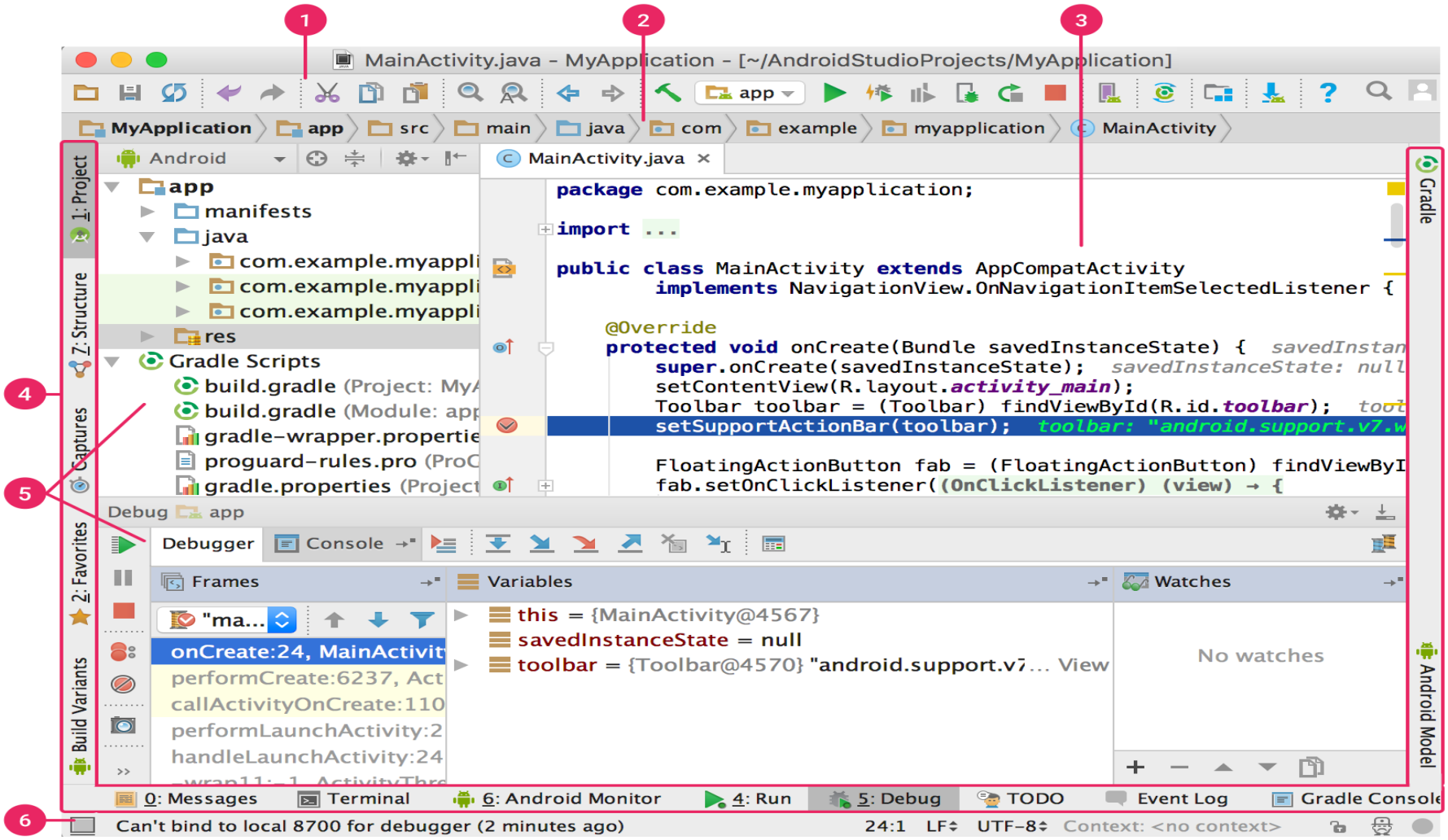


Fig 3.3: The Android Studio main window

Table 3.1: Shortcuts for some useful tool windows

Tool window	Windows and Linux	Mac
Project	**Alt+1**	**Command+1**
Version Control	**Alt+9**	**Command+9**
Run	**Shift+F10**	**Control+R**
Debug	**Shift+F9**	**Control+D**
Return to Editor	**Esc**	**Esc**
Hide All Tool Windows	**Control+Shift+F12**	**Command+Shift+F12**

3.4.2 Navigation

Here are a few suggestions which helps to navigate Android Studio.

- Use the Navigate to File option to browse to a file, directory or folder. Control+Shift+N (Command+Shift+O on a Mac) will bring up the Browse to file option. Add a / to the ending of your statement to search for directories rather than documents.
- Using the Navigate to Symbol action, this navigate to a technique or field by keyword. Control+Shift+Alt+N (Command+Option+O on a Mac) will bring up to browse to Symbol function.
- Press Alt+F7 (Option+F7 on a Mac) to identify all the lines of code that reference the class, procedure, field, function, or statement at the present current point.

3.4.3 Style and formatting

As modification was carried out, Android Studio uses the formatting and styles that have been selected in program work settings. The change in code style parameters for each programming language, includes tab and indentation, spaces, enclosing and brackets, and blank lines. To change the appearance of your code, go to Settings > Code Styles, click **File > Settings > Editor > Code Style** (**Android Studio > Preferences > Editor > Code Style** on a Mac.)

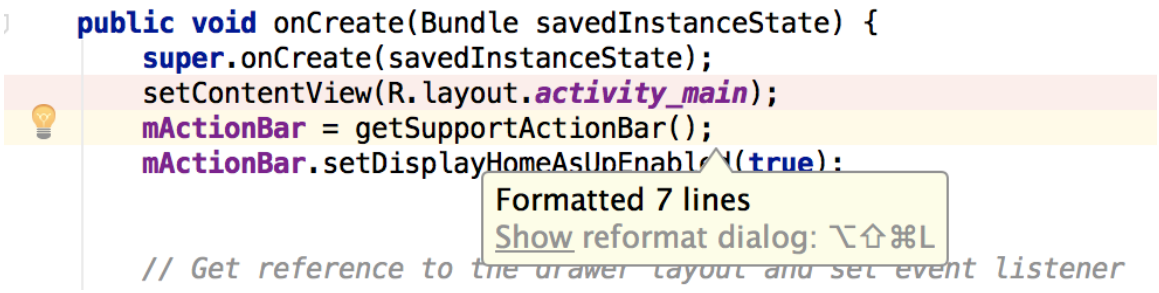


Fig 3.4: **Code after formatting.**

3.4.4 Version control basics

Version control systems (VCSs) supported by Android Studio include Git, GitHub, CVS, Mercurial, Subversion, and Cloud Services Source Repositories.

After loading application into Android Studio, utilise the VCS menu choices in Android Studio to activate VCS compatibility for the preferred version control system, build a repositories, import documents into version control, and execute additional version control operations:

1. In Android Studio **VCS** menu, click **Enable Version Control Integration**.
2. In drop-down menu, select a version control system to linked and associate to the project root, and then click **OK**.

On selection of platform, the VCS menu now presents a number of system version control alternatives.

3.5 Gradle build system

Android Studio's build system depends on Gradle, with additional Android-specific functionalities supplied by the Android module for Gradle. This build method is available as an integrated tool first from Android Studio menu as well as from the command - line interface. It can utilize the build system's functionality to do the following:

- Personalize, configure, and expand the setup and build process.
- To use the same application and modules, develop multiple APKs with a varied set of features.
- Make use of shared code and services across source sets.

Accomplishing everything by utilizing Gradle's extensibility without altering your app's main source code. Build.gradle files are used by Android Studio. Each project contains a single top-level development file for the overall project, as well as multiple module-level build scripts for each section. Whenever importing an existing program, Android Studio creates the appropriate build files for you.

3.5.1 Build variants

The build system can assist in generating many versions of the same software from a specific project. Free and premium version of software or released various APKs on Play Store for different device setting is not important.

3.5.2 Multiple APK support

The support for multiple APKs enables to produce multiple APKs dependent on display resolution or ABI. For example, generate distinct APKs of an app for hdpi and mdpi display resolutions while still treating them as an unique variant and enabling them to share test APK, javac, dx, and ProGuard parameters.

3.5.3 Resource shrinking

Android Studio's resource reduction feature automatically removes unneeded resources from packed app and library dependency. For example, if application uses Play Store Service to access Google Drive features but does not currently use Google Sign-In, resource shrinking may eliminate the different drawable components for the Sign-In Key.

3.5.4 Managing dependencies

The build.gradle file specifies the development's dependency by title. Gradle finds dependency and make them accessible to the project. In build.gradle file, you can specify module dependency, remote binary dependencies, and local binary interdependences.

3.6 Debug and profile tools

Android Studio includes inline troubleshooting and performance evaluation tools help to debug and improve the performance of your program.

3.6.1 Inline debugging

Inline troubleshooting can be used to improve code walkthroughs in the programmer view by verifying references, equations, and parameter values inline. Inline debugging information comprises the following:

• Variable values inline

• Referring objects that point to a specific object

• Process return values

• Lambda and controller expressions

• Tool tip values

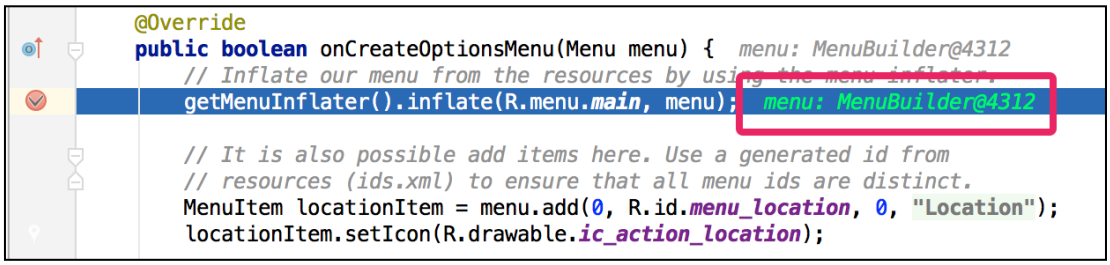


Fig 3.5: An inline variable value

3.6.2 Performance profilers

Android Studio includes performance profilers that allows to easily watch app's processor and CPU Memory consumption, detect deallocated objects, find memory leaks, enhance image quality, and evaluate network requests. Open the Android Profiler tab when app is running on such a smartphone or emulator.

3.6.3 Memory Profiler

Memory Profiler allows to trace memory allocation and see where entities are stored when you execute specific operations. Recognizing these allocations allows to improve app's functionality and memory usage by modifying the function calls associated with those actions.

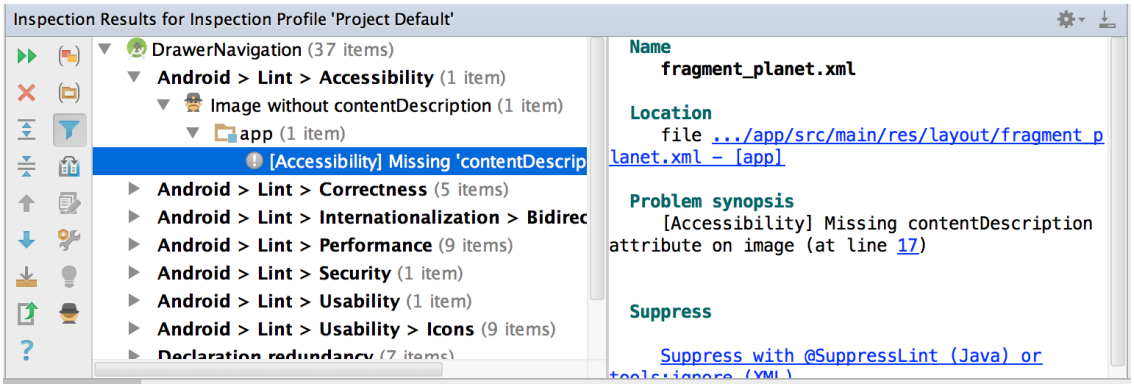


Fig 3.6: The results of a Lint inspection in Android Studio

3.6.4 Code inspections

When you compile the programme, Android Studio executes preset Lint and other IDE inspections to assist you easily find and correct format of quality issues in the code. The Lint tool examines your Android application source files for potential flaws as well as optimization enhancements for accuracy, protection, performance, readability, accessibility, and internationalization. To help with the coding programming workflow, Android Studio also does IntelliJ code evaluations and verifies annotations.

3.6.5 Annotations in Android Studio

Annotations for dependent variable, parameters, as well as local variables in Android Studio can helps to catch errors like null pointer exceptional cases and resource type quarrels. The Support-Annotations library is packaged in the Android Support Data base for use with Android Studio by the Android SDK Manager. In code inspection, Android Studio validate the configured annotations.

3.6.6 Performance profiling

To assess the CPU, storage, and signal strength of app, open the Android Profiler by choosing **View > Tool Windows > Android Profiler** in the sidebar.

3.6.7 Android Profiler

The Android Profiler substitutes the Android Monitor features in Android Studio 3.0 and beyond. The Android Profiler utilities provide authentic data to assist in understanding how app utilises CPU, storage, networking, and battery capacity. Android 5.0 (API level 21) and higher are supported by the Android Profiler.

Select the platform on which to profile the programme if presented by the **Select Deployment Targeted** window. Linked a USB device but don't see it displayed, make sure you've configured USB debugging. The use of Emulators or a rooted device, the Android Profiler displays all executables processes, even if they aren't debuggable.

3.7 Application Development Environments

3.7.1 Android Studio

The creation of an Android-based mobile application using the Android Studio programme. The developer must be proficient in a programming language in order to operate this software. This may come with a Software Development Kit (SDK). Android Studio is Google's official software development environment for developing Android applications. This was officially unveiled on May 16, 2013, at the Google summit. Android Studio is available for free under the Apache License 2.0. Android studio was created by merging a powerful code editor with a fully UI build mechanism based on gradle. The app designer's research project, Android version 2.1 (released in April 2016) with JDK 1.8 was utilized. An android studio is featured, which includes tool windows, a source code editor , and layouts editor utilities.

3.7.2 AVD and API level

An Android Virtual Device (AVD) is a system configuration that may be executed on the Android emulator. It collaborates with the emulators to provide a virtual environment upon which to download and install Android apps to run. In Android Studio, AVD is useful for assessing and ensuring the correct functionality of

the programme. The software that emulates the genuine device while sharing the system resources of the server devices is known as an AVD. AVDs are typically used in conjunction with their respective APIs.

3.7.3 Android Studio

Android Studio is an Integrated Development Environment (IDE) for developing Android apps that is based on IntelliJ IDEA. While IntelliJ is a strong code editor and developing tool, Android Studio provides even more capabilities that improve efficiency when developing Android apps, such as:

- A customizable Gradle-based development system.
- A powerful and functionality emulator.
- A unified development environment including all Android devices.
- Use Apply Changes to deploy code and module changes to the running programme without having to reboot it.
- Code templates and GitHub integrations to aid with the development of typical app features and the import of source file.
- A wide range of testing frameworks and tools.Lint tools to catch performance, usability, version compatibility, and other problems.
- Support of C++ and NDK.
- Google Cloud Platform compatibility is built-in, making it simple to combine Cloud Based Communication and App Engine.

3.7.4 Android library

An Android repository has the same structure as an Android app modules. It can contain everything required to develop an app, such as program code, resource files, and an Android manifests. An Android library, on the other hand, rather than converting into an APK that executes on a device, converts into an Android Archive (AAR) file that use as a dependence for an Android app unit. AAR files, as opposed to JAR files, provide the following functionality for Mobile applications:

1. AAR files now include Android resources as well as a manifests file, allowing you to include shared resources such as designs and drawables in combination to Java classes and functions.
2. AAR files may include C/C++ libraries for usage by the application by C/C++ coding.

A library module is useful in the following situations:

1. When developing several apps that share some elements, such as activity, resources, or UI designs.
2. When creating an app with various APK versions, such as a free and premium edition, and you require the same basic components throughout both. Simply place the files to recycle in a library module and make the library a dependent for each app section. This page shows methods to do both simultaneously.

3.8 Developer workflow basics

The process of developing an app for Android is fundamentally similar to that of other application platforms. However, in order to create an effective and well-designed android application, you will need some specific tools. The list below presents an overview of the Android application development process which includes some Android Studio components you should utilize throughout each stage of development.

3.8.1 Set up your workspace

Install Android Studio and start working on a project. The tutorial on Building your 1st app for an overview with Android Studio that explains some Android development principles.

3.8.2 Write your app

Android Studio contains a number of tools and technologies to assist in working faster, writing excellent code, designing a UI, and creating resources for various device kinds.

3.8.3 Build and run

During this stage, you convert your application into a debuggable APK packages which you can install and run on an Android operating system or a

smartphone. Also start personalizing the build. You can, for example, establish build variations that generate multiple versions of app out of the same projects, and compress the coding and resources to build the app simpler.

3.8.4 Debug, profile, and test

This is now the iterative stage in which it continue to write the app while focusing on bug elimination and app performance and functionality. Developing tests will assist in the objectives.

3.8.5 Publish

Once you're ready to make the application available to consumers, there are only a few more steps to take, such as modifying and existing it, creating an Android App Package, and registering it with a key.

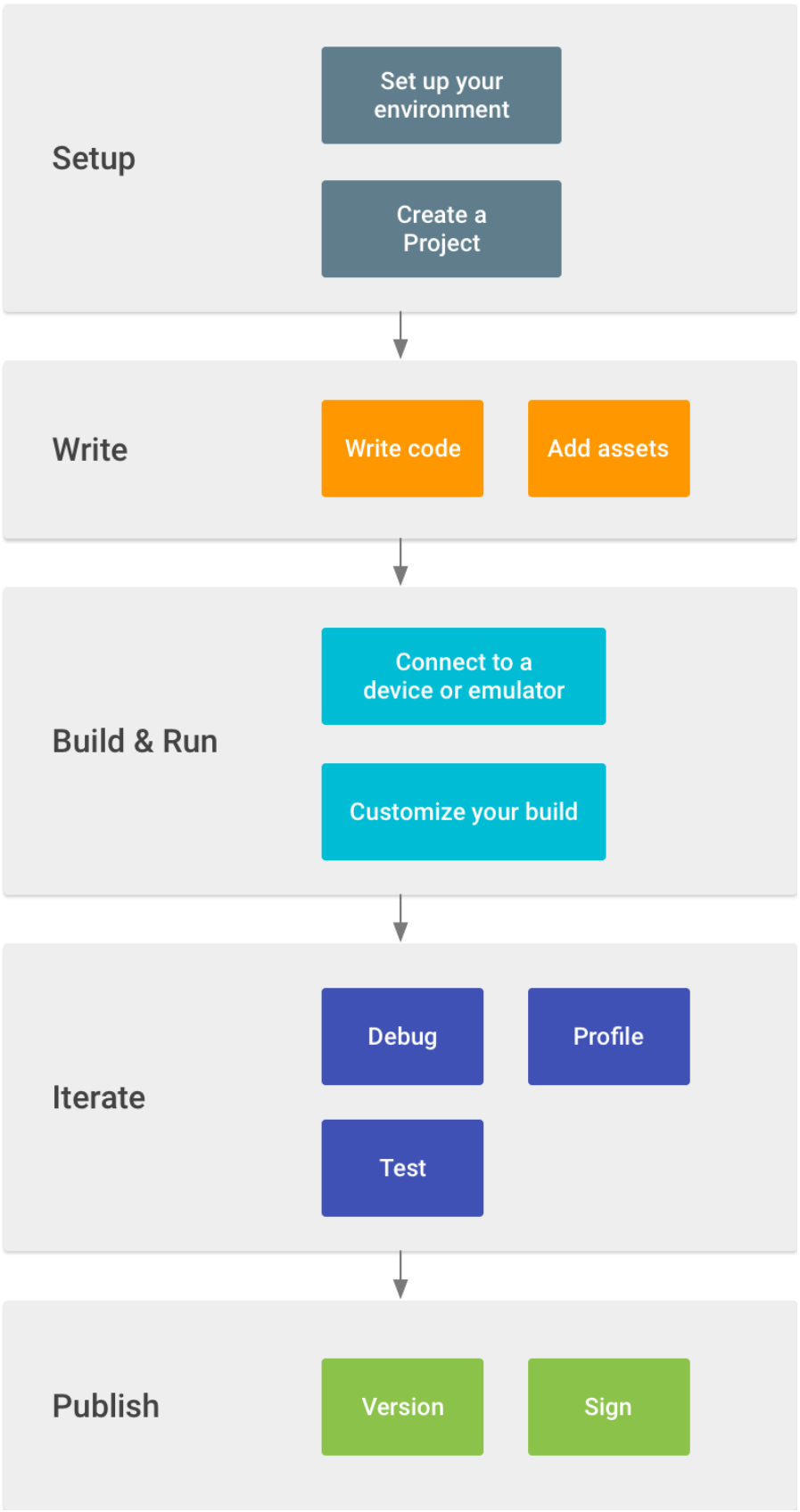


Fig 3.7: Workflow of android app

3.9 Key Android Build/ Developing Tool

3.9.1 Android Studio

Among Android development tools, Android Studio is the first. It's an official integrated platform for developing Android apps that allows to simply edit coding, debugging, and test.

In 2013, software made a big splash and dethroned Eclipse Android Development Tools as the exclusive IDE for natively Android apps. It's completely free and openly supported by a thriving Android developer community.

3.9.2 AIDE

It is an Android IDE that allows users to create an Android app on Android platform. AIDE allows you to not only create code in the tablet or smartphone, but also execute, test, and inspect it. In comparison to Android Studio or IntelliJ IDEA, this platform is a good choice for new developers. The only drawback is that it only allows Java and C/C++. The Android IDE is utilised when developing an app in Kotlin.

3.9.3 Gradle

Gradle is a build integrated system process that is open - source software. This method is suited for multi-project, large-scale construction. Gradle makes it simple to include a third-party resource with a single line of code. Gradle is mostly used for Java-based Android software design; however there are also Groovy and Scala plugins.

3.9.4 Android Asset Studio

It's a great collection of simple tools for creating the many types of logo and icons that need designing, building and developing applications.

3.9.5 IntelliJ IDEA

JetBrains developed a Java IDE that supports Android. It's a better substitute to Android Studio, and it's primarily used for basic projects. IntelliJIDEA is quick and

includes a number of developer tools straight out of the box, including smart code completion, immediate debugging, restructuring, and JetBrains plugins.

3.9.6 Source Tree

A simple and free tool for managing Git repositories through the Git GUI. You can see all modifications, commits, and sections without having to type a single command in the command - line interface. It is accessible for both Mac and Windows clients.

3.9.7 Android Debug Bridge

This is an effective command-line utility that enables communication with a gadget. ADB helps to exchange information with an operating system in Android emulator or a real-world Android device connected to external gadgets (via USB or Wi-Fi network) (tablet, phone). The utility is part of the Android SDK System Toolkit package and corresponds to the Android SDK Tools category.

3.10 Testing Tools

3.10.1 AVD Manager

Android Virtual Device is a valuable tool for Android Studio. It allows developers to generate virtual emulations of all Android devices on the single desktop machine in order to evaluate the performance and interactivness of an app. The AVD management eliminates the requirement for each hardware machine, pixel density, and display.

3.10.2 NimbleDroid

A system for testing the Android app for flaws before publishing it to Play Store. Because it speeds up the testing stage, this technology adds a lot more value to the Quality Assurance system.

3.10.3 Appium

Appium is without a doubt one of the greatest mobile app functional testing tool utilised by the majority of professional developers. Appium is a powerful method for evaluating or testing web and mobile apps, and it also works well with hybrid

apps. There is no need to change any code in the program because Appium interfaces mostly with the Android and iOS platforms. Appium is also intended for automated functional testing, with the goal of improving the overall effectiveness of applications.

3.10.4 Robotium

Robotium is a testing platform that is only intended for use with Android apps in an automated software framework. Robotium is designed primarily for black-box testing of Android applications. The test codes are created using JavaScript. Android SDK, Eclipse for the testing procedure, Android development Kit, and JDK are some of the extra prerequisites for this utility to work smoothly.

3.10.5 Xamarin.UITest

UITest is a framework that may be used to test overall UI acceptability of iOS or Android applications. It is ideal for testing Android, iOS, and native apps. It provides cross-platform testing procedures, and the test codes are designed in C.

3.10.6 Espresso

Espresso, a smartphone app testing tool created by Google, is integrated into Android Studio. This is an excellent tool for anyone who needs to conduct tests on core Android applications. Espresso can just develop Android UI Tests to assist the development of features. It creates tests with Java and Kotlin and provides a basic API which can be easily modified. These UI tests can be run on actual devices and emulators to gain a better understanding of the performance. Furthermore, there is an integrated test recorders that enables for the recording of tests rather to writing comprehensive code.

3.11 Project Overview

In Android Studio, a project comprises everything that specifies your app's workspaces, from programming language and assets to test code and build options. When you begin a new program, Android Studio develops the required structure for almost all of your resources and displays them in the Project window somewhat on left side of the IDE (click View > Tool Windows > Project). This provides a high-level description of the essential components in the project.

3.12 Modules

A module is a combination of source codes and build parameters that allows to break your project into different functional components. Project may contain one or more modules, and one unit may rely upon that module. Every module can be produced, evaluated, and debugged individually.

Additional modules are frequently useful for developing code libraries within your own projects, or construct various sets of code and resources for various device varieties, while keeping all files scoped even in the same research work and sharing certain code. There are several sorts of modules available in Android Studio:

3.12.1 Android app module

This provides as a framework for app's program code, resource files, and app-level parameters like the module-level compilation and Android Manifest file. Start creating a new design, the module identity is set to "app."

Android Studio provides the following sorts of app plugins in the Create New Component window:

• Phone and Tablet Modules

• Wear OS Modules

• Android TV Modules

• Module for Glass

They also providing the necessary files as well as some code templates associated to the app or device type.

3.12.2 Library module

This serves as a container for reusable source code, which is use as a dependent in other app modules or importing into other programs. A library module is basically identical to an app module, but when it is generate a code archive file rather than an APK, so it cannot be installed on a smartphone.

In the **Create New Module** window, Android Studio offers the following library modules:

- **Android Library:**

This kind of library can include all file formats accessible in an Android project, such as source code, tools, and manifest files. The build output is an Android Archive (AAR) file, it include as a dependence in your Android app modules.

- **Java Library:**

Only Java source codes can be found in this type of archive. The build produces a Java Archive (JAR) file, it include as a requirement in your Android app modules as well as other Java programs.

3.12.3 Google Cloud module

This serves as a container for the Google Cloud backend code. This module contains the necessary code and resources for a Java App Service backend that connects to your app via plain HTTP, Cloud Endpoints, and Cloud Messaging. Build your backend to supply the cloud computing requires in application.

When you use Android Studio to create Google Cloud based module, it can manage both the application coding and the backend software code in the same project. It also execute and test backend code locally and release your Google Cloud module using Android Studio.

3.13 Project files

In the Android interface, Android Studio shows the source codes and project files. This layout does not mirror the actual file structure on disc, but is grouped by modules and data formats to simplify navigating between your project's important source documents, hiding unnecessary files or directories which are not accessible and used. The following are among the structural distinct between the structure on disc include the following:

- Exhibits all build-related setup files for the application in a top-level Gradle Scripting group. Shows all manifest files for each module in a module-level group

- Displays all substitute resource files in a single group rather than in different folders for each resource qualification. For example, all density variants of the launchers icon beside each other is visible.

Files are displayed in the following sections into each Android app module:

manifests: This directory contains the AndroidManifest.xml file.

java: This directory contains the Java resources files, that are grouped by package name and include JUnit test code.

res: This directory contains all non-code data, such as XML layouts, UI strings, and bitmap graphics, which are organised into subdomains.

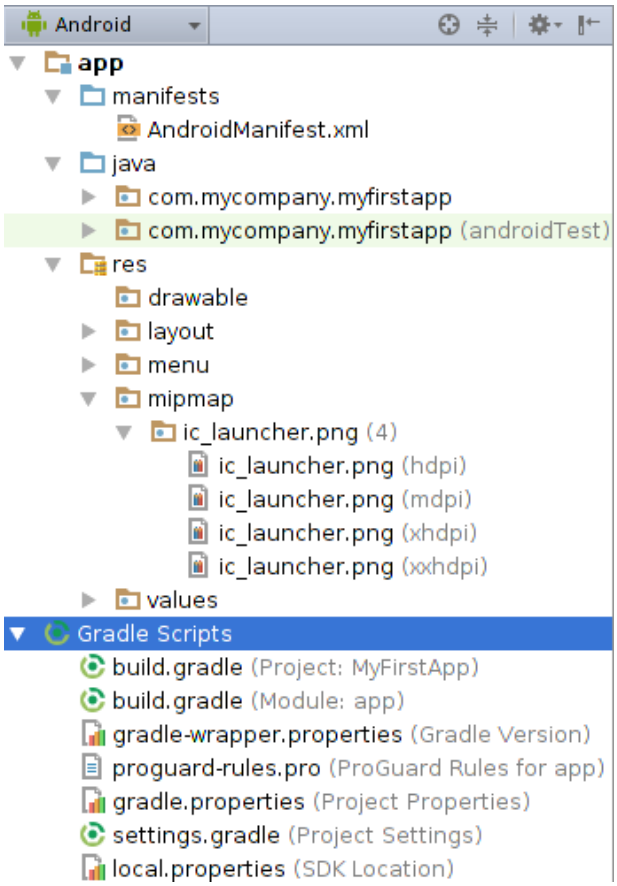


Fig. 3.8: Project files

3.14 The Android project view

Select Project from the option just at top of the Project window, dropdown it to show the true file structure of the project in the system, including all folders concealed from the Android view.

Choose Project view, that able to see a many more files and folders. The following are the most important:

module-name/

build/

Contains build outputs.

libs/

Contains private library.

src/

Contains all source code and resource files for the module in the following subdirectories:

androidTest/

Contains code for instrumentation tests that compile and run on an Android device.

main/

Contains the "main" sourceset files: the Android code and resources shared by all build variants (files for other build variants reside in sibling directories, such as src/debug/ for the debug build type).

AndroidManifest.xml

Describes each of its components and the nature of the application.

java/

Contains Java code sources.

jni/

Contains native code using the Java Native Interface (JNI).

gen/

Contains the Java files generated by Android Studio, such as your R.java file and interfaces created from AIDL files.

res/

Contains application resources files, such as drawable files, design files, and UI panel and strings.

assets/

Contains file that should be compiled into an .apk file. You can navigate this directory in the same way as a typical file system using URIs and read files as a stream of bytes using the AssetManager.

test/

Contains code for local tests that run on your host JVM.

build.gradle **(module)**

This defines the module-specific build configurations.

build.gradle **(project)**

This section defines build settings, which applies to all modules. Because this file is critical to the project, it must be kept in revision control alongside the rest of the source code.

3.15 Modules

The Modules configure section allows to modify configuration settings for each module in the project. The configuration page for each module is broken down into the following tabs:

- **Properties:** Defines the SDK and build methods to be used to compile the module.
- **Signing:** Defines the certificate that will be used to sign your app.
- **Flavors:** Allows to construct numerous build flavours, each of which provides a collection of configuration options such as the module's base and targeted SDK versions, as well as the version code and version name.
- **Build Types:** This feature allows to define and alter build settings, as explained in Customizing Gradle Builds. Every module includes debug and release development types by default, but then it can define additional as needed.
- **Dependencies:** Displays a list of this module's package, file, and module constraints. This pane allows to add, update, and delete dependents.

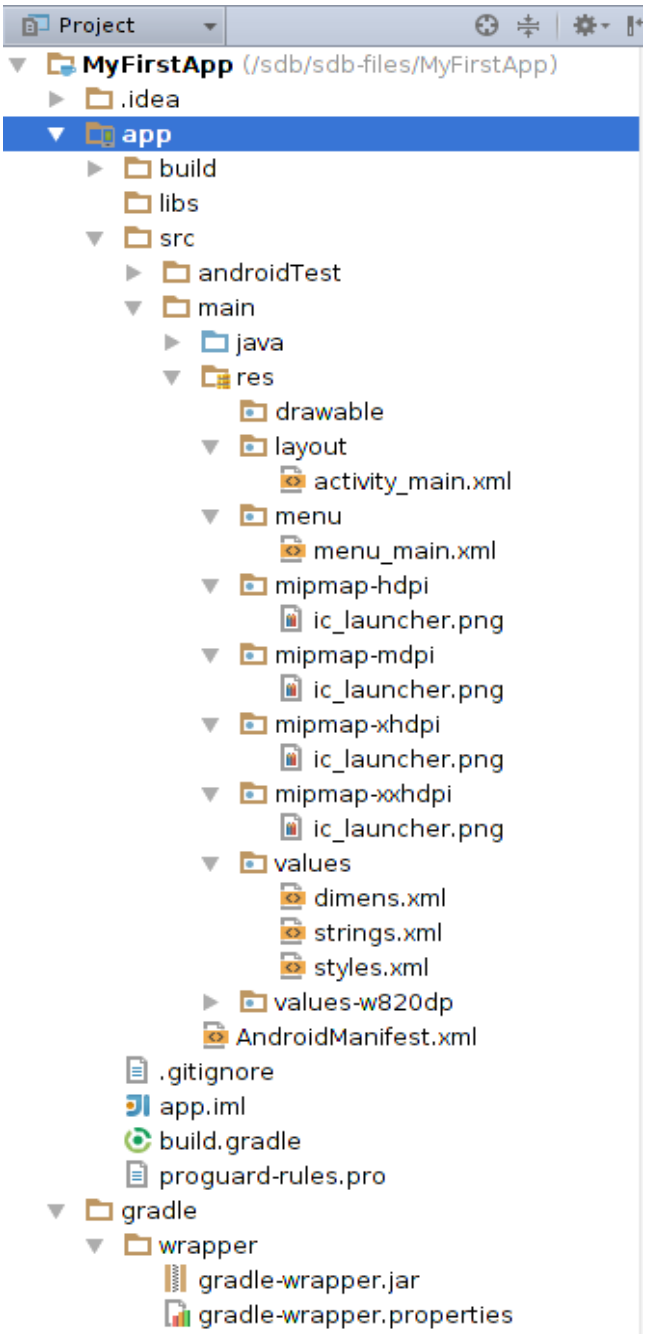


Fig. 3.9: Modules

3.16 Mobile Application Development Process

An advanced application development workflow includes ten important stages. Following are the key phases of Application Development.

1. Idea of Application
2. Concept creation/ strategy
3. Data Collection
4. Analysis and Planning
5. UI/UX Design
6. Implementation
7. App Development
8. Execution
9. Testing
10. Deployment & Support

Fig 3.10: Process of app developing

3.16.1 Collection of Data

a. Selection of crop
b. Mode of Operations
c. Activities in Agriculture
d. Implements/ Equipments for operations
e. Analysis of Information
f. Testing of Application

a. Selection of Crops:

The selection of crops is done, based on the major crop in selected areas. Taking into consideration of crop, cropping pattern, and yield of crop grown this only is selection parameter for the package of operation for a different crop.

b. Mode of Operation:

The modes of operations the implements or types of equipment categorized on the basis of Manually, Animal, Tractor, and Self-propelled. According to the use of implements in a mechanized way with less human drudgery and ergonomically designed machinery with efficient and effective use of implements or equipment.

c. Activities in Agriculture:

The activities in Agriculture are considered as different operations undertaken while farming. The operation may be categorized as seedbed preparation, sowing, inter-cultivation, spraying, harvesting, threshing, and crop residue management. For this different types of operations are undergone through crop production.

d. Analysis of Information

Collection of Information on different variety of machines essential for a different types of crops. The collected data and information of different machines from a different company registered in the Indian Market. In consideration of collected data get sorted in a sequence of machines required for different operations for the crops. The data may contain in the format of the description of the machine, video and images, specifications, and availability of machines (addresses). This type of data format get sorted according to the crop and making up the package of operation for the machines of the different major crop of Maharashtra.

e. Testing of Application

Testing of a developed app in various smartphones with a minimum API range. This shows the compatibility of the app on different smartphones. The required minimum API is 5.0 and above 5.0 is compatible with the app. The testing of compatibility is necessary due to every farmer does not have a high-rated version smartphone to take into consideration of basic version of the smartphone which is

easily available on every device. This is only the reason for taking into consideration and designing of minimum API range based mobile application.

The second consideration is that nearly all the small-scale, marginal and large-scale farmers have Android-based smartphones. In the survey of smartphones and farmers, utilization of Android-based versions was largely found, and a minimum number of iOS users was also available, mainly they are large-scale farmers. By seeing this issue, designed an application for farmers on Android-based supporting versions with a minimum API 5.0. The most featured thing is that farmers are well equipped with Android smartphones. To overcome these two issues, designed and developed an android-based mobile application for the utilization by every farmers.

3.17 Create a project

Android Studio makes it simple to develop Android apps for a variety of gadgets, including smartphones, tablets, TVs, and Wearable devices.

If do not have projects are open, Android Studio displays the Welcome screen, by begin with a new project by clicking **Start or Create a new Android Studio project**.

The Build New Project wizard by the appearing, allowing to select the sort of project that wish to create and load it with resource code and tools to get you started.

3.17.1 Choose your project

In the **Choose your project,** display that appears, by choosing the type of layout that want to design from categories of device, which are displayed as tabs nearer to the top of the wizard. For example, Figure 3.11 shows a layout with a basic Android Activity for a phone and tablet.

Android Studio might include example code and tools that help to get begin by choosing the sort of project to design.

The upcoming step is to setup some settings and create new project as illustrated in Figure 3.12.

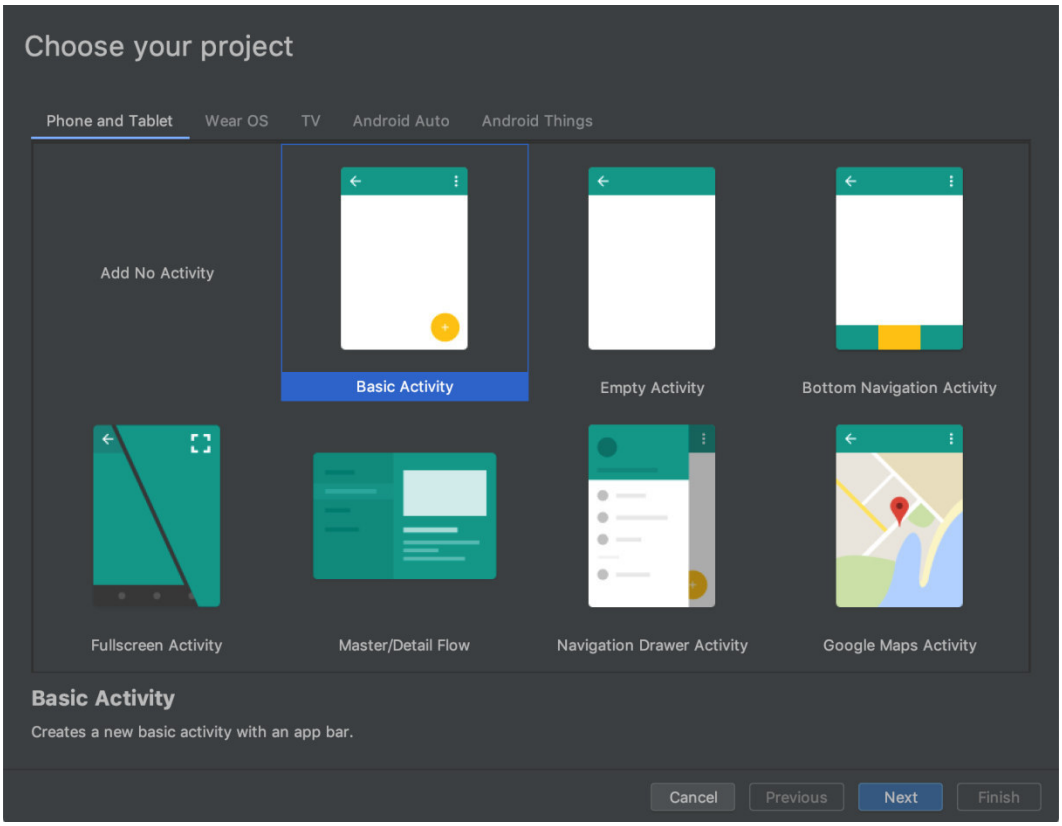


Fig 3.11: Screen of the wizard, the type of project you want to create

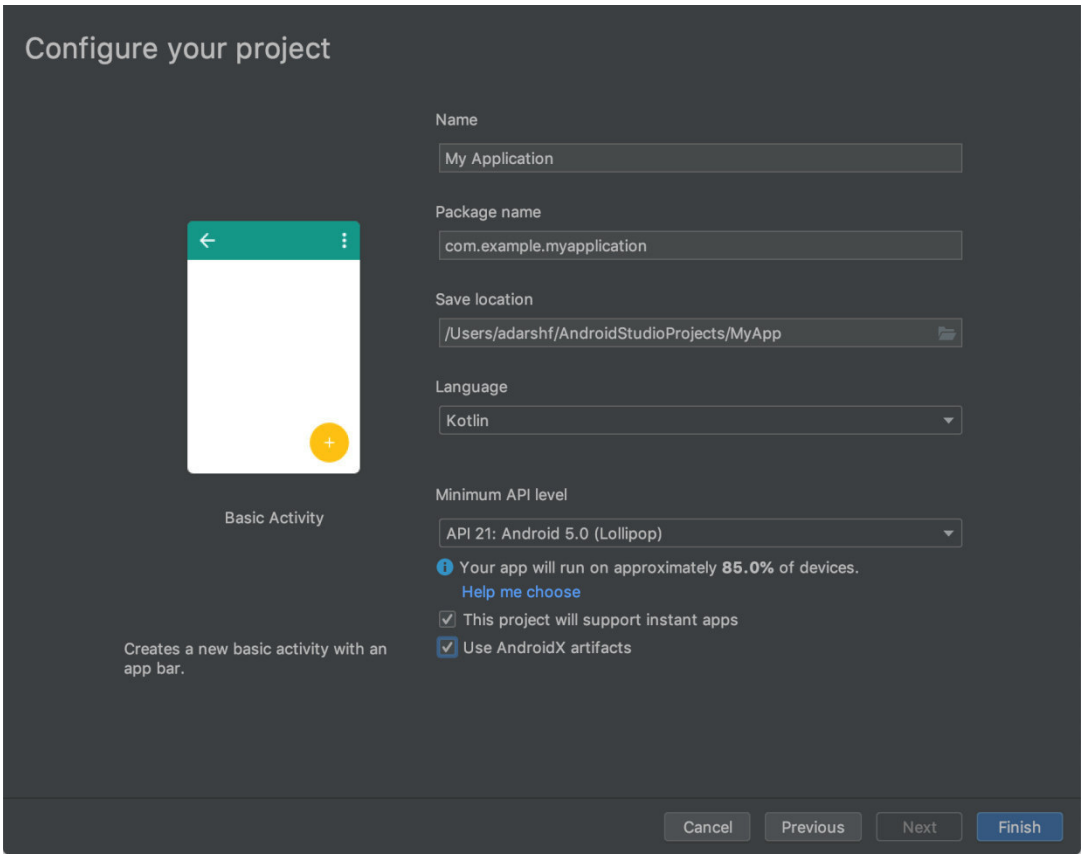


Fig 3.12: Configure your new project with a few settings

3.17.2 Configure your project

The upcoming step is to setup some settings and create new project as illustrated in Figure 3.12.

1. Provide a **name** for the project.
2. Enter the **Package name**. This project topic is also as application ID by default, that laterally modify.
3. Enter the **Save destination** where to save your work locally.
4. Choose the **programming language** that Android Studio should use for developing sample code for development task. It is not confined to solely using that only language to create the project.
5. Choose the **minimum API** level that app should support. Choose a lesser API level, that app will be able to use fewer contemporary Android APIs. However, your software will be able to execute on a larger percentage of Android smartphones.
6. Check a box next to **Use AndroidX artifacts**, if you want your program to use AndroidX libraries by default, which are enhanced replacements for the Android Support modules.
7. Once you are able to start working on your project, click **Finish.**

To get started, Android Studio creates a new application project with certain basic code and tools. Latterly, I f you start adding support for a various device form factor, you can do so by adding a module to existing project. You may also share logic codes and information between modules by developing an Android libraries.

3.18 C and C++ Programming Language

Add C / C++ code to the Android project by storing it in the application module's cpp directory. This program is compiled into the a native library that Gradle may bundle with applications whenever need to create the project. The Java Native Interface that allows your Java or Kotlin code to call operations in the native library (JNI).

If, on the other hand, to write native code to a current project, following points considered:

1. Develop new native source codes in Android Studio and include them to program.
2. You can neglect it by skipping it, even if have a coding language and want to incorporate a prebuilt native libraries. Configure Gradle by specifying the location of your CMake or ndk-build program code.

Gradle loads source code to the software Android Studio and packages of the native library (the SO file) into the app using the build code.

The following elements are required to compile and execute native code for your app:

1. The Android Native Development Kit (NDK): a toolkit that permits you utilise C and C++ coding with Android, as well as platform utilities for managing native activity and accessing hardware components like as sensors and touch interface.

2. CMake: An additional build tool which works in conjunction with Gradle to create native libraries. If you simply intend to utilise ndk-build, you do not require this element.

3. LLDB: The debugger used by Android Studio to troubleshoot native code.

3.19 JAVA Programming Language.

JAVA is a popular programming language that is mostly used for application development. Java was created in 1991 by a group of Sun Microsystems employees led by James Gosling. It was first known as 'GreenTalk,' then 'Oak,' and finally 'JAVA,' in 1995. Java is a high-level programming language that is also widely utilised in the design and development of desktop and web applications. Java is a platform-independent programming language that provides users with a safe, simple, and mobile interface. Java is based on Object Oriented Programming principles such as encapsulation, polymorphism, and inheritance

Encoding in Java is the process of enclosing methodological approaches in the same container. Encapsulation is accomplished by specifying the methodologies within the classes and packages and allocating them various levels of access

permissions. Polymorphism in Java is performed through method overloaded and override, with the goal of using that method for many purposes. The OOPs Inheritance concept establishes a parent-child relationship between various classes and allows for the accumulation of properties from one class to the other.

We can develop applications for the Android app using Java. Many module, interface, and abstract classes were accessible in Android, which is written in the JAVA programming language. Every action in the Android app is a java class that is a subclass of the application's Activity class.

3.20 Kotlin Programming Language

It is a statically typed accessible programming language maintained and created by JetBrains and open-source contributors. One of the really important advantages of Kotlin is its compatibility for multiplatform development. It saves time in creating and maintaining the same code for several systems while keeping the flexibility and conveniences of native coding.

3.20.1 Build Environment Requirement

This repository is using Gradle tool chains feature to select and auto-provision required JDKs from Adopt Open Jdk project.

Unfortunately Adopt Open Jdk project does not provide required JDK 1.6 and 1.7 images, so you could either download them manually and provide path to installation via JDK_16 and JDK_17 environment variables or use following SDK managers:

- Asdf-vm
- Jabba
- SDKMAN!

3.21 Database Designing

3.21.1 SQLite

There are numerous methods for storing permanent data in Android. SQLite is the database used to store app data. It is a very lightweight database which comes with the Android operating system. Integrating SQLite in Android is a time-consuming job

because it necessitates the writing of a large amount of code to maintain simple data. SQLite is an open - source platform SQL database that saves information to a device's text file. Android includes SQLite database implementation out of the box. SQLite supports all of the capabilities of a relational database.

SQLite is built into Android Studio. SQLite is a library that runs in the background and provides a self-contained, zero-configuration, server-less transactional SQL database server.

3.22 Project Development

Android Studio contains tools for each and every phase of development, however the most crucial is just writing your app: writing a program, designing layouts, making graphics, and staying productive. The tools that will assist in writing the app quickly.

3.22.1 Coding productivity

The features listed below are just a few that can help for becoming more productive while programming.

3.22.2 Code completion

Code completion reduces typing errors is the need to search up class, process, and variable names, which increases the speed application development. Basic completion, smart completion, and statement completion are all available in the coding language.

3.22.3 Create custom code-completion templates

By adding code snippets into live templates for quick insertion and execution of short sections of code. Type the pattern abbreviation and press the **Tab key** to add a live template. The code snippet connected with the template is inserted into your program by Android Studio.

3.22.4 Get quick fixes from lint

Lint is a code scanning tool provided by Android Studio that can help to navigate and repair issues with the quality attributes of the programming without launching the app or developing tests.

Fig. 3.13: Flow of crop cultivation practices

Lint checks the source files for potential problems and searches for optimization enhancements in accuracy, security, speed, usability, accessible, and internationalisation any time you create your application.

3.23 Working with resources

To assist in creating and managing resource files, Android Studio contains the following capabilities and resources.

3.23.1 Create images for all screen densities

Vector Asset Studio is a tool included with Android Studio that allows to produce images that support each display resolution. It can modify your own SVG file or choose one of several Google-provided graphical icons.

To get started, click **File > New > Vector Asset**.

3.23.2 Preview images and colors

When the reference images or symbols in the code, a representation of the system design in the left margin to assist you check the reference.

Click the thumbnail as in left margin to see the complete picture. Alternatively, position the cursor on the inline reference to the asset and click **F1** to display the details of the image, along with all available sizes.

3.24 Create new layouts

Android Studio includes a advanced layout editor which enables to drag and drop objects into the pattern while modifying the XML.

To get started, click on the module where you have to add the layout, then click **File > New > XML > Layout XML File**.

3.25 Translate UI strings

The Translations Editor utility provides a unified view of all of the converted resources, allowing to easily update or add interpretations, as well as detect missing transcripts, without having to open each version of a strings.xml file. Arrange translation services by uploading your strings data.

To get started, right-click on any copy of your strings.xml file then click **Open Translations Editor**.

3.26 Project component

The number of templates available in Android Studio is constantly rising. Figure 1 shows how Android Studio categorises templates depends on the nature of component they include, such as an Activity or an XML file.

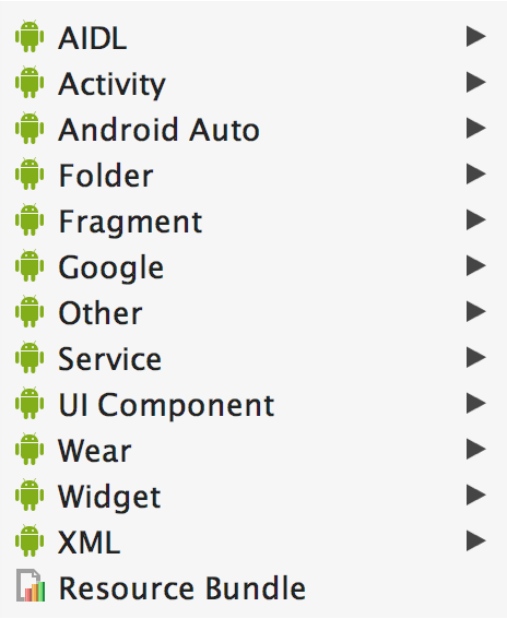


Fig. 3.14: The templates menu, accessible through the File > New

Use the Project screen to insert an Android project component using the template. Select New from the context menu when press right-click on the directory or folder to which islike to insert the new component or element. Getting a variety of template types similar to those displayed in figure 3.14 depends on what components or elements can be inserted to the folder you selected.

When users select a template or pattern to insert, a related wizard box displays and requests configuration information for the elements and component, such as its names. Android Studio produces and accesses the files and folders for your new module when you provide the configuration information. It also performs a Gradle build to synchronise your projects.

3.27 Activity Template

Adding additional activities to a preexisting app module is among the most typical uses of templates and themes. Insert an activities with the Login Activity

framework or template, for example, to build a login screen for the application's subscribers.

Android Studio also includes templates for several app module types, such as Wear OS, Android TV, and Cloud App Engine. When developing an app module, users can see templates for such various module categories. Templates for more API-specific modules and activity, such as Google AdMobs Ads and Google Maps, are also available.

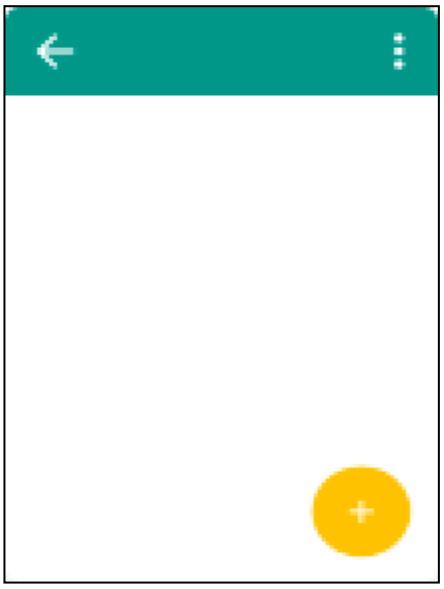
The code components for certain usage contexts, also including logging into an account, displaying a list of objects with attributes, or scrolling through a big block of text, are provided in the smartphone and tablet templates shown in below artifacts. Each can run as a full app module or as a single activity task.

3.27.1 Basic Activity

This template generates a basic app with an application bar and a floating action button. It serves as a jumping-off point for application by supplying regularly utilized UI components and elements.

This template includes:

- App Bar
- Floating Action Button
- Two layout design files: one to separate text content and one for the activity



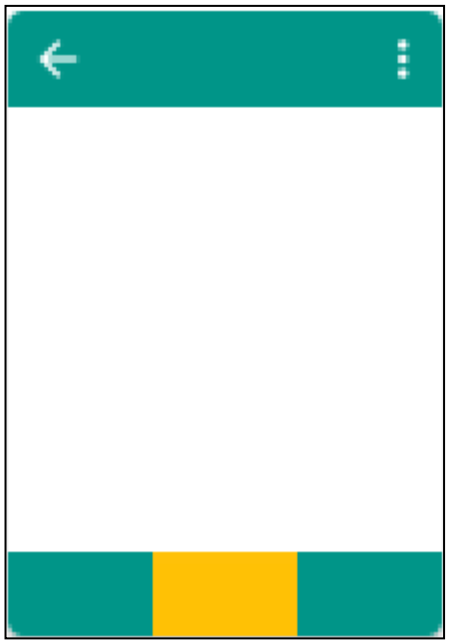
3.27.2 Bottom Navigation Activity

This design includes a conventional bottom navigation bar for an activity, allowing users to easily browse and change from top-level displays with a single tap. When your project has 3 to 5 top-level targets, use this template and style.

This template includes the following components:

• AppBar

• Sample layout with a single file in bottom navigation



3.27.3 Empty Activity

This template generates an empty activity as well as a basic layout file containing sample textual data. It enables you to develop application module or activity from the initial concept.

This template includes:

• A simple layout file with textual content.

3.27.4 Fullscreen Activity

This template builds an app that switches between a widescreen display and a view with typical user interface (UI) settings. The full screen mode is the default, while the regular view can be activated by tapping the gadget display.

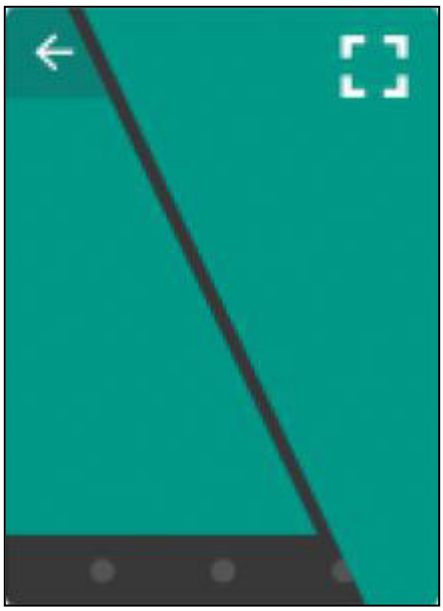
This template provides the following features:

• Implementation of touch listener for hiding the regular view elements.

• A button that displays in the regular view but accomplishes nothing.

• App Bar for the regular view.

• Basic layout file with both the full screen display and a frame layout pattern for regular view elements.



3.27.5 Login Activity

This template creates a regular login screen. The user interface contains username and password boxes, as well as a sign-in buttons. It is more typically used it as an activities template than as a template for an application module.

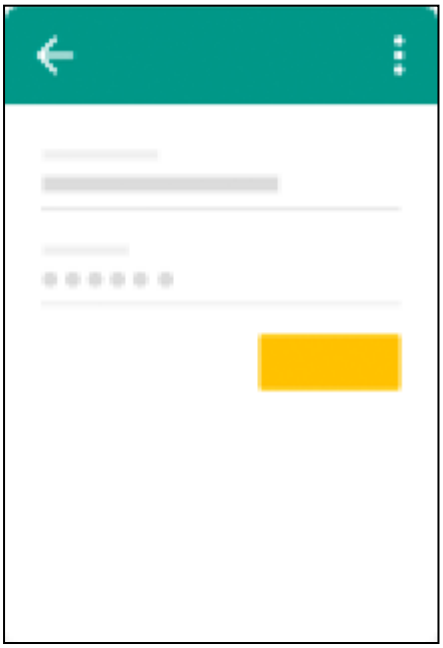
• AsyncTask implementation for managing network operations independently of the main user interface threads.

• Network operation performance indication

• Standard layout file also with recommended login UI

• Input fields for id and passcode

• Sign-in button



This template creates a regular login screen. The user interface contains username and password boxes, as well as a sign-in buttons. It is more typically used it as an activities template than as a template for an application module.

• AsyncTask implementation for managing network operations independently of the main user interface threads.

• Network operation performance indication

• Standard layout file also with recommended login UI

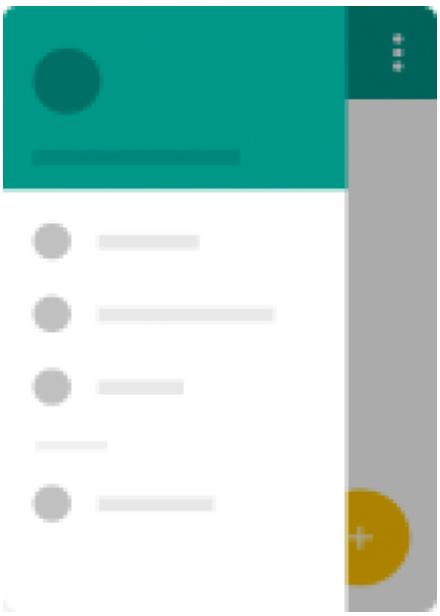
• Input fields for id and passcode

• Sign-in button

3.27.6 Primary/Detail Flow (Renamed and updated in 4.2 Canary 8)

- This template develops an app with an object list display as well as a display for the details of a specific element. When click on an selection of item listed, a screen with the item's detailed information appears. The structure of the two displays is influenced by the device on which the application is running.
- This template also includes API stubs for dealing with specific keyboard and mouse inputs, like as right-click actions on items listed and common keyboard shortcuts keys.
- A fragment encoding a list of things is included in this template.
- A fragment detailing the specifics of a single object.
- Each display has a floating Action Button.
- Toolbar for the item detail display that collapses.
- Substitutes of Service layout files for various device setups.
- The Context Click Listener on items listed to handle right-click actions.
- Detect keyboard shortcut keys on the object listing fragment using the Unhandled Key Event Listener.

3.27.7 Navigation Drawer Activity



This template builds a Basic Activity with the drawer navigation menu bar. The navigation bar displays in addition to the usual application bar and expands from the left edge or right side of application.

• Navigation drawer functionality with a Drawer Layout, related event handlers, and example menu selections are included in this template.

• AppBar.

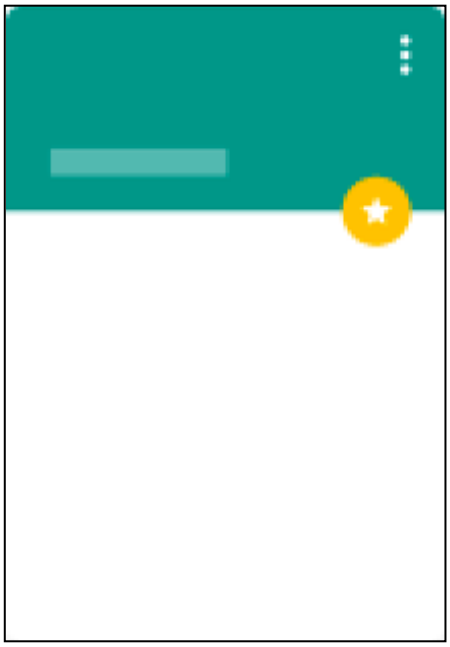
• Floating Action Button.

• In addition to the layout files from the **Basic Activity template**, there are structure files for the navigation drawer as well as the navigation drawer header

3.27.8 Scrolling Activity

This template builds an app that has a falling toolbar and a sliding view for extended textual information. The toolbar, which could also function as a header, gradually condenses as move down the page, and the floating action button disappears.

- A collapsing or falling toolbar in place of the standard AppBar.
- Floating Action Button
- Two layout files: separate the textual content into a Nested Scroll View and for activity.



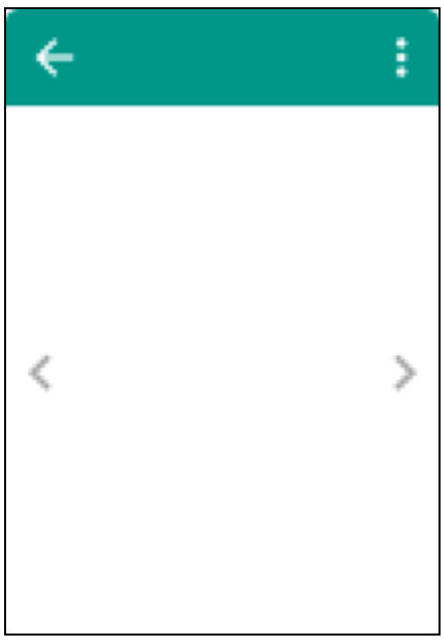
3.27.9 Settings Activity

This template builds an activities that displays a user's preferences or application settings. It extends the Preferences Activity class that is used as an activities template rather than an application module template.

This template has the following elements:

• Activity which expanded Preference Activities.

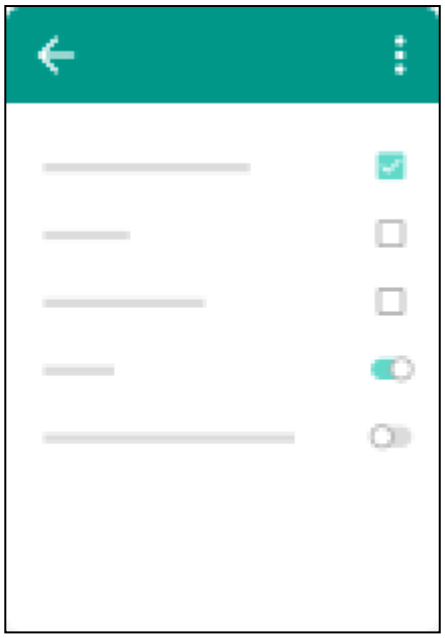
• XML files (in your project's res/xml/ directory) to define the presented settings



3.27.10 Tabbed Activity

This template builds a multi-section app with swipe navigation and an application bar. The sections are described as segments between which you can move by swiping left and right.

- Adapter that extends Fragment Pager that generates a fragment for every segment
- View Pager instance, a design manager for sliding between segments
- Two layout files are required: one for the activities and one for single fragments.



Java 8 language features and APIs

Android Gradle plugins 3.0.0 and later enable all Java 7 linguistic features, as well as a subset of Java 8 linguistic features that differ based on platform level. When developing your project with the Android Gradle plugins 4.0.0 or higher, able to access a variety of Java 8 programming language APIs without demanding a minimum API level.

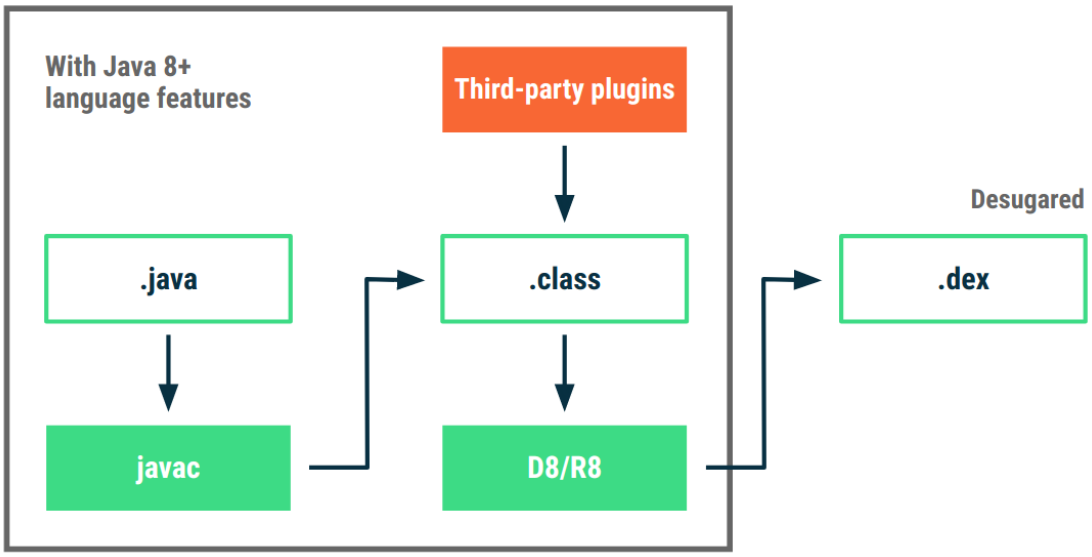


Fig 3.15: Java 8 language feature support using *desugar* bytecode transformations

The Android Gradle plugins has built-in support for various Java 8 linguistic features as well as third-party libraries that make use of them. As illustrated in Figure 1, the usual toolchain implements the new language functionality through bytecode modifications known as desugar, which are performed as part of the D8/R8 compilation of class files into dex coding.

3.29 App resources

Bitmaps and layouts are organised into type-specific directories within each module's res/ directory. It also include optimised copies of the each file for various device setups.

Depending on the sort of resource to add, Android Studio can assist by inserting a new and substitute resources in a variety of methods. This page explains how to insert basic resource files, where to find the resources, and how resource combining works.

3.30 UI with Layout Editor

The Layout Editor allows to create layouts fast by dragging UI components into a visual aesthetics editor. The design editor allows to view the layout on various Android gadgets and versions, and you can adaptively resize the design to ensure that it works well on various screen sizes.

The Layout Editor is notably useful for creating a structure with Constraint Layout, a layout designer that is compatible with Android 2.3 (API level 9) and above.

3.30.1 Introduction to the Layout Editor

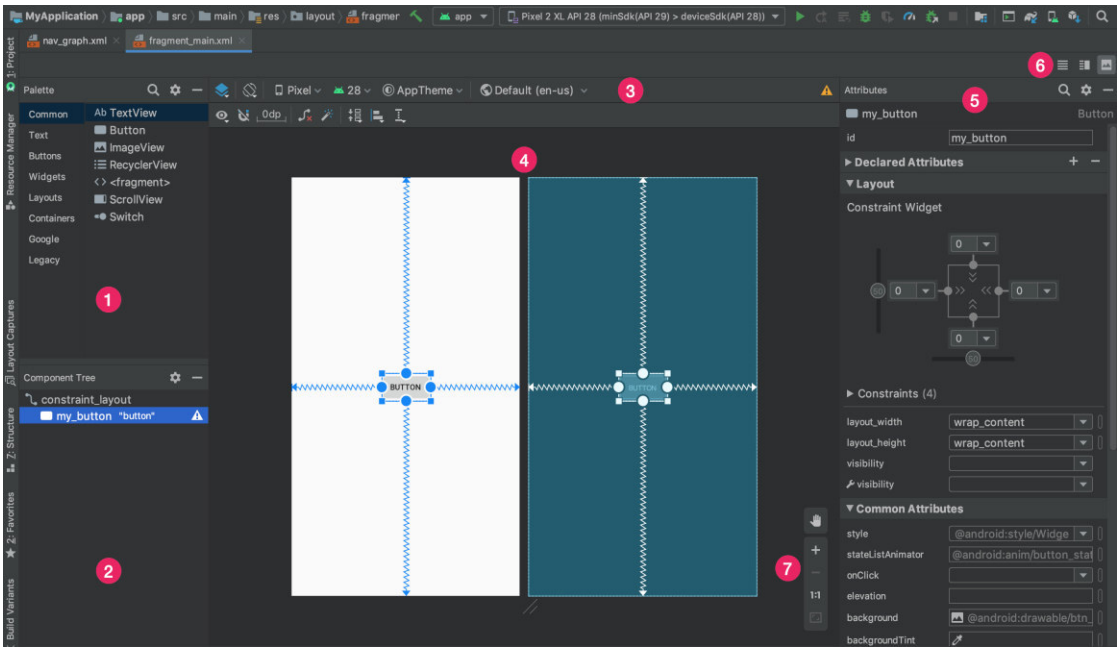


Fig 3.16: Layout Editor

The Layout Editor appears when you open an XML layout file.

1. **Palette**: Contains different displays and view groups that able to drag into your structural layout.
2. **Component Tree**: Structural Layout that shows the hierarchy of components.
3. **Toolbar**: Click these buttons to configure the layout appearance in the editor and change layout attributes.
4. **Design editor**: Edit both the layout in Design display and Blueprint view.
5. **Attributes**: Controls for the selected view's attributes.
6. **View mode**: Preview layout in either **Code**, **Design**, or **Split** modes. **Split** mode shows both the **Code** and **Design** windows at the same time.
7. **Zoom and pan controls**: Control the preview size and position within the editor.

3.30.2 Android App Links

Android App Links are HTTP URLs that take the users to specific context within the Android app. Android Applications Links can increase traffic to application, assist in determining which application content is more frequently used, and make it easily accessible for users to share and find content in installed app.

To insert support for Android App Links:

1. Develop intent controls in the manifest.
2. Include programming in the application's activities that handles incoming URLs.
3. Use Digital Asset Links to connect the mobile application and websites.

3.31 Build and Run App

With a few clicks, Android Studio creates new projects that can be deployed to the Android Emulator or an USB connection. After deployed the app, by utilizing Apply Updates to distribute specific code and resource updates without need to build a new APK.

Steps to build, compile and run app are as follows:

1. From the run setups drop-down menu in the toolbar, choose the app.
2. Select the gadget to execute the application from the target devices drop-down box.
3. Do not have gadgets configured, need to join a device via USB or build an AVD in order to use the Android Emulator.
4. Click **Run**

3.31.1 Debug Configuration/ Change the run

Android Studio utilizes a preset run configuration whenever application get started for the first time. The execute configuration describes whether the programme get deployed from an APK or an Android App Bundle, the program to run, the packages to deploy, the activities to start, the target device, emulator configurations, logcat parameters, and other details.

The default run/debug setup creates an APK, opens the default project activity, and selects the targeted device using the Select Deployment Target dialogue. If the default settings do not fit your application or module, it can edit or build, or new program run/debug configuration at the application, default, and module levels.

3.31.2 Architecture of Mobile app Data

This app will provide the details of crop as like seed rate, variety, spacing etc. From the basics of crop details upto the end of value addition it may provide the package of operation. In this crop included of Maharashtra. On that particular crop the operation should be taken that it may guide this mobile app.

This may also guide from State to district. In that district wise crop taken and what operation should we have to perform for increasing in productivity and better outcome of crop by forwarded to value addition supporting with a regional language.

3.32 Work flow diagram of a proposed system

A work flow diagram or work tasks that specify and illustrate a workflow or process of how the actions would be flow across resources, whether they are humans or machines to continue, and what conditions permit the order to occur. In the system, the user has the option of different crops which undertake operations required for the crops to be undertaken. This operation have different machinery required for on field operation of crops. In that it have different machinery with the specification and availability of it. The machinery are displayed in the view sample activity.

3.32 Software maintenance and enhancement

3.32.1 Software maintenance

The system's maintenance is an important part of SDLC. They are divided into three categories.

• **Corrective maintenance**:

It is the reactive change of a software package took delivery of corrected identified issues.

• **Adaptive maintenance**:

After-delivery customization of a software system to maintain the software product functional in an altered or changing environment.

• **Perfective maintenance**:

After-delivery improvement of a software product to increase performance or manageability.

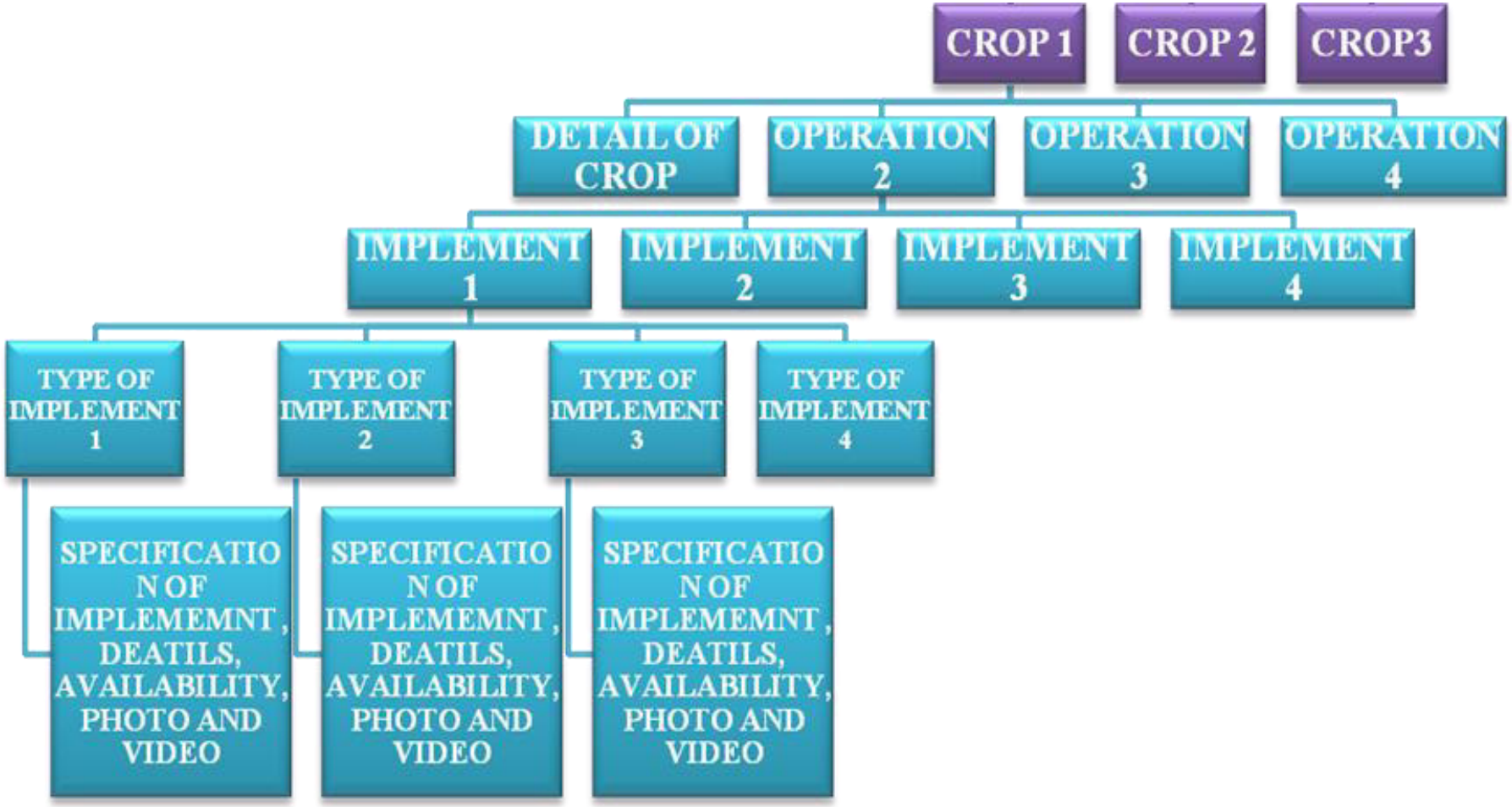


Figure 3.17: Hierarchy of outline of mobile app

3.32.2 Software enhancement

Any product change or modified that expands software or hardware functionality beyond the initial customer specification is referred to as an improvement. Refinements enable scalability of software and hardware quality performance.

The requirements for improvement could be linked to:

1. Product Efficiency
2. Usability Engineering
3. Compatibility
4. Minor enhancement with the addition of a new function
5. Significant enhancement with multiple features
6. Switch to a new platform

3.33 Hardware and software requirements

The various hardware and software requirements of "Design and Development of Android based Mobile Application for Farm Mechanization" applications are described below:

3.33.1 Hardware requirements:

a. Intel Corei3-2370Mprocessor
b. RAM4GBminimum
c. 64-bit Operating System
d. Android phone with5.1 version or above
e. 1280X800 minimum screen resolution

3.33.2 Software requirements:

a. Windows2007 or Windows XP
b. JAVA with JDK 1.8.0 (Java Development Kit)
c. JRE 1.8.0_73
d. Android Studio with5.1 version (lollipop)/ Android.

3.34 Steps to use the mlcc (farm_mechanization) application

The proposed application has two modules:

- For the user
- For the admin

3.34.1 For the user:

- User first needs to download the application for better service.
- User need not to be have internet just only need to be have internet for download.
- User must have the internet connectivity on his device to download.
- After then user needs to select the crop.
- After clicking, the crop they show list of operation undertaken in the crop.
- Then, operation have specified machinery required for the operation.
- A button is displayed on that screen which having availability of machinery.
- The samples are also available in the menu of the application for the user.

3.34.2 For the admin:

- Admin has access to the contents of the users who have registered for his application.
- The administrator can enhance the samples to strengthen the precision of the findings.

Above procedure is used for designing and development of android based mobile application.

CHAPTER -IV

RESULTS AND DISCUSSION

CHAPTER-IV

RESULTS AND DISCUSSION

A dynamic android based mobile app named as 'FARM MECHANIZATION' has been developed using the methodology and technologies given in previous chapter. The FARM MECHANIZATION app i.e. Mechanization based app for agricultural activities in the farm by use of mechanized machinery by Manually, Animal, Tractor and Self Propelled operated. App has been developed based on requirement of need based work, materials and methods discussed in the previous chapter followed with analysis.

'FARM MECHANIZATION' is one of the android based application which support users from making their decisions and ease of work in which they are supposed to do and farmer think to do farming in mechanized way.

4.1 Features of the Application

The 'FARM MECHANIZATION' app has been successfully developed with formulated machinery for different operations and tested with the compatible version for the smartphone with the farmers. This app contains mainly following features such as-

- Agricultural activities and farm machinery need the practice with the crop.
- Provision of the information related to available farm equipment's/tools and machineries.
- App has machinery with their specification with different range of operation according to their size, operational width, required for different operation/ crop with respect to power.
- Availability of machinery locally at your city or near-by city.

4.2 User Interface Representation

To make the project more interactive, many controls were used and developed with the layout file. The following are the key controls that have been designed and implemented in this application.

- **Text View**: The textual view component is an element of the GUI's view group. It shows the user the text or content overview of any activity and enables them to modify it.

- **Edit Text**: This makes the text in the text itself is editable.
- **Button**: One of the most critical components required by the application. It is primarily linked with the action that occurs when the user clicks it. We can symbolize the button with any text that has the action class attached to it.
- **Image Button**: Imagine we have to include a picture for the button we built. We can do so by inserting the origin or location of the jpeg image file within the tags in the layout display file.
- **List View**: This is a critical component of the view group that displays information about everything when we click the action button. It also enables us to navigate through the display and examine the information shown. The content is retrieved from the database using the list adapter.

4.3 User Interface of the proposed application

The GUI is quite well designed even though it is the gateway to user interaction. It is quite basic to direct the user to the needed information even without difficulty or trouble.

4.4 Splash screen

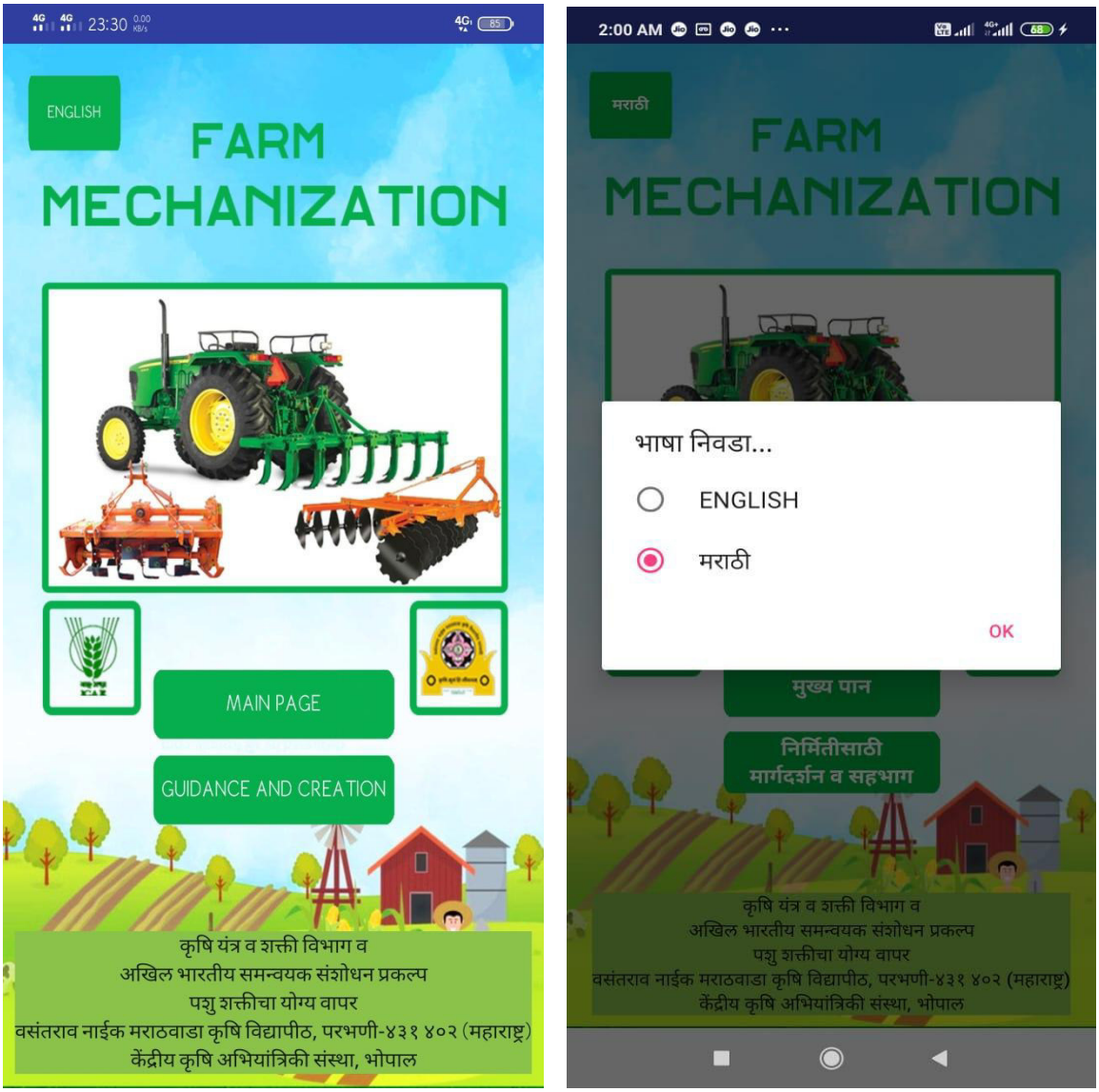
Screenshot 4.1 shows the splash screen of "FARM MECHANIZATION" app which includes images regarding content of the app. This screen display for 10 seconds after opening of the app.

4.5 Home Screen and Selection of Activity

Home screen contains the logo of ICAR and VNMKV in bottom of activity with the Main page and Guidance and Creation field. In the bottom of Main screen contain Textual part of college name associated with mobile app.

The two fields on main home screen was Main Page and Guidance and Creation, whereas Guidance and creation field are available both the languages viz., English and Marathi Language shown in Screenshot 4.2.

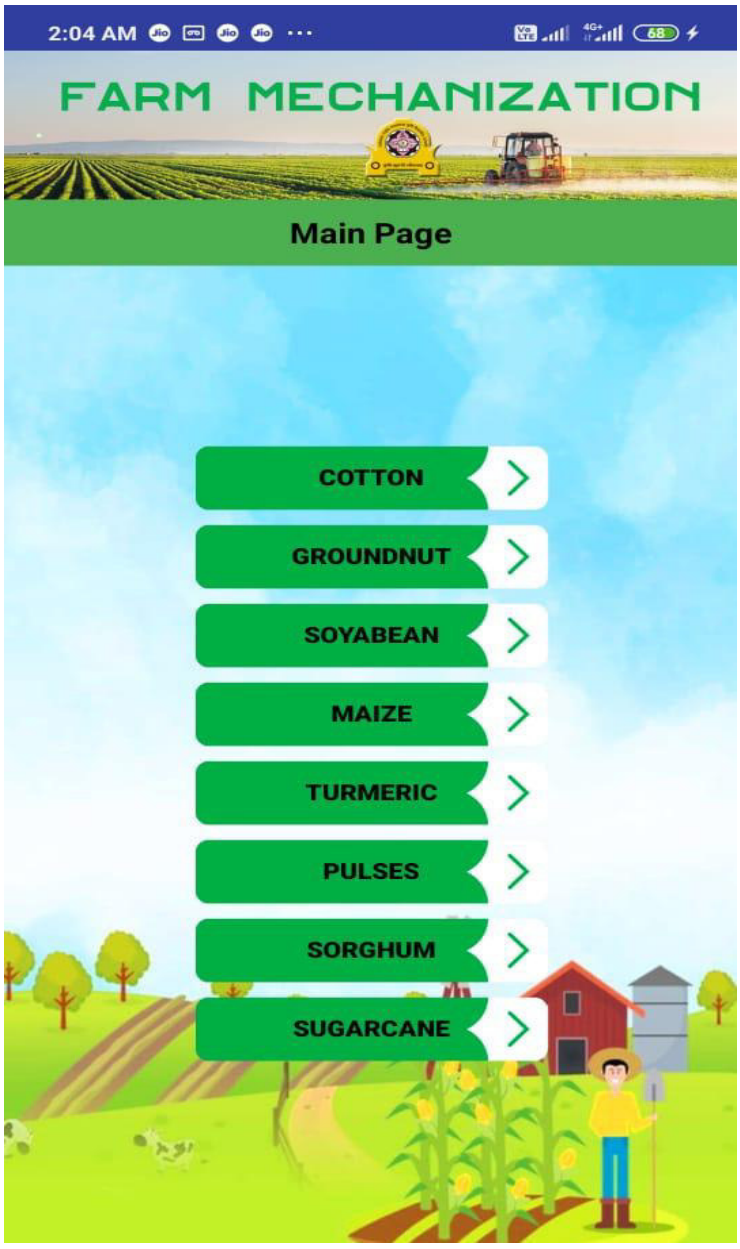
Screenshot 4.1: Splash Screen



Screenshot 4.2: Home-screen and Activity screen

4.6 Crop database

This app is totally depends upon the crop database file. In main page of the app , there are eight fields of major crops database which is in the form of ordered list. Each crop field contain list of operations to be practices i.e mechanization of recommended cultivation practices from land preparation to crop residue management. Four mode of farm mechanization i.e Manual, Animal, Tractor, or self propelled are included in each crop fields Screenshot 4.3 describes the ordered list of crop database.



Screenshot 4.3: Ordered list of Database

4.5 Crop Cultivation Activities list

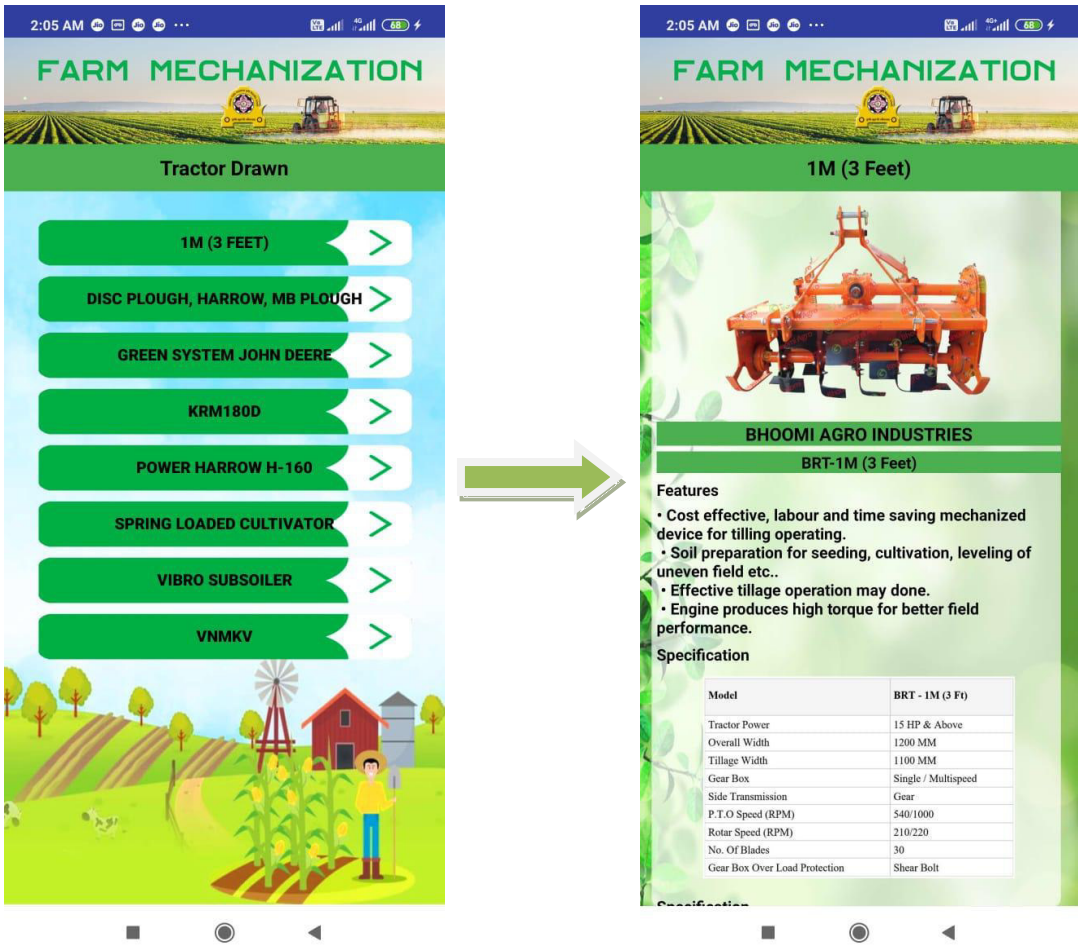
Crop Cultivation activities which required various machineries for the various operations is listed in each crop field. These operations to be carried out over a period of crop growth. Different operations may carried out with the help of different machineries operated by different modes for operations during growing period of crop.

This contains different cultivation practices from Land Preparation to crop Residue Management i.e. sowing, harvesting, weeding, threshing, land preparation, fertilizer application, transplanting, hoeing and other agricultural and allied activities in Screenshot 4.4. Farm Operation wise mechanization is included under each crop field.

Screenshot 4.4: Crop cultivation activities

4.6 Database about operation wise Equipments & Implements

Database about operation wise the app is collection of various type of equipments and Implements developed and recommended by ICAR institutes. Which are made available in market by Private company, and local fabricators. Mainly Equipments and Implements are mostly gender friendly.



Screenshot 4.5: Database of farm equipment

Content in each field of selected Machinery & Equipment. These database contain various information related to equipments, Implements or machines which is shown in Screenshot 4.5 & 4.6 such as-

- Suitable crop
- Capacity of equipments

- Developed at
- Specification of the equipment
- Features and compatibility with tractor capacity
- Source of availability
- Availability of video link to understand well equipped

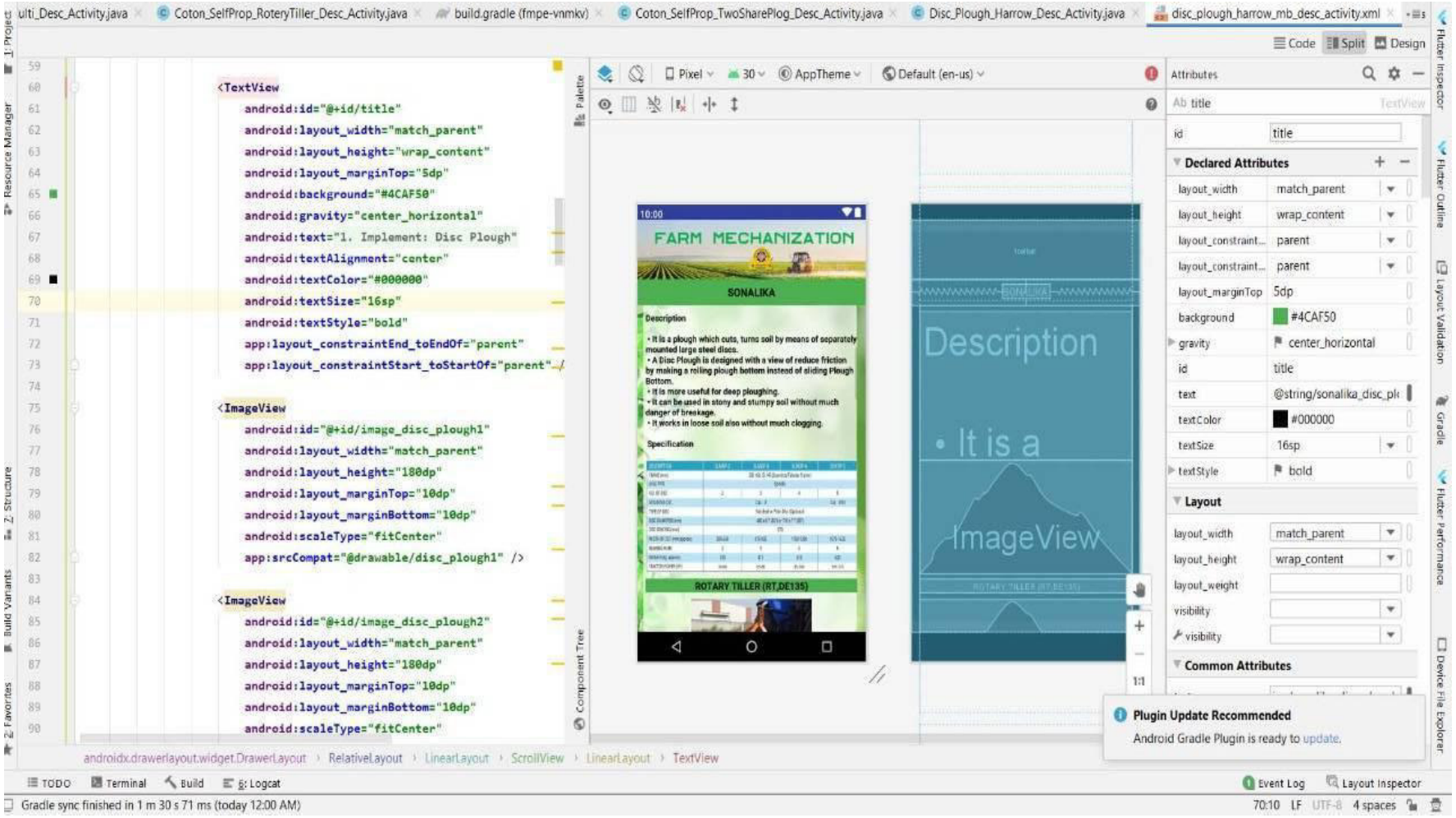
4.8 Dealer Information

On this field, (Screenshot 4.6) list of dealers, and availability of machinery at near by city. Field of availability or address is in between selection of machine and mode of work. Similar types of field are created and provided crop wise in four mode of operation.

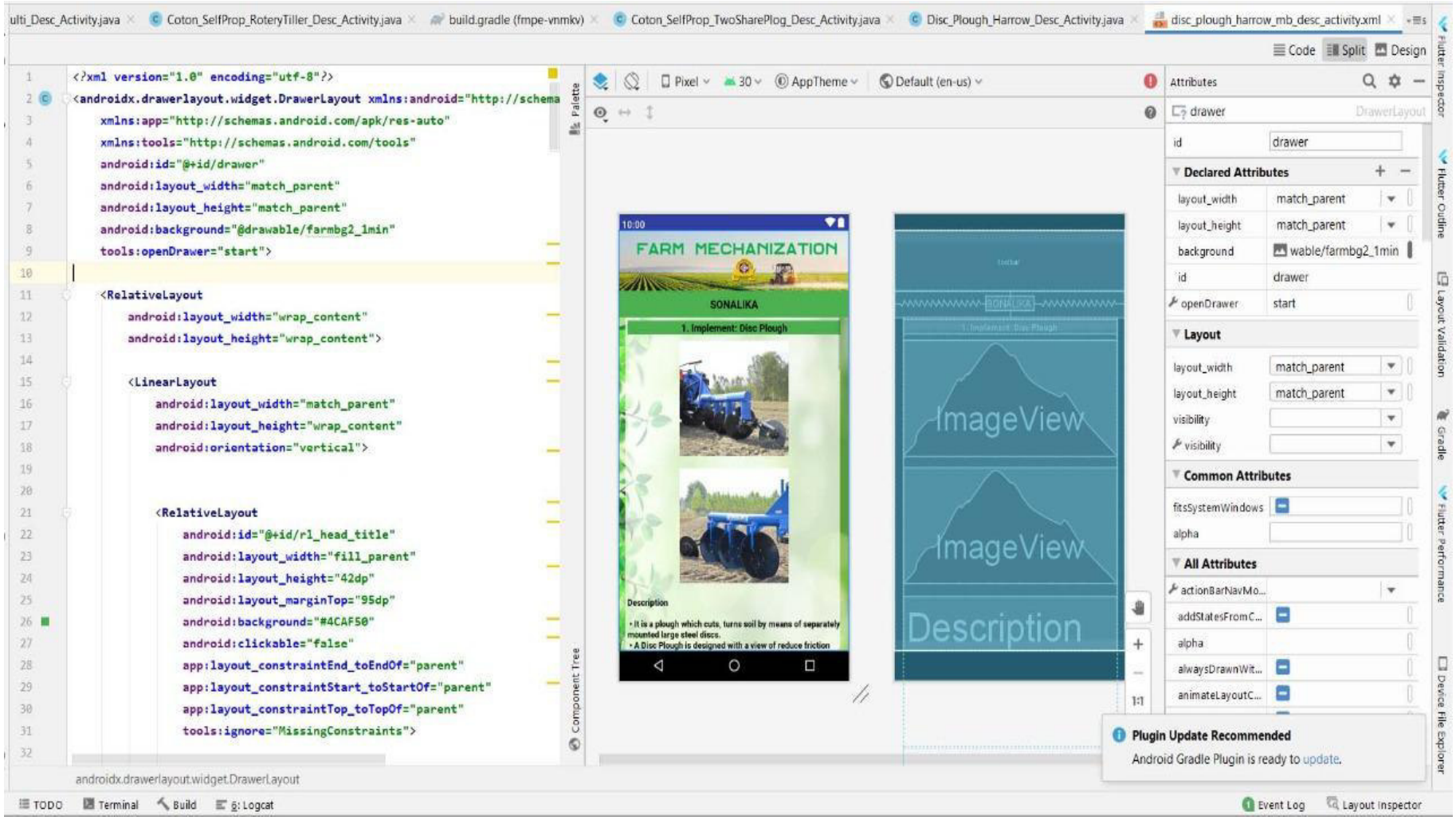
Screenshot 4.6: Availability of machine

```java
import android.app.AlertDialog;
import android.content.Context;
import android.content.res.Resources;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;

import com.github.chrisbanes.photoview.PhotoView;
import com.google.android.material.navigation.NavigationView;
import com.vnmkv.womeninagri.LocaleHelper;
import com.vnmkv.womeninagri.R;

import androidx.annotation.Nullable;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.widget.Toolbar;
import androidx.core.graphics.drawable.RoundedBitmapDrawable;
import androidx.core.graphics.drawable.RoundedBitmapDrawableFactory;
import androidx.drawerlayout.widget.DrawerLayout;

public class Disc_Plough_Harrow_Desc_Activity extends BaseActivity {

    NavigationView nav;
    ActionBarDrawerToggle toggle;
    DrawerLayout drawerLayout;
    Toolbar toolbar;
    String lang;
    Context context;
    Resources resources;
    TextView txttitle,txtsubtitle,txtsahitya, txtsahityadetails,txtkruti,txtkrutidetails,txtposhan,txtposhandetails,txtupyog,txtupyogdetails;
```

Screenshot 4.7: Code for testing



Screenshot 4.8: (i) Griddle view with layout editor



Screenshot 4.8: (ii) Griddle view with layout coding

4.9 Testing of App

Developed app was tested in Emulator which is compatible to 5.1 Lollipop android version. The app required mobile having 8 GB RAM. Before finalization of the app, the app was tested on various configurated smartphone with different versions.

4.10 Description of Application Features and the Approach

The proposed work is a designing of software application based on android platform. This application allows quicker, user friendly and a positive user experience. The user can know about the required machinery for the crop of a operation. This application is developed in android studio software tool. This equipment/ machinery is available at nearest authorized dealers of the company. Also this application can be further deployed in precision agriculture equipments for practical application in availability of machines.

4.10.1 Start of App

If the users want to use the App, must download the application and use it. It does not have any registration or login/ Signup process. It is easy to use, understandable and compatible to all android smartphones. After opening of app splash screen displayed for 10 second and main page to see full functionality of the application.

Screenshot 4.9: Splash and Home Screen

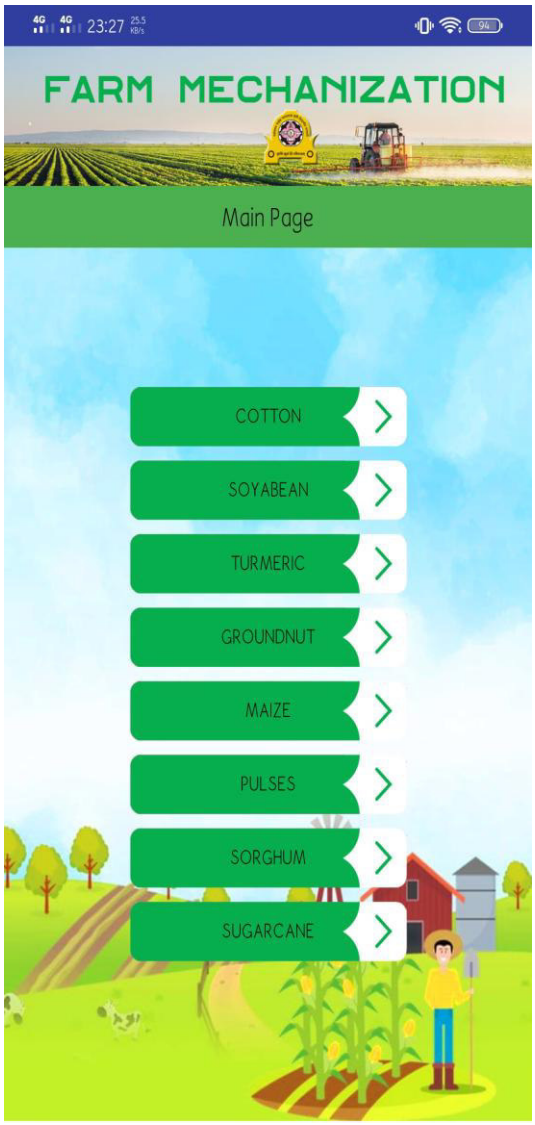
4.10.2. Guidance and Creation

Guidance and Creation field of Home-screen contain the team which support for collection and compilation of required information for designing the app. The app is formulated by coding of C, C++ and JAVA Language. Android Studio is used as developing tool.

Screenshot 4.10: Guidance and creation page in Marathi & English format

4.10.3 Crops

This is the proposed mobile application main page; these pages consist of eight categories and sub categories. After processing successfully to the main interface of the crop listed is appeared which is instantly visible without scrolling. This could help the users in saving time by instant visible for selection activity. This models is created by using Web Services which display the information or list to the user. Screenshot 4.11 shows the action as follows:



Screenshot 4.11: Main page contain list of crop

4.10.4 Operations

These pages consist of different categories and sub categories. After preceding to the next page by click on crop. The main interface of the operations list is appeared. Crops have different operation for different crop from seed bed preparation to harvesting or threshing, this operation easily visible without scrolling up or down. This could help the users in saving time by instant visible for selection of activity. After selecting any of operation undergone through it, shows the required machinery for particular operation of crops in the four modes of operations. This models is created by using Web Services which display the information or list to the operations. Screenshot 4.12 shows the action as follows:

Screenshot 4.12: Cotton under operations

4.10.5 Mode of Work/ Drawn

The selected machine may work with different mode. The modes of work are categorized in four ways:

1. Manual Drawn
2. Animal Drawn
3. Tractor Drawn
4. Self Propelled

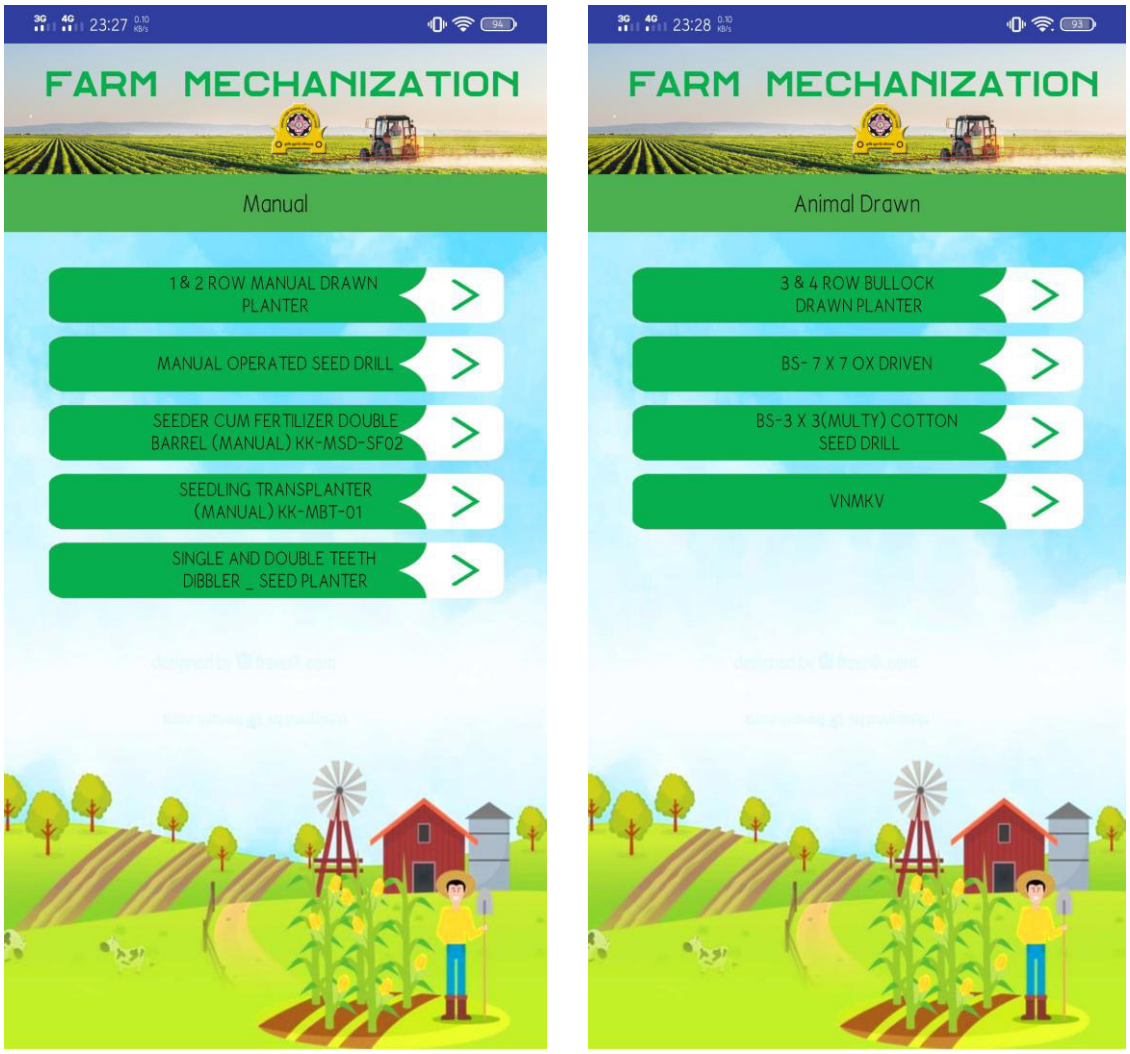
Every mode of work have different range of machinery for different operations. This operation carried out in the farm and demonstrated. All the mode of work, use of mechanized machinery for reducing mechanization gap in every operations.

Screenshot 4.13: Mode of Work or source of work under sowing

4.10.6 Machines/ Implements

The machines section of different mode of work has been categorized into different machineries for different operation. This contain actual information database of app is Short Description of Machine/ Equipment, Specification and Image and Video. This is available in wide range with different sizes of work with required torque and power.

This most important section contains the equipments/ machineries are available at nearest authorized dealers of the company. This also contains contact number of authorized dealer for contacting them.



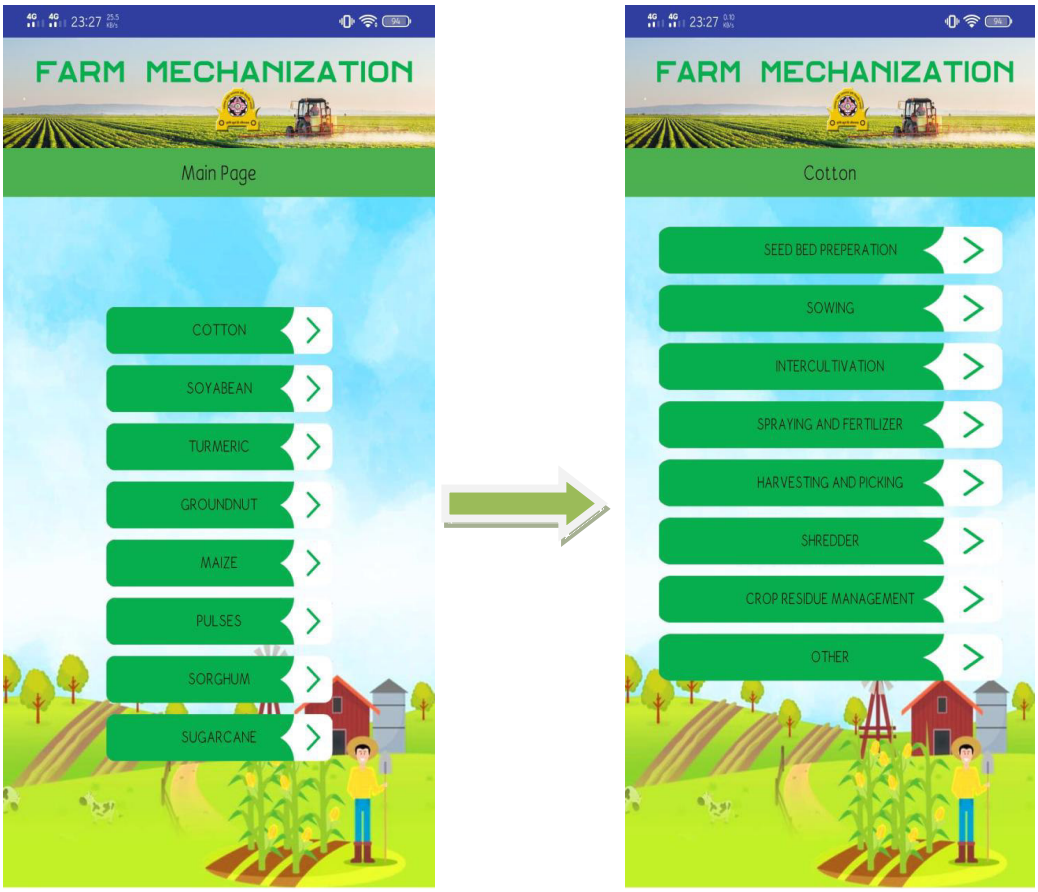
Screenshot 4.14: List of machines under Manual and Bullock drawn mode of work

Screenshot 4.15: List of machines under Tractor and Self propelled mode of work

4.11 Proposed flow of Application

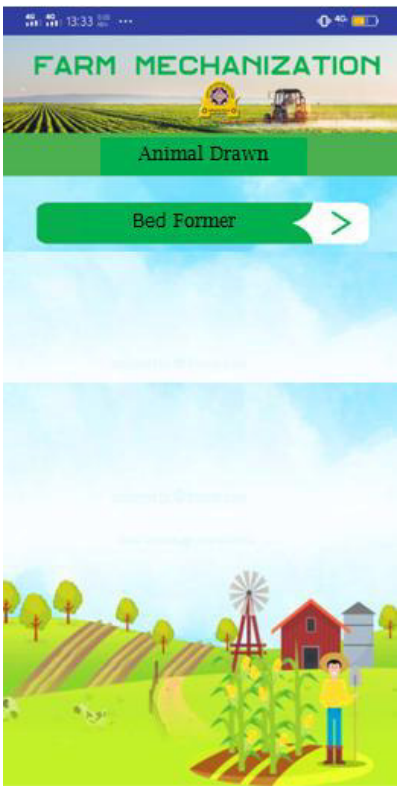
The screenshot 4.16 shows the proposed flow of cotton from the start of application with the selection of crop i.e cotton. It shows cultivation practices of cotton viz. Seed bed preparation to Crop residue management. Similarly the cultivation practices of another crops are shown in the application.

Screenshot 4.16: Proposed flow of cotton shown from the start of app to the end



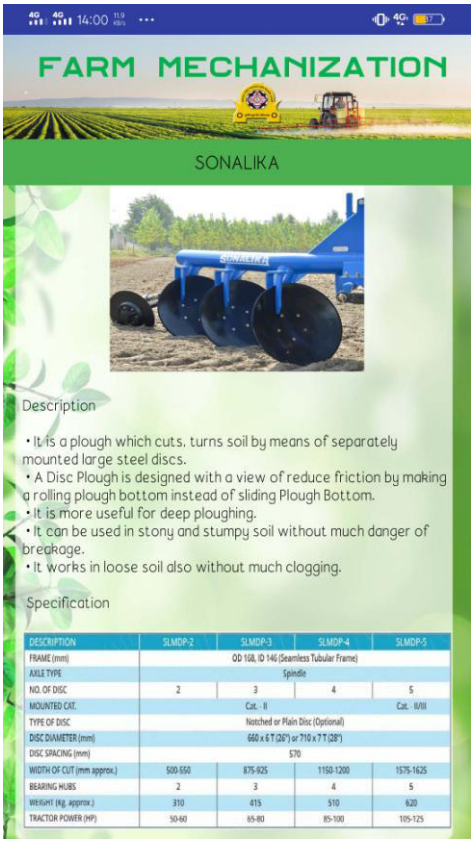
In cotton crop field, after clicking on seed bed preparation field, following screen is displayed.

In field of seed bed preparation, by clicking on Animal drawn following screen is displayed.



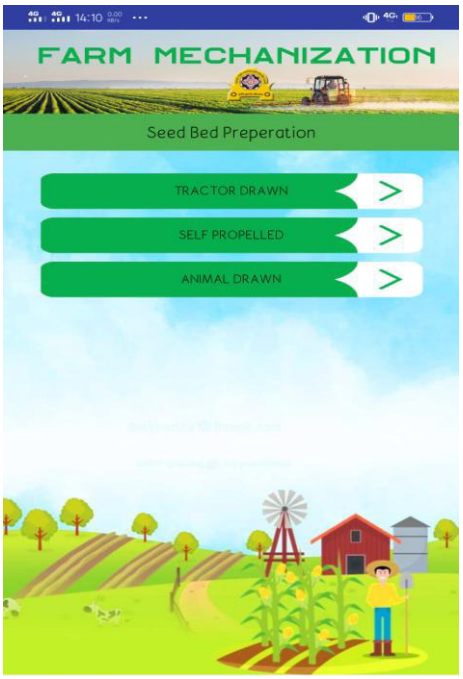
Part	Specification	Dimension
Main frame	Square Bar(mm)	50*50
	length(mm)	1850
	width(mm)	550
Supporting wheel	diameter(mm)	300
Ridger	height(mm)	600
	width(mm)	450
	Max. spacing(mm)	1750
	Min. spacing(mm)	630
Foot stand	Width(mm)	500
	length(mm)	540
Handle	height(mm)	740
	width(mm)	60

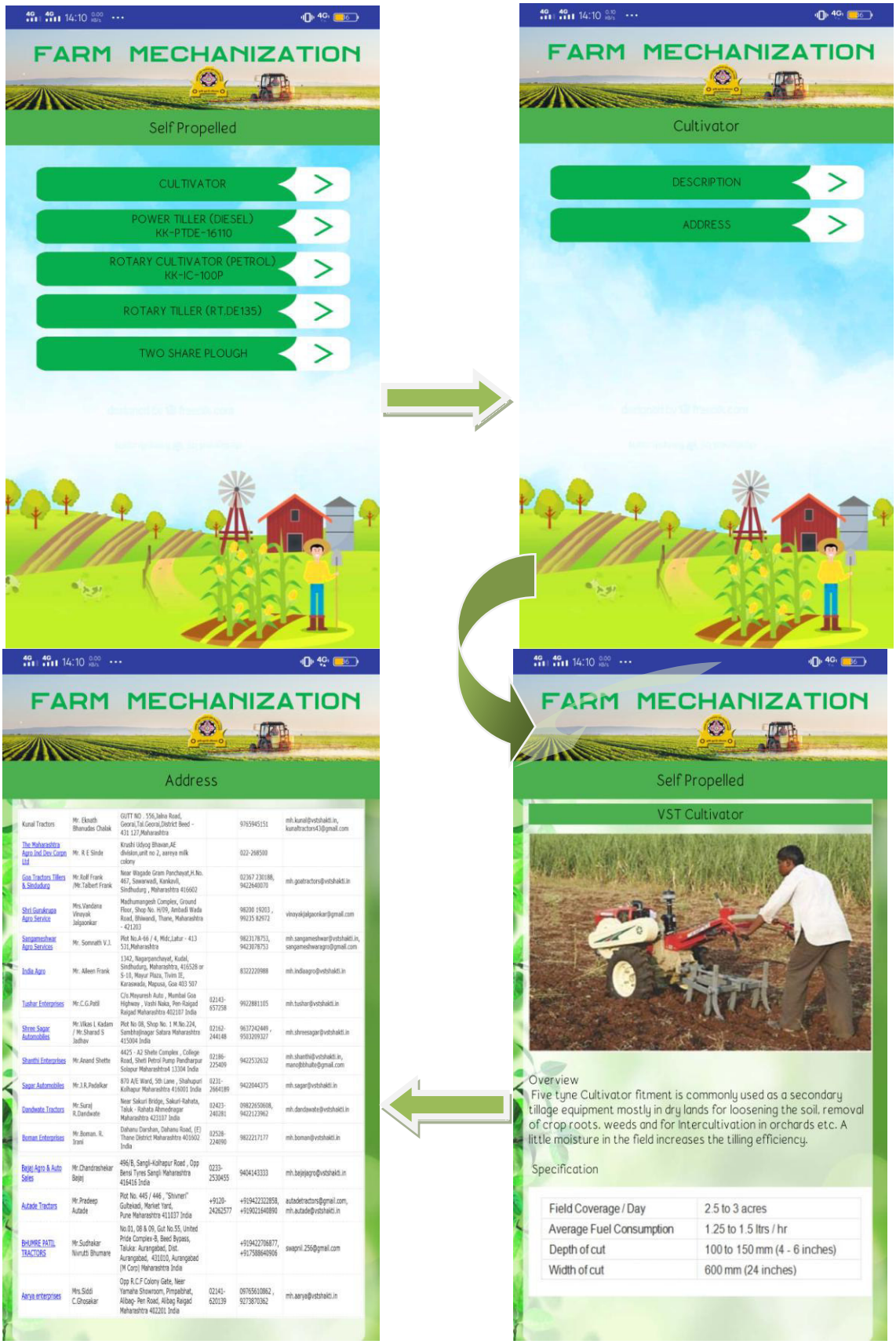
In field of seed bed preparation, by clicking on Tractor drawn following screen is displayed.



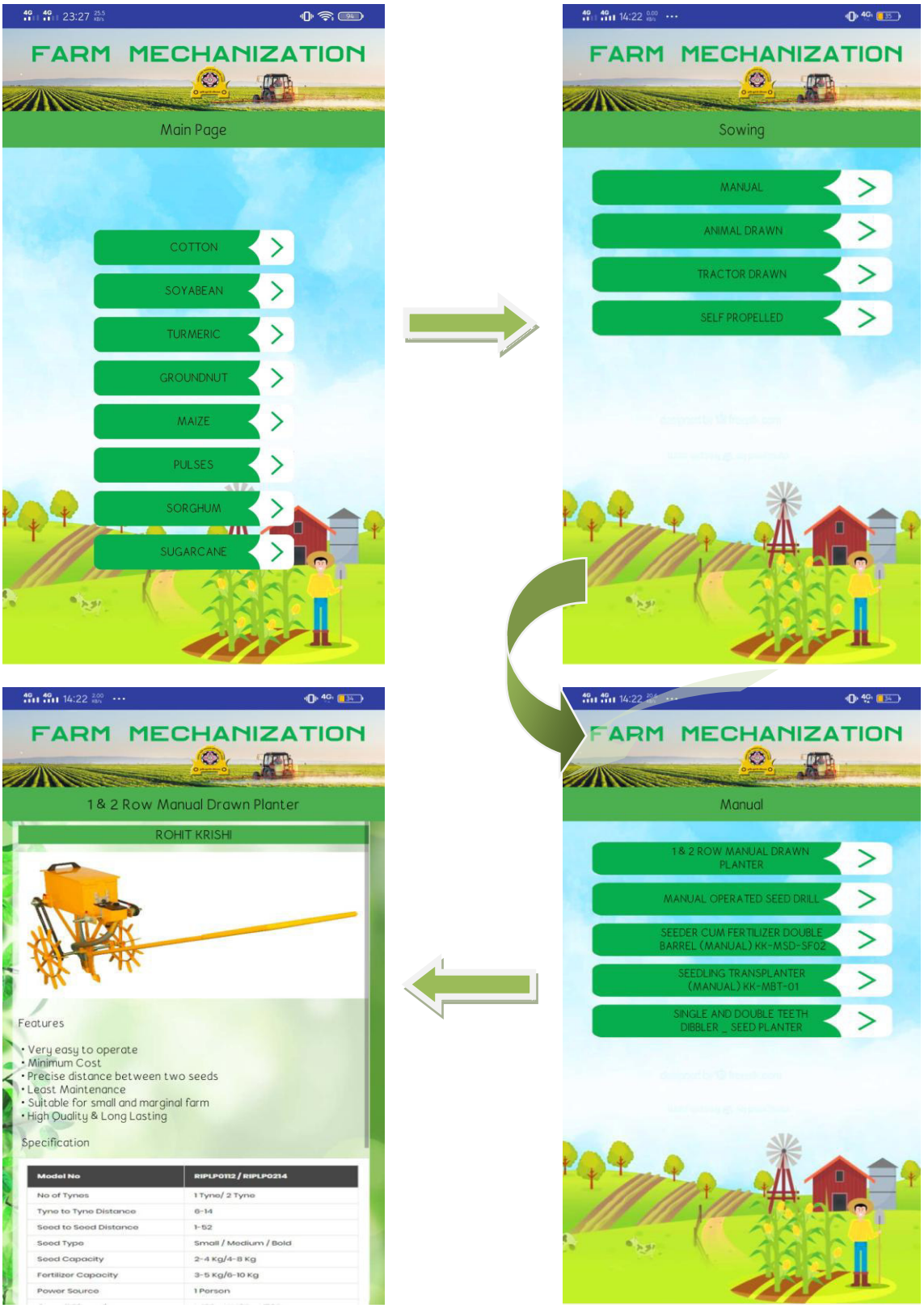
DESCRIPTION	SLMDP-2	SLMDP-3	SLMDP-4	SLMDP-5
FRAME (mm)	OD 168, ID 146 (Seamless Tubular Frame)			
AXLE TYPE	Spindle			
NO. OF DISC	2	3	4	5
MOUNTED CAT.	Cat. - II			Cat. - II/III
TYPE OF DISC	Notched or Plain Disc (Optional)			
DISC DIAMETER (mm)	660 x 6 T (26") or 710 x 7 T (28")			
DISC SPACING (mm)	570			
WIDTH OF CUT (mm approx.)	500-550	875-925	1150-1200	1575-1625
BEARING HUBS	2	3	4	5
WEIGHT (Kg. approx.)	310	415	510	620
TRACTOR POWER (HP)	50-60	65-80	85-100	105-125

In field of seed bed preparation, by clicking on Self propelled following screen is displayed.

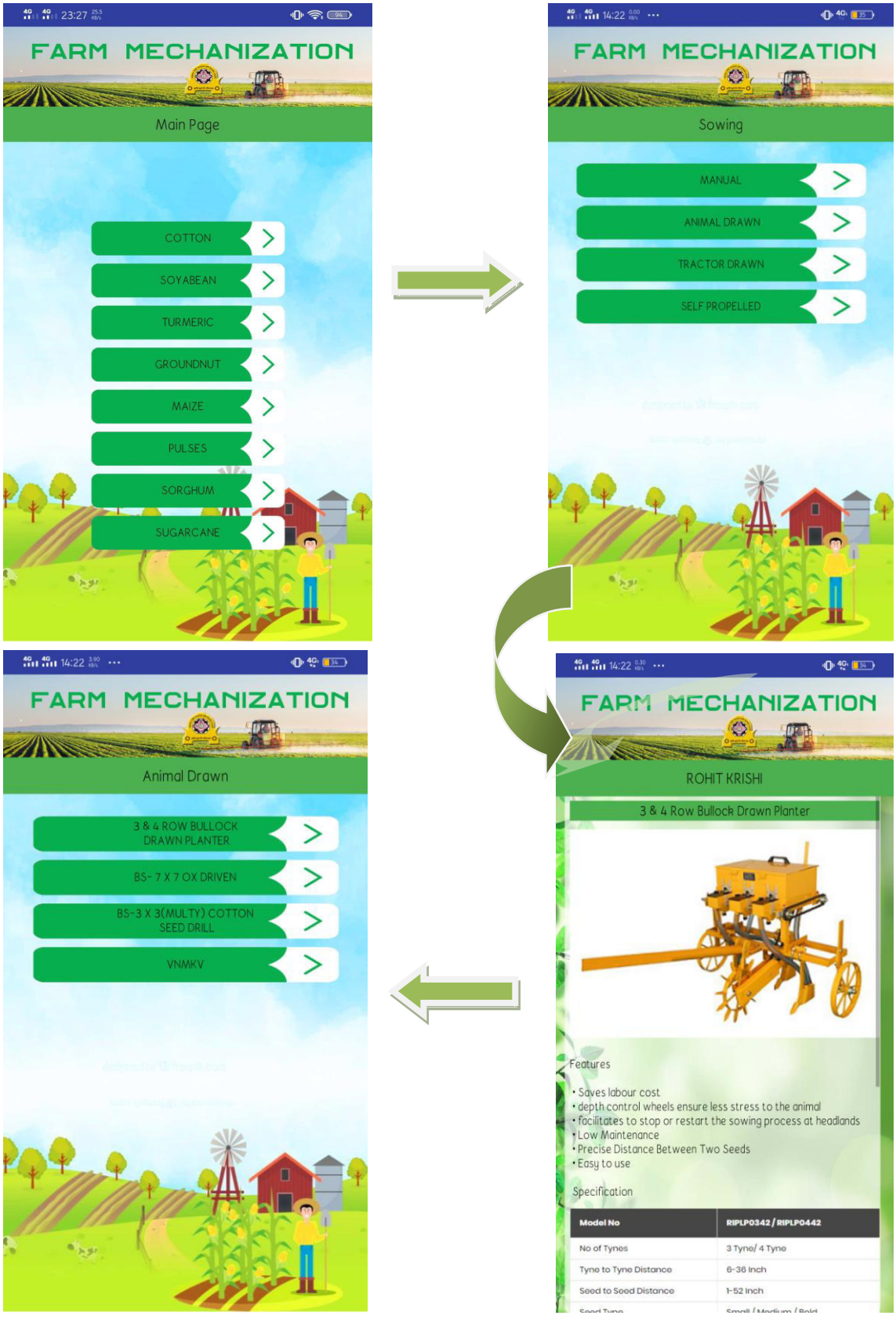



FARM MECHANIZATION
Self Propelled
CULTIVATOR
POWER TILLER (DIESEL)
KK-PTDE-16110
ROTARY CULTIVATOR (PETROL)
KK-IC-100P
ROTARY TILLER (RT.DE135)
TWO SHARE PLOUGH
FARM MECHANIZATION
Cultivator
DESCRIPTION
ADDRESS
FARM MECHANIZATION
Address
FARM MECHANIZATION
Self Propelled
VST Cultivator
Overview
Five tyne Cultivator fitment is commonly used as a secondary tillage equipment mostly in dry lands for loosening the soil, removal of crop roots, weeds and for Intercultivation in orchards etc. A little moisture in the field increases the tilling efficiency.
Specification
Field Coverage / Day
2.5 to 3 acres
Average Fuel Consumption
1.25 to 1.5 ltrs / hr
Depth of cut
100 to 150 mm (4 - 6 inches)
Width of cut
600 mm (24 inches)

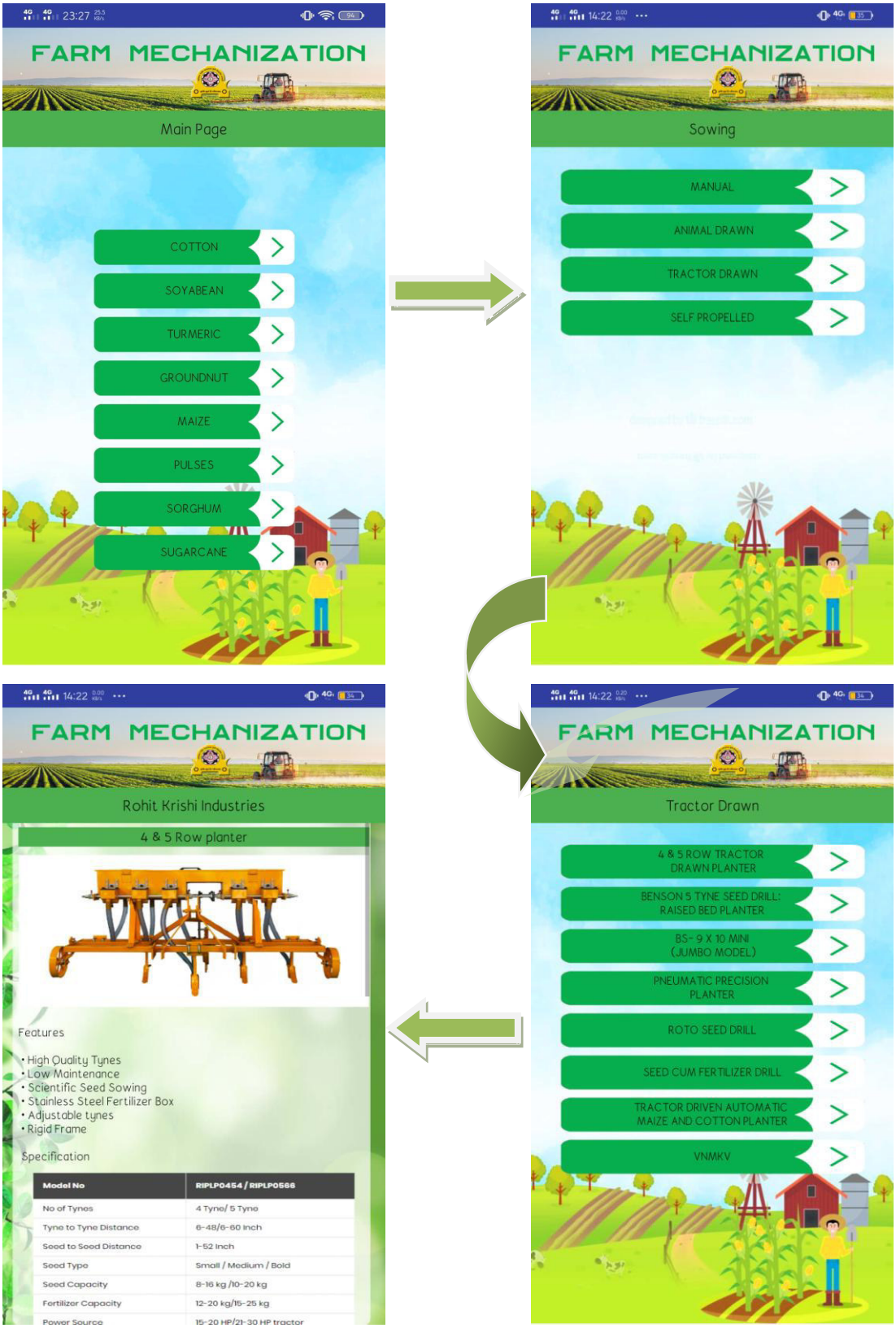
Now, sowing as the another operation. Sowing categorized shown in following screens.



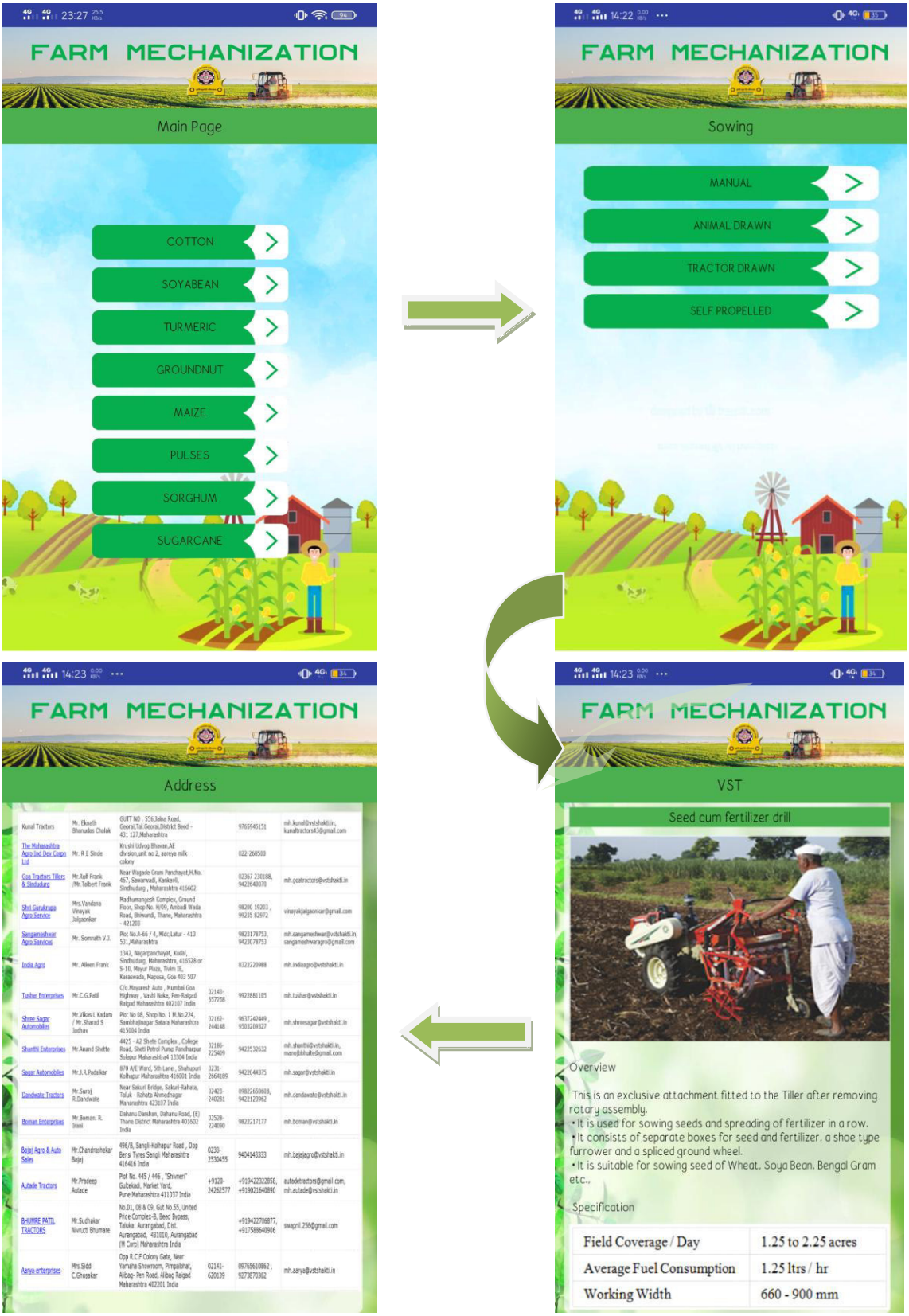
In field of sowing, by clicking on Animal drawn following screen is displayed.



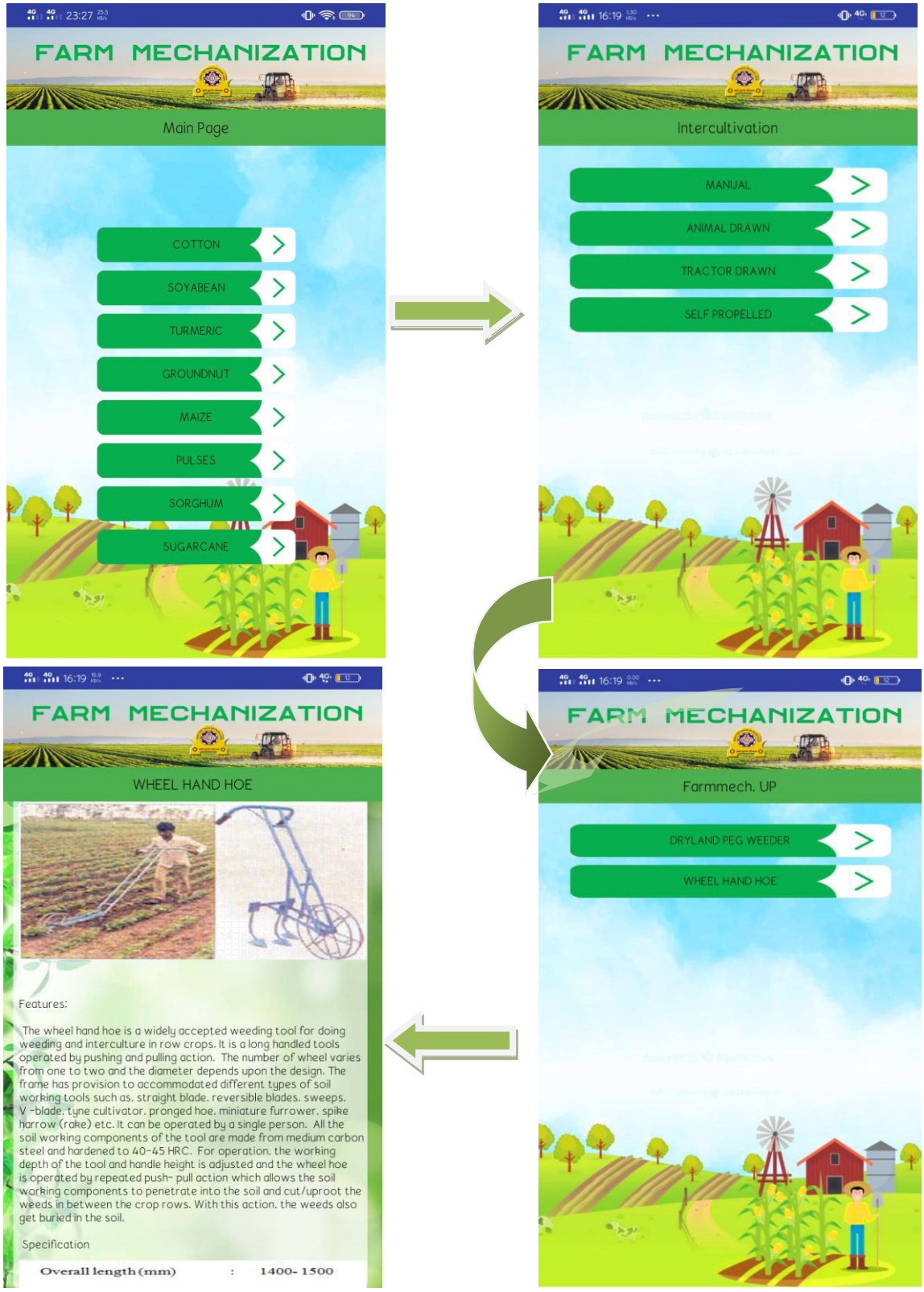
In field of sowing, by clicking on Tractor drawn following screen is displayed.



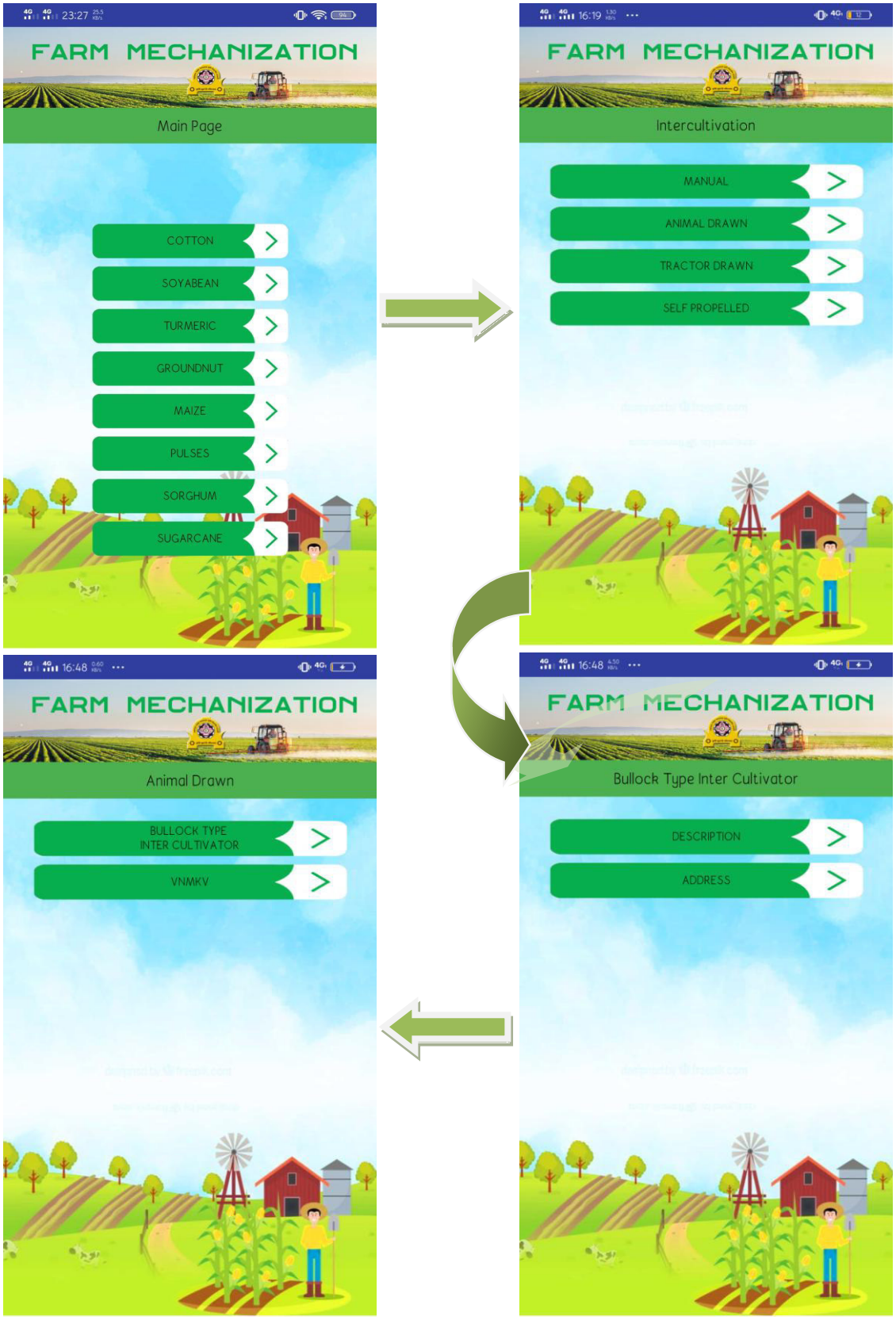
In field of sowing, by clicking on self propelled following screen is displayed.



Now, Intercultivation as the another operation. Intercultivation categorized shown in following screens.



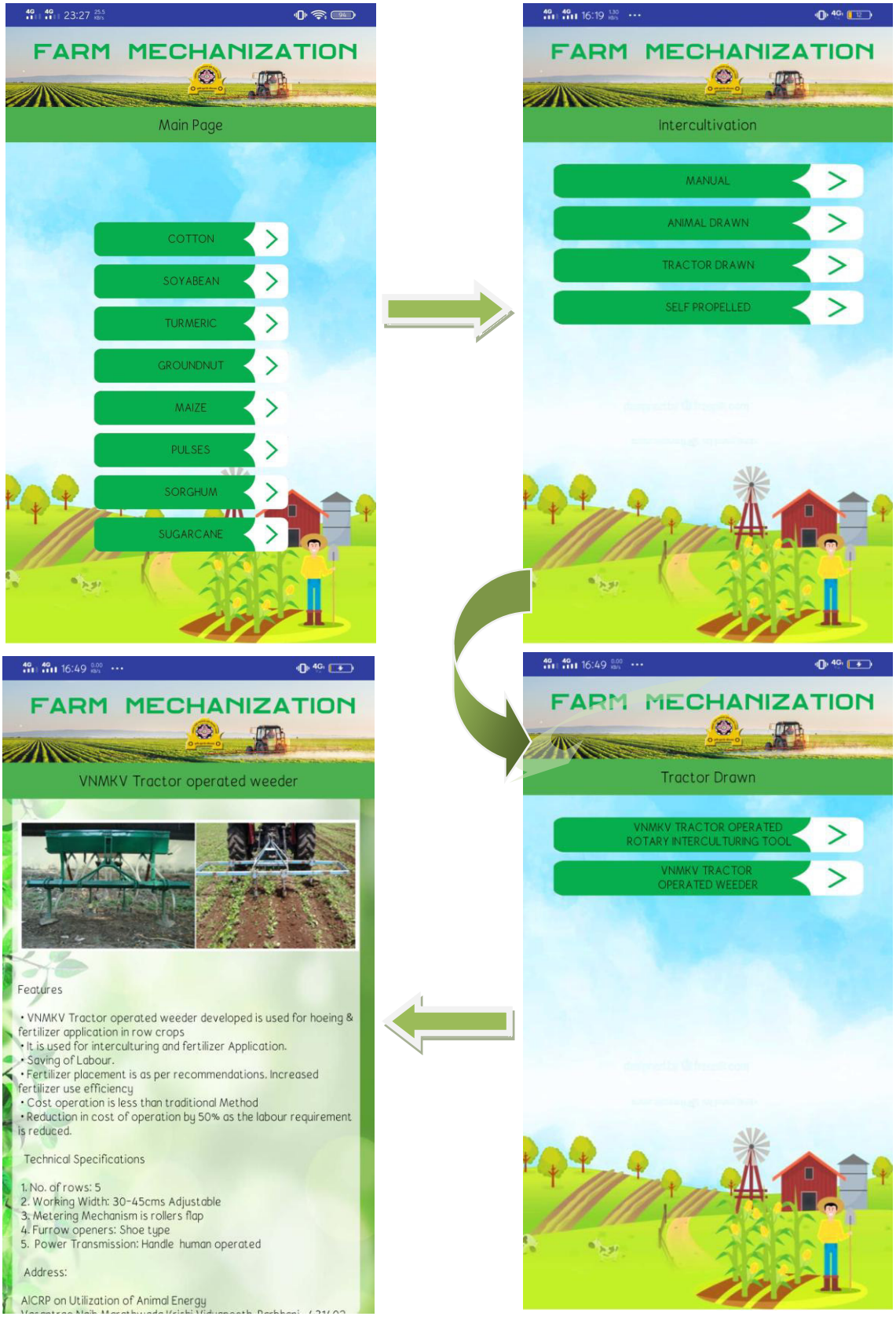
In field of Intercultivation, by clicking on Animal drawn following screen is displayed.



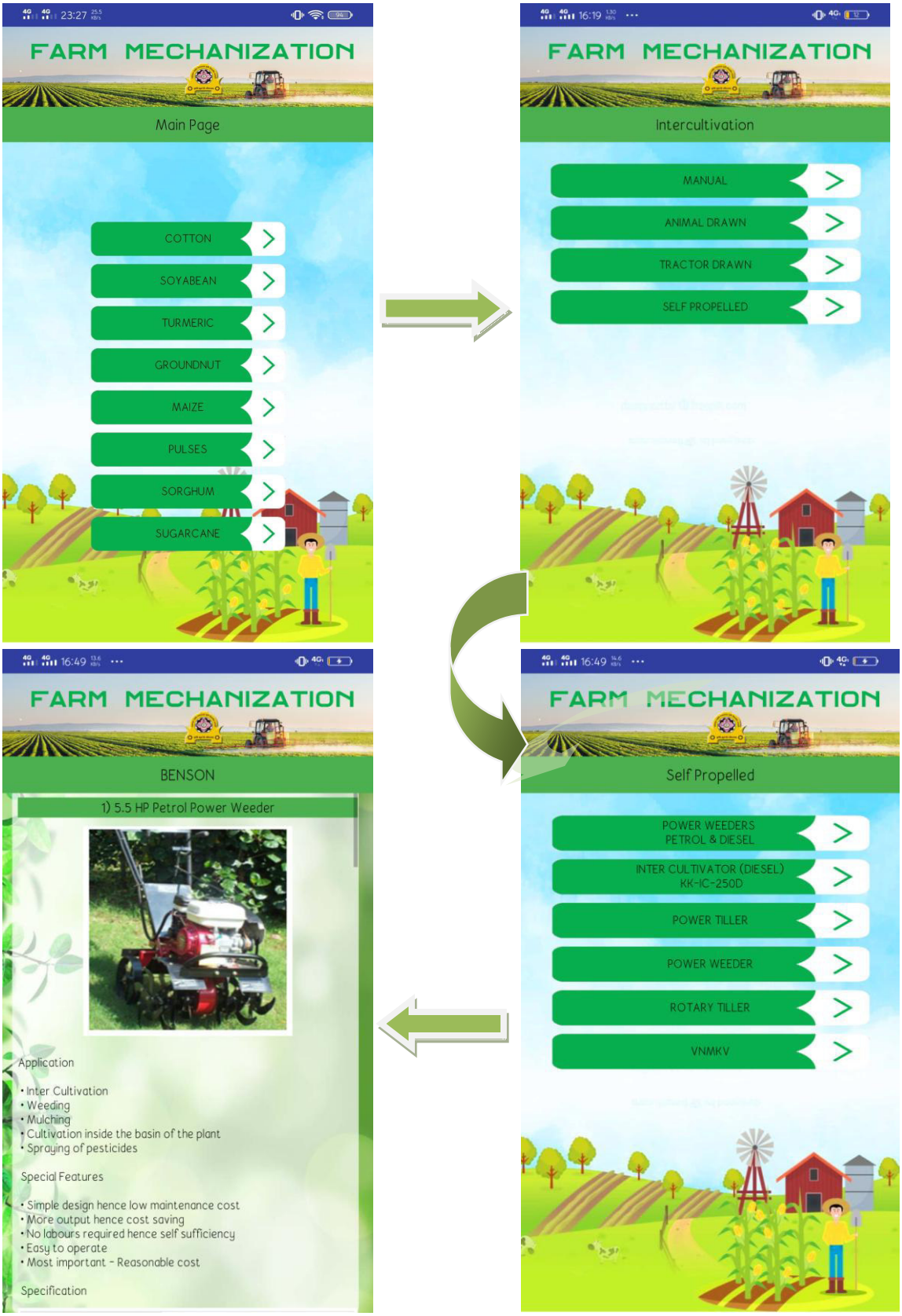

16:49
FARM MECHANIZATION
VST
Bullock type intercultivator
Overview
Inter-cultivation, also known as interculturing, is the cultivation of soil between crop rows. The soil between two rows of crop is ploughed using dedicated plough equipment for weeding, improving soil aeration and loosening the soil compaction.
Features
• A simple design attachment helps for inter cultivation in between the rows of cotton crops.
• Weeds / unwanted grass / plants are removed by using this attachment without damaging the main cotton crop.
• A person can stand on the attachment and drive the tiller.
• The field can be levelled using a Blade in the place of cultivator.
Specification
Field Coverage
1.25 to 2 acres / hr
Average Fuel Consumption
1 - 1.25 ltrs / hr


16:49
FARM MECHANIZATION
Address
Kunal Tractors | Mr. Eknath Bhanudas Chalak | GUTT NO . 556,Jalna Road, Georai,Tal.Georai,District Beed - 431 127,Maharashtra | | 9765945151 | mh.kunal@vstshakti.in, kunaltractors43@gmail.com
The Maharashtra Agro Ind Dev Corpn Ltd | Mr. R E Sinde | Krushi Udyog Bhavan,AE division,unit no 2, aareya milk colony | | 022-268500 |
Goa Tractors Tillers & Sindudurg | Mr.Rolf Frank /Mr.Talbert Frank | Near Wagade Gram Panchayat,H.No. 467, Sawarwadi, Kankavli, Sindhudurg , Maharashtra 416602 | | 02367 230188, 9422640070 | mh.goatractors@vstshakti.in
Shri Gurukrupa Agro Service | Mrs.Vandana Vinayak Jalgaonkar | Madhumangesh Complex, Ground Floor, Shop No. H/09, Ambadi Wada Road, Bhiwandi, Thane, Maharashtra - 421203 | | 98200 19203 , 99235 82972 | vinayakjalgaonkar@gmail.com
Sangameshwar Agro Services | Mr. Somnath V.J. | Plot No.A-66 / 4, Midc,Latur - 413 531,Maharashtra | | 9823178753, 9423078753 | mh.sangameshwar@vstshakti.in, sangameshwaragro@gmail.com
India Agro | Mr. Aileen Frank | 1342, Nagarpanchayat, Kudal, Sindhudurg, Maharashtra, 416528 or S-10, Mayur Plaza, Tivim IE, Karaswada, Mapusa, Goa 403 507 | | 8322220988 | mh.indiaagro@vstshakti.in
Tushar Enterprises | Mr.C.G.Patil | C/o.Mayuresh Auto , Mumbai Goa Highway , Vashi Naka, Pen-Raigad Raigad Maharashtra 402107 India | 02143-657258 | 9922881105 | mh.tushar@vstshakti.in
Shree Sagar Automobiles | Mr.Vikas L Kadam / Mr.Sharad S Jadhav | Plot No 08, Shop No. 1 M.No.224, Sambhajinagar Satara Maharashtra 415004 India | 02162-244148 | 9637242449 , 9503209327 | mh.shreesagar@vstshakti.in
Shanthi Enterprises | Mr.Anand Shette | 4425 - A2 Shete Complex , College Road, Sheti Petrol Pump Pandharpur Solapur Maharashtra4 13304 India | 02186-225409 | 9422532632 | mh.shanthi@vstshakti.in, manojbbhulte@gmail.com
Sagar Automobiles | Mr.J.R.Padalkar | 870 A/E Ward, 5th Lane , Shahupuri Kolhapur Maharashtra 416001 India | 0231-2664189 | 9422044375 | mh.sagar@vstshakti.in
Dandwate Tractors | Mr.Suraj R.Dandwate | Near Sakuri Bridge, Sakuri-Rahata, Taluk - Rahata Ahmednagar Maharashtra 423107 India | 02423-240281 | 09822650608, 9422123962 | mh.dandswate@vstshakti.in
Boman Enterprises | Mr.Boman. R. Irani | Dahanu Darshan, Dahanu Road, (E) Thane District Maharashtra 401602 India | 02528-224090 | 9822217177 | mh.boman@vstshakti.in
Bajaj Agro & Auto Sales | Mr.Chandrashekar Bajaj | 496/B, Sangli-Kolhapur Road , Opp Bensi Tyres Sangli Maharashtra 416416 India | 0233-2530455 | 9404143333 | mh.bajajagro@vstshakti.in
Autade Tractors | Mr.Pradeep Autade | Plot No. 445 / 446 , "Shivneri" Gultekadi, Market Yard, Pune Maharashtra 411037 India | +9120-24262577 | +919422322858, +919021640890 | autadetractors@gmail.com, mh.autade@vstshakti.in
BHUMRE PATIL TRACTORS | Mr.Sudhakar Nivrutti Bhumare | No.01, 08 & 09, Gut No.55, United Pride Complex-B, Beed Bypass, Taluka: Aurangabad, Dist. Aurangabad, 431010, Aurangabad (M Corp) Maharashtra India | | +919422706877, +917588640906 | swapnil.256@gmail.com
Aarya enterprises | Mrs.Sidd C.Ghosekar | Opp R.C.F Colony Gate, Near Yamaha Showroom, Pimpalbhat, Alibag- Pen Road, Alibag Raigad Maharashtra 402201 India | 02141-620139 | 09765610862 , 9273870362 | mh.aarya@vstshakti.in

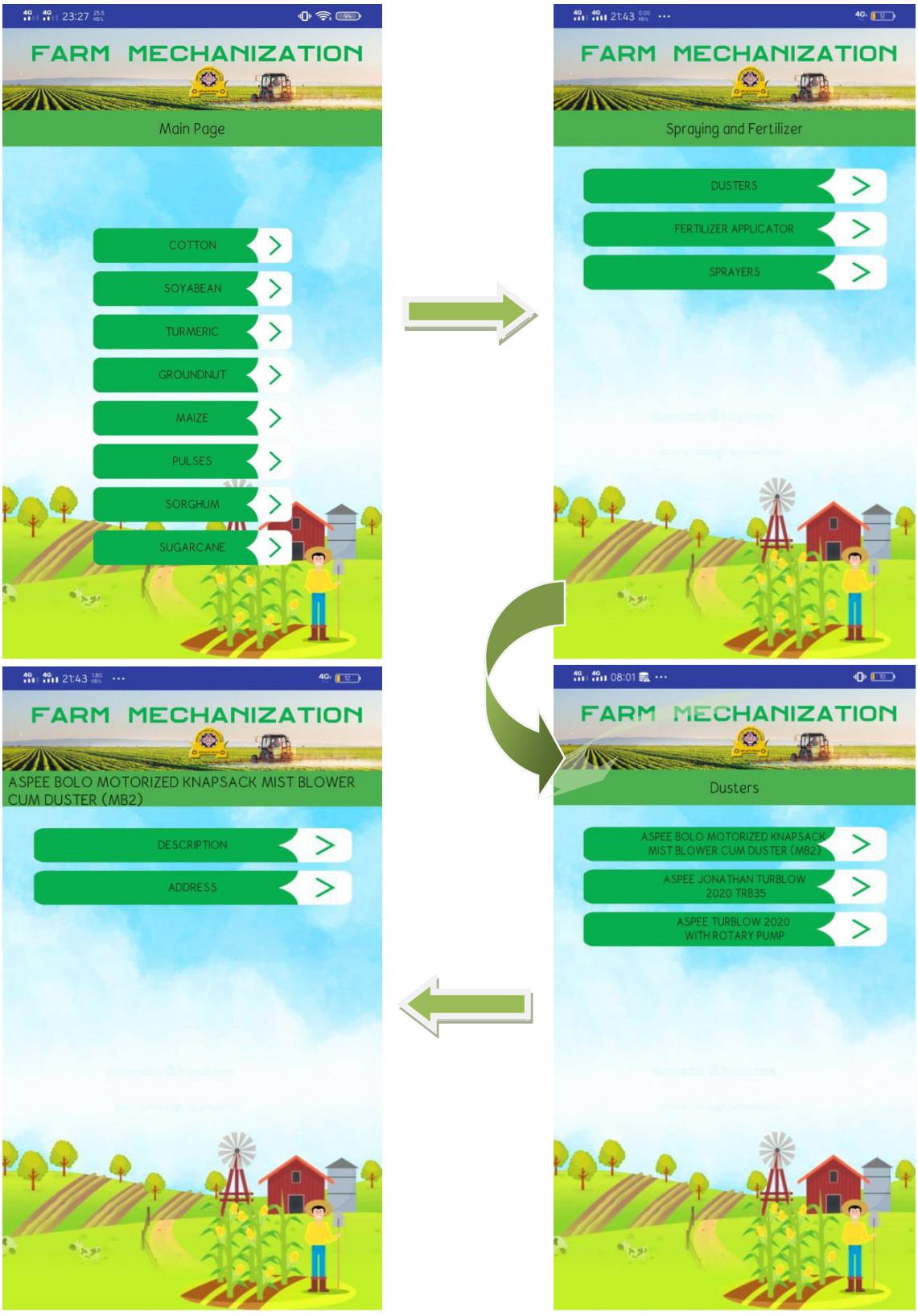
In field of Intercultivation, by clicking on Tractor drawn following screen is displayed.



In field of Intercultivation, by clicking on Self propelled following screen is displayed.

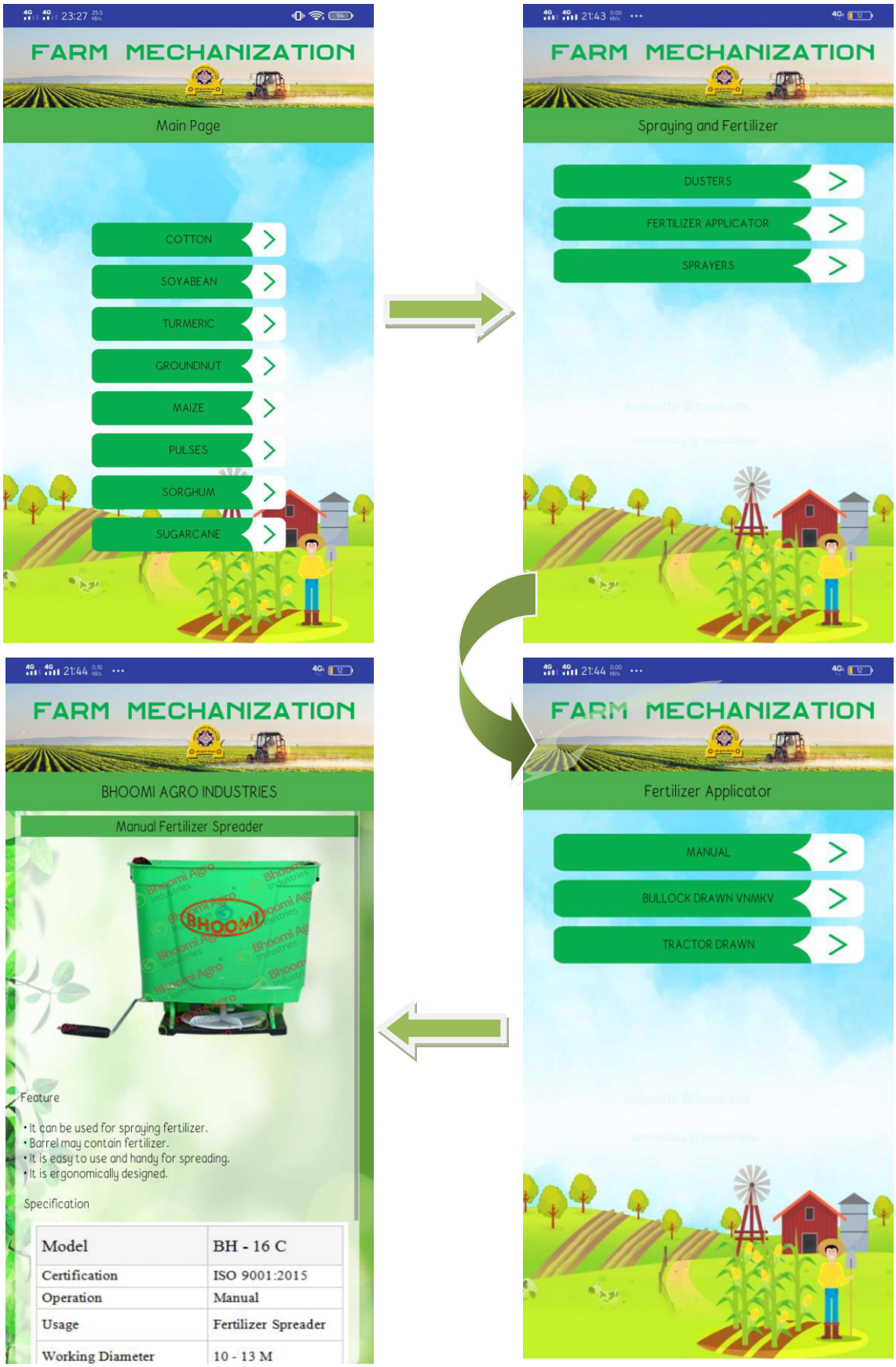


Now, Spraying as the another operation. Spraying categorized shown in following screenshots.

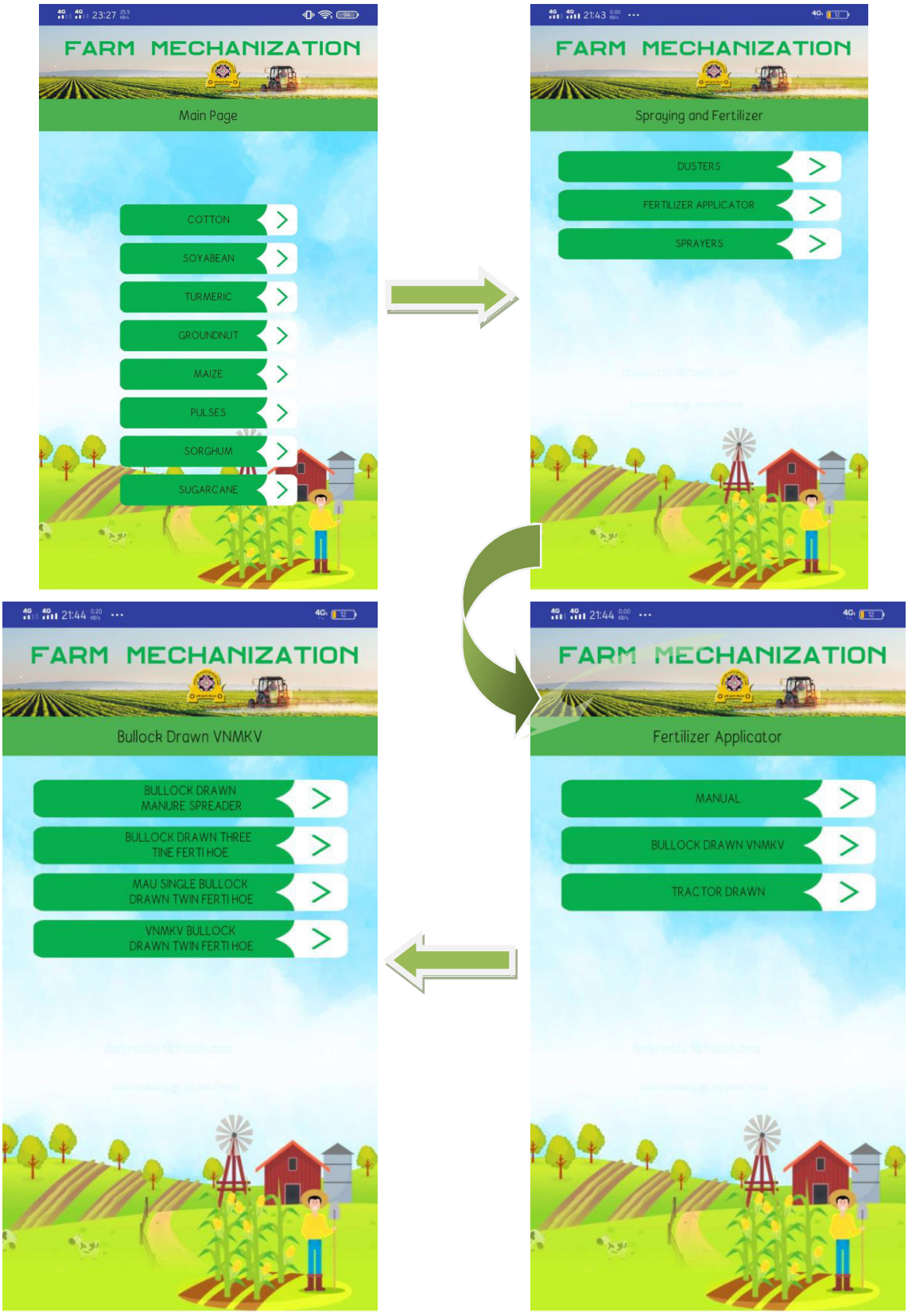


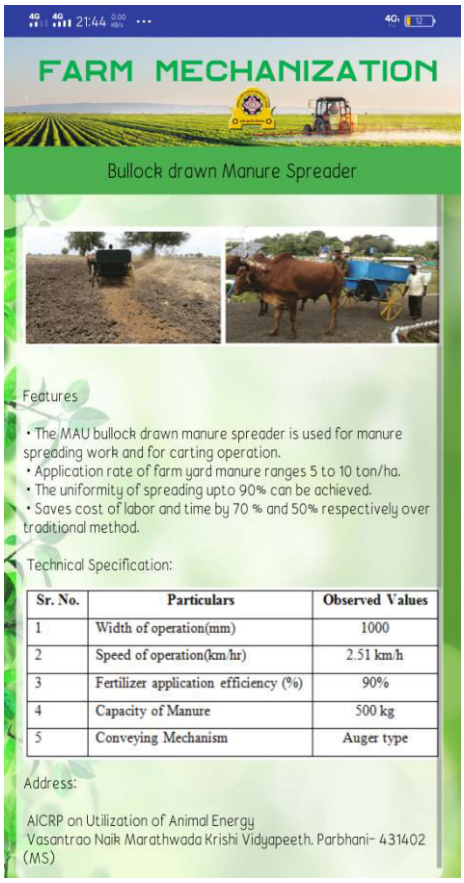

21:43
FARM MECHANIZATION
ASPEE
Knapsack Mist blower
ASPEE BOLO MOTORIZED KNAPSACK MIST BLOWER CUM DUSTER (MB2)
Spraying operations in orchards, tea, coffee & other crops
• Rest comfortably on operators back
• Heavy duty 11.5 liter chemical tank
• Sturdy frame design
• Sturdy aluminum rotor develops powerful blast of air for effective coverage
• Equipment supply with ISI mark
Features
• Rest comfortably on operators back
• Heavy duty 11.5 liter chemical tank
• Sturdy frame design
• Sturdy aluminum rotor develops powerful blast of air for effective coverage


21:44
FARM MECHANIZATION
Address
1. Nilesh Hardware and Machinery : Aspee Spray Pump in Nashik
Nashik, Maharashtra · 080555 49898
2. ASPEE Plant Protection Applicances Pvt. Ltd.
Nare, Maharashtra
3. Laxmi Agro Agencies
Pune, Maharashtra · 095454 62125
4. ASPEE American Spring and Pressing Works Pvt. Ltd.
Mumbai, Maharashtra · 022 6774 5700
5. Siddheshwar Agro Services
Aherwadi, Maharashtra
6. Aspee Industries
Mumbai, Maharashtra · 022 2285 0769
7. Agro Auto
Solapur, Maharashtra
8. SONGANGA TRADERS
Gadhinglaj, Maharashtra
9. Shri. Swami Samarth Agro Industries Main, Mahuli
Patil Mala, Maharashtra
10. Anu Enterprises
Sangli, Maharashtra · 098224 13223
11. Shree Agency
Sangli, Maharashtra
12. Jay Yogeshwar Agro Industries
Wai Bazaar, Maharashtra

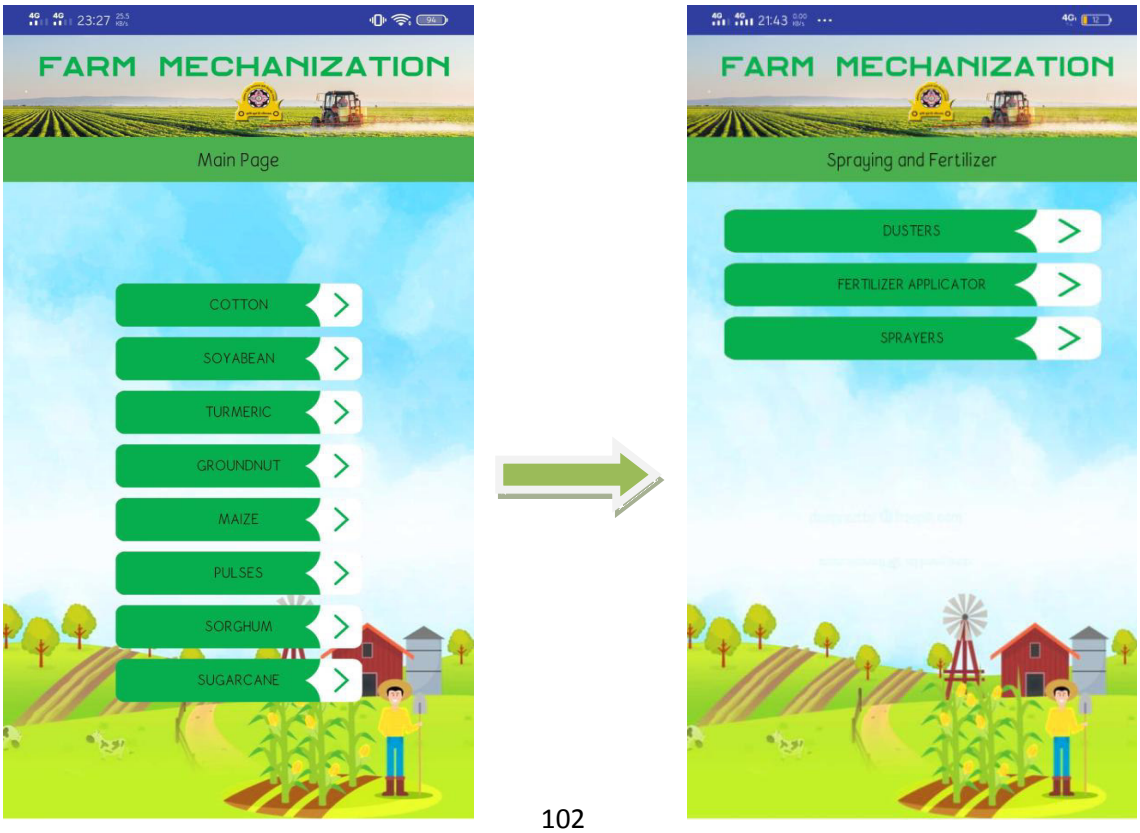

FARM MECHANIZATION
Main Page
COTTON
SOYABEAN
TURMERIC
GROUNDNUT
MAIZE
PULSES
SORGHUM
SUGARCANE
FARM MECHANIZATION
Spraying and Fertilizer
DUSTERS
FERTILIZER APPLICATOR
SPRAYERS
FARM MECHANIZATION
Fertilizer Applicator
MANUAL
BULLOCK DRAWN VNMKV
TRACTOR DRAWN
FARM MECHANIZATION
BHOOMI AGRO INDUSTRIES
Manual Fertilizer Spreader
Feature
• It can be used for spraying fertilizer.
• Barrel may contain fertilizer.
• It is easy to use and handy for spreading.
• It is ergonomically designed.
Specification
Model BH - 16 C
Certification ISO 9001:2015
Operation Manual
Usage Fertilizer Spreader
Working Diameter 10 - 13 M

In field of Fertilizer Applicator, by clicking on Animal drawn following screen is displayed.





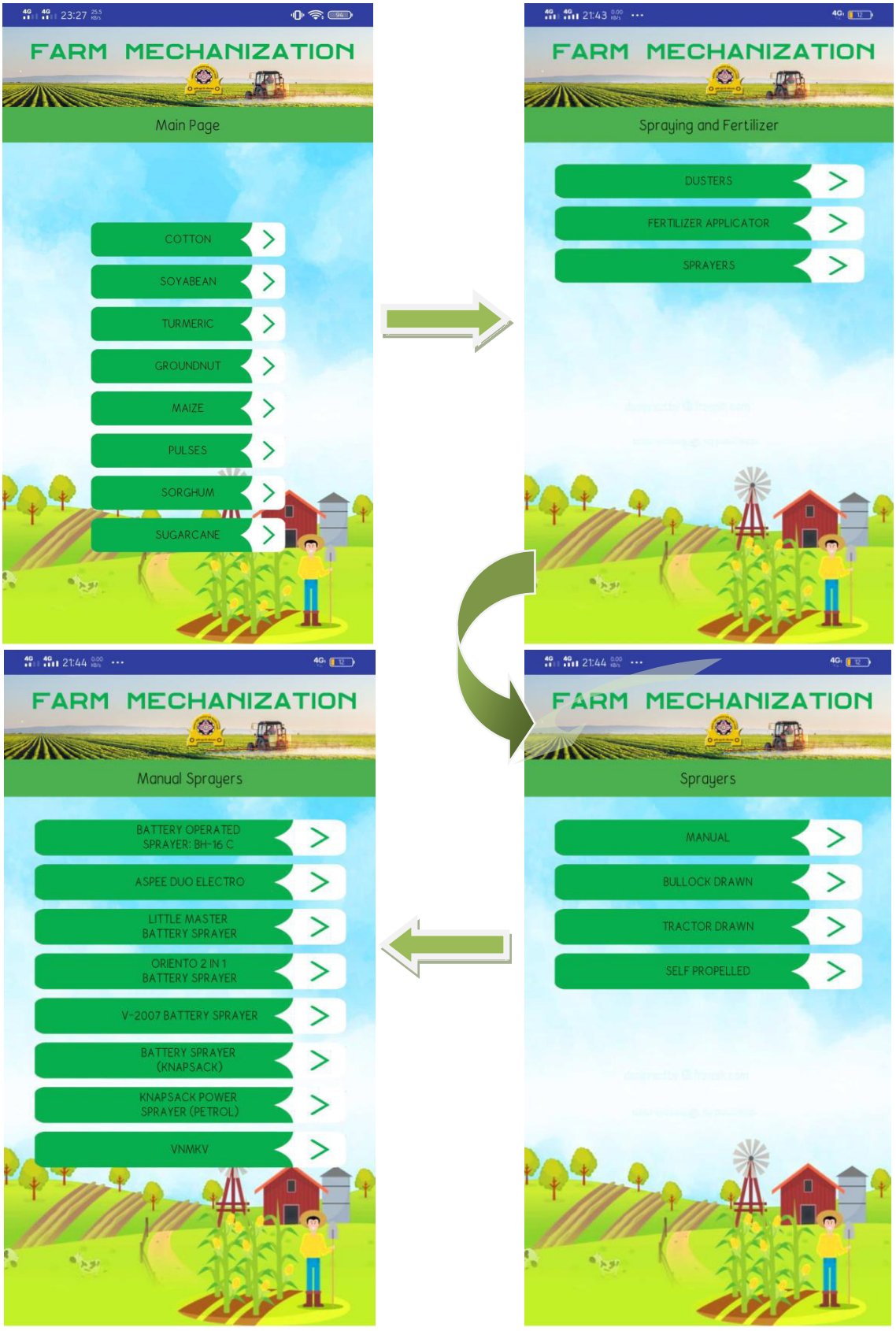
In field of Fertilizer Applicator, by clicking on Tractor drawn following screen is displayed.

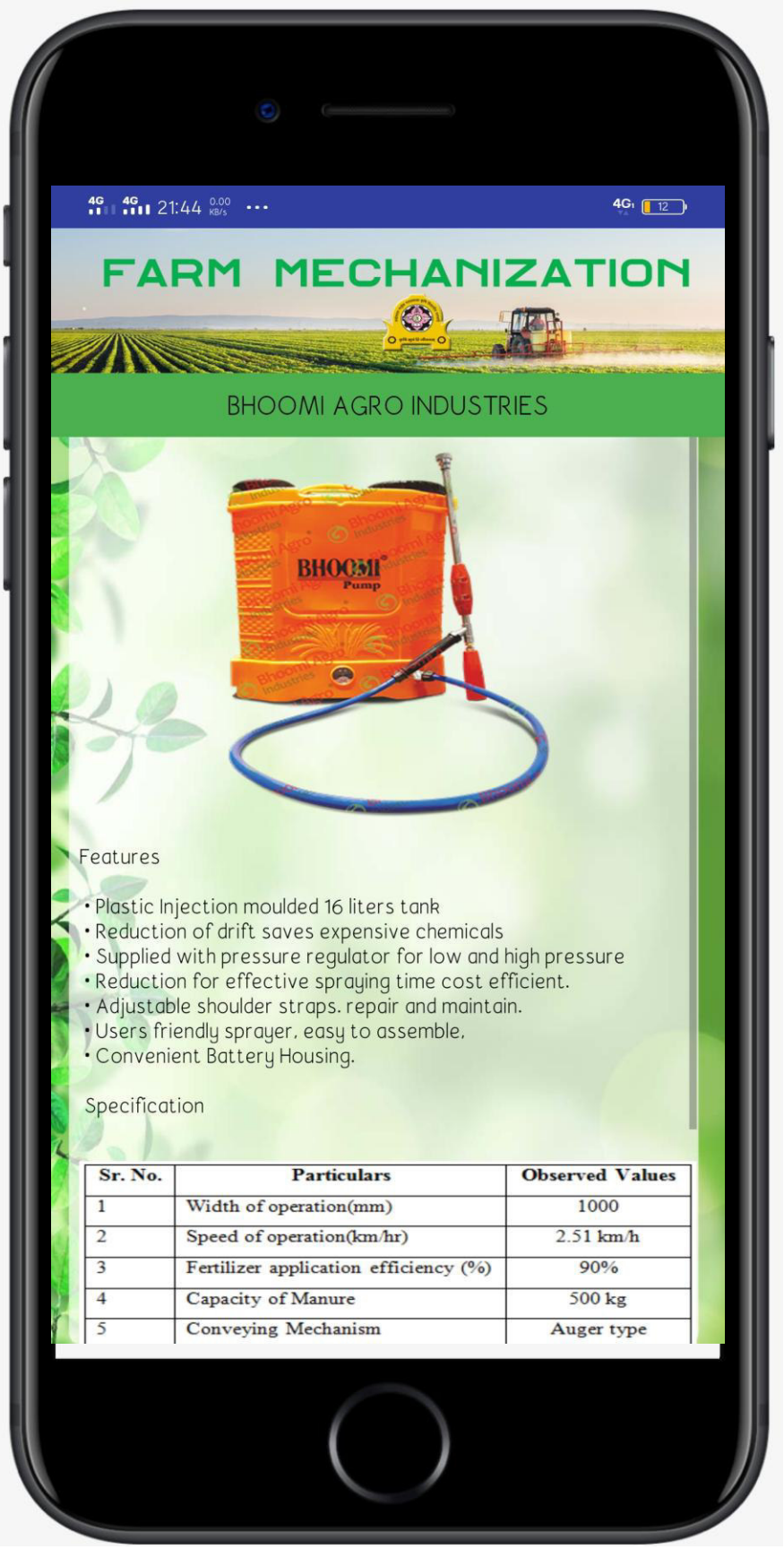



21:44
FARM MECHANIZATION
Tractor Drawn
CONICAL FERTILIZER BROADCASTER
FERTILIZER BROADCASTER
SQUARE FERTILIZER BROADCASTER

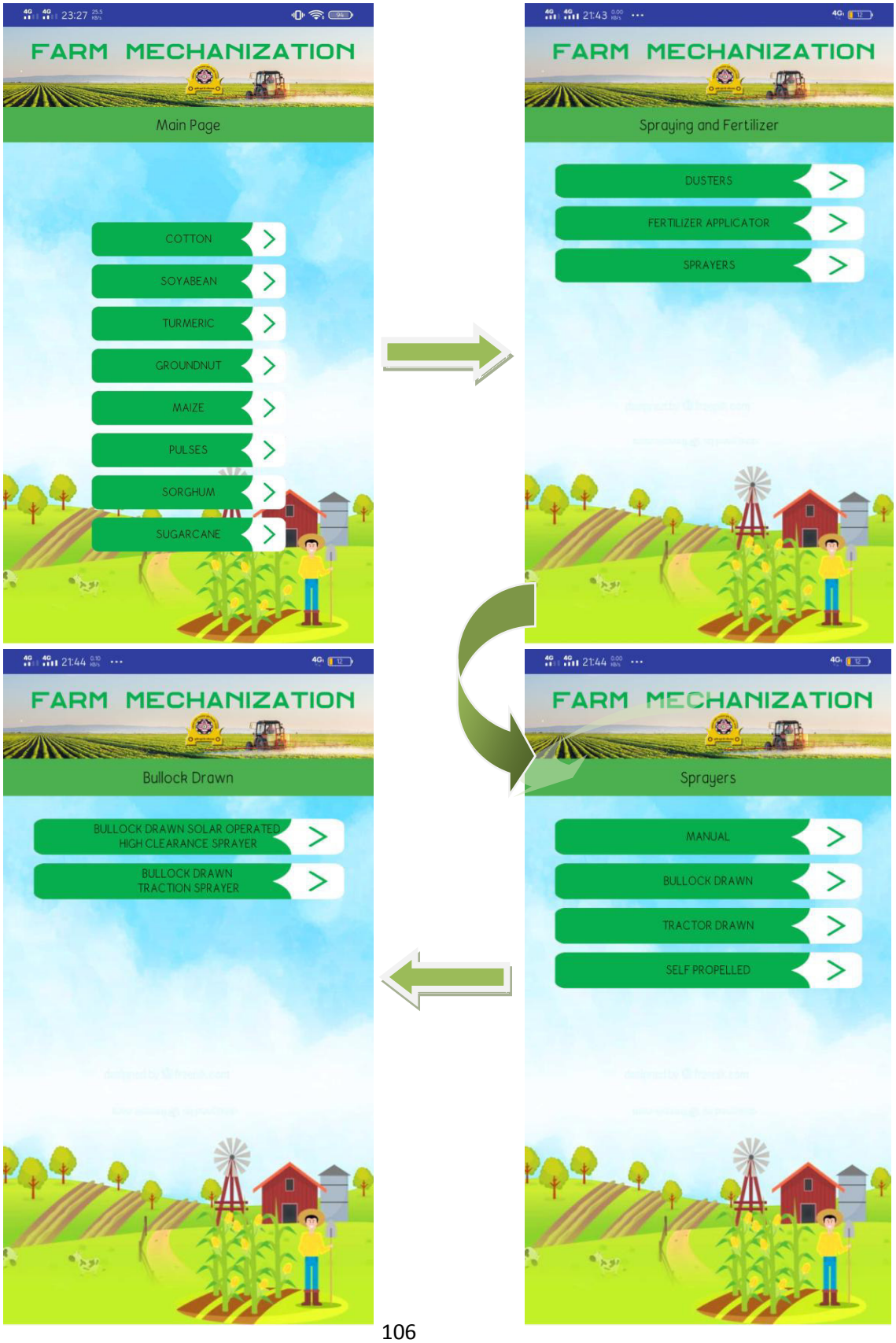

21:44
FARM MECHANIZATION
SHAKTIMAN
Conical Fertilizer Broadcaster
Description
• SHAKTIMAN Fertilizer Broadcaster implement designed to spread fertilizers with greater ease and efficiency.
• They can spread the fertilizer fall on 12 to 14 meter width in field.
• The opening system with independent levers allows the operator to choose to spread the fertilizer only where it is required.
Advantages
• It allows the operator to spread fertilizers in larger land portion in lesser time.
• The unique distributing system ensures uniform spreading of fertilizers over greater distanceswhich improves organic structure of the land.
• Faster operation saves time and power consumption.
• Its 17 teeth gears stabilize the impact of sudden power change and safeguard the machine.
Technical Specification
Capacity | litres | 345
Fertilizer Broadcaster Height | inch/mm | 44.5 / 1130

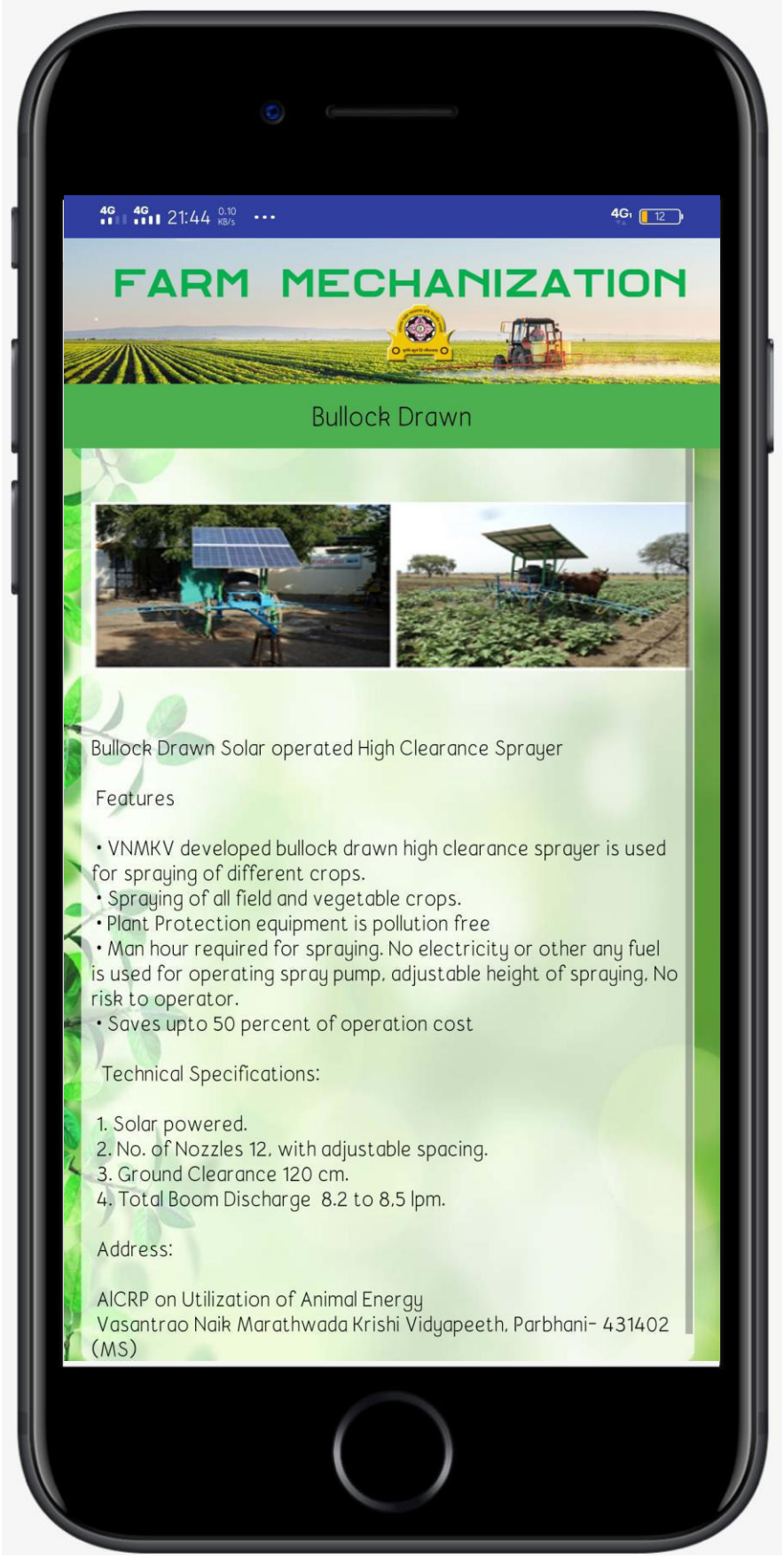
In field of Sprayer, clicking on manual drawn sprayer following screen is displayed.



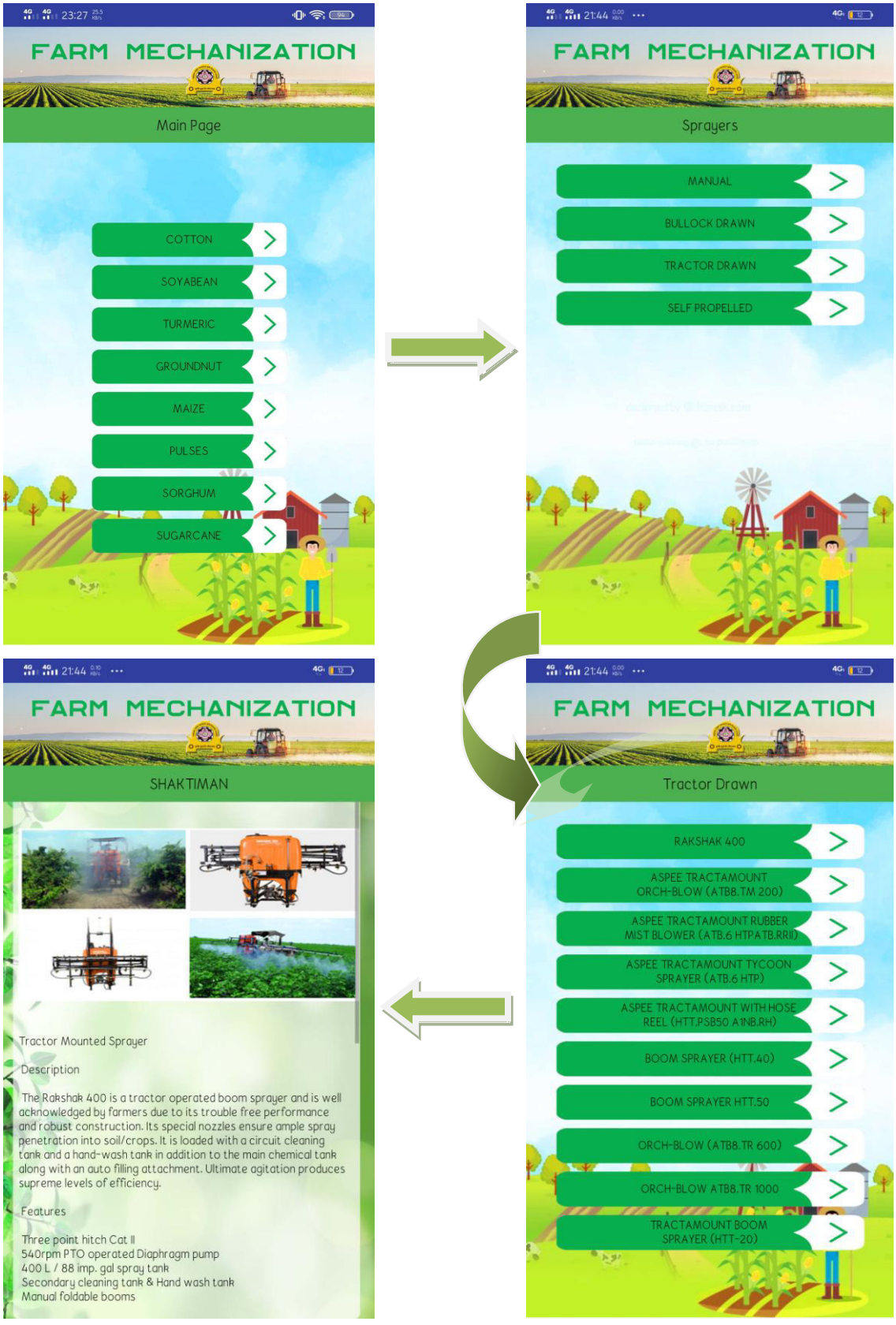

21:44
FARM MECHANIZATION
BHOOMI AGRO INDUSTRIES
BHOOMI
Pump
Features
• Plastic Injection moulded 16 liters tank
• Reduction of drift saves expensive chemicals
• Supplied with pressure regulator for low and high pressure
• Reduction for effective spraying time cost efficient.
• Adjustable shoulder straps. repair and maintain.
• Users friendly sprayer, easy to assemble.
• Convenient Battery Housing.
Specification
Sr. No. | Particulars | Observed Values
1 | Width of operation(mm) | 1000
2 | Speed of operation(km/hr) | 2.51 km/h
3 | Fertilizer application efficiency (%) | 90%
4 | Capacity of Manure | 500 kg
5 | Conveying Mechanism | Auger type

In field of Sprayer, clicking on bullock drawn sprayer shown in following screens.

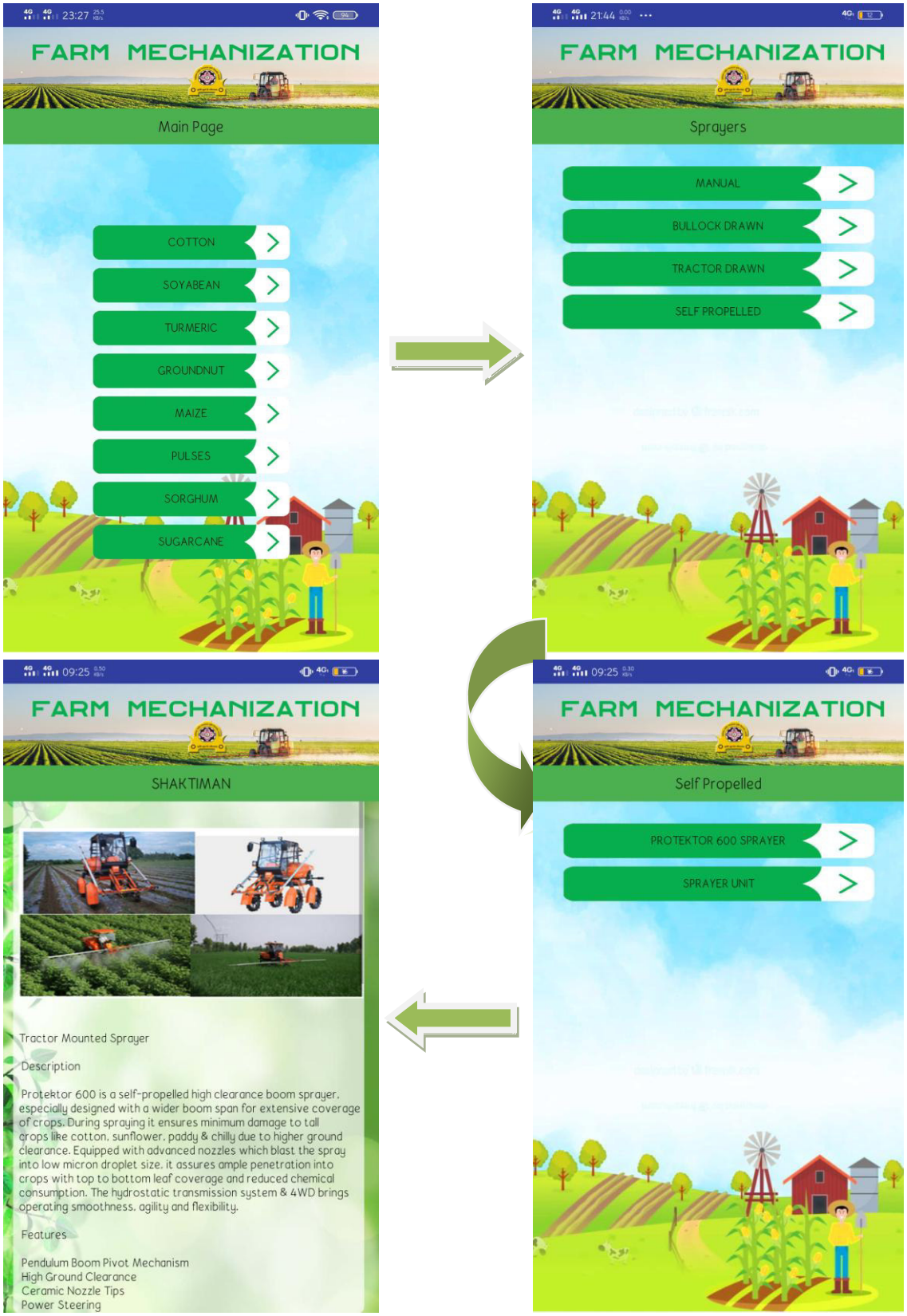



21:44
FARM MECHANIZATION
Bullock Drawn
Bullock Drawn Solar operated High Clearance Sprayer
Features
• VNMKV developed bullock drawn high clearance sprayer is used for spraying of different crops.
• Spraying of all field and vegetable crops.
• Plant Protection equipment is pollution free
• Man hour required for spraying. No electricity or other any fuel is used for operating spray pump, adjustable height of spraying, No risk to operator.
• Saves upto 50 percent of operation cost
Technical Specifications:
1. Solar powered.
2. No. of Nozzles 12, with adjustable spacing.
3. Ground Clearance 120 cm.
4. Total Boom Discharge 8.2 to 8.5 lpm.
Address:
AICRP on Utilization of Animal Energy
Vasantrao Naik Marathwada Krishi Vidyapeeth, Parbhani- 431402 (MS)

In field of Sprayer, clicking on tractor drawn sprayer following screen is displayed.

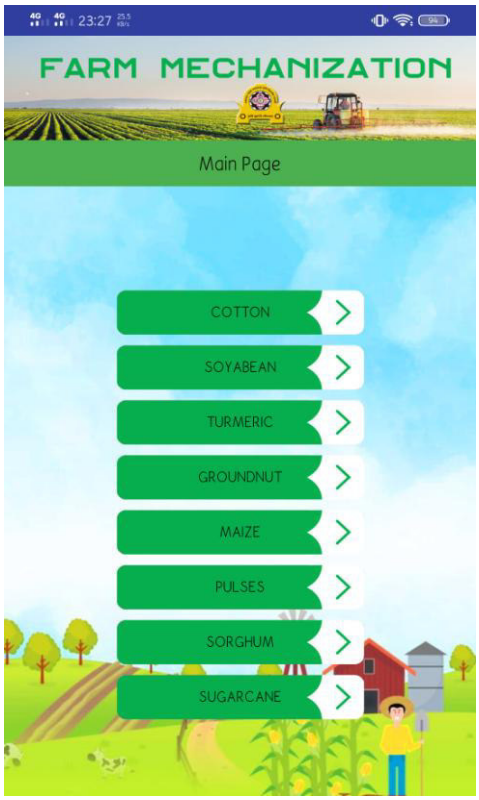


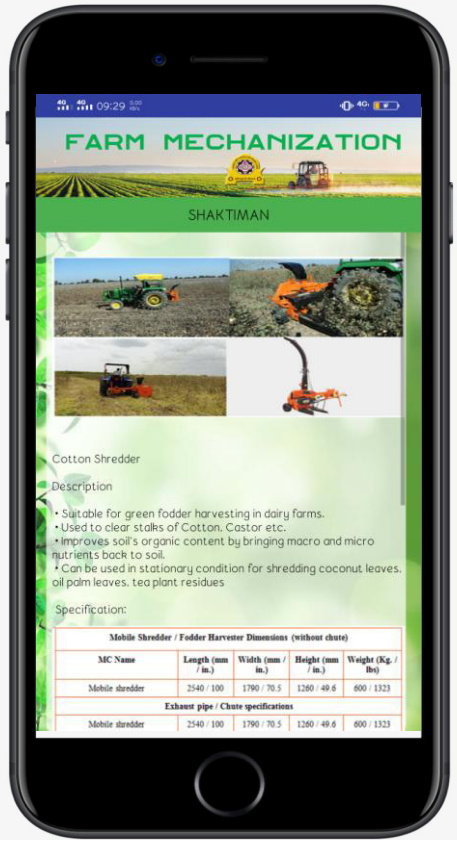
Now field of sprayer, clicking on self propelled following screen is displayed.



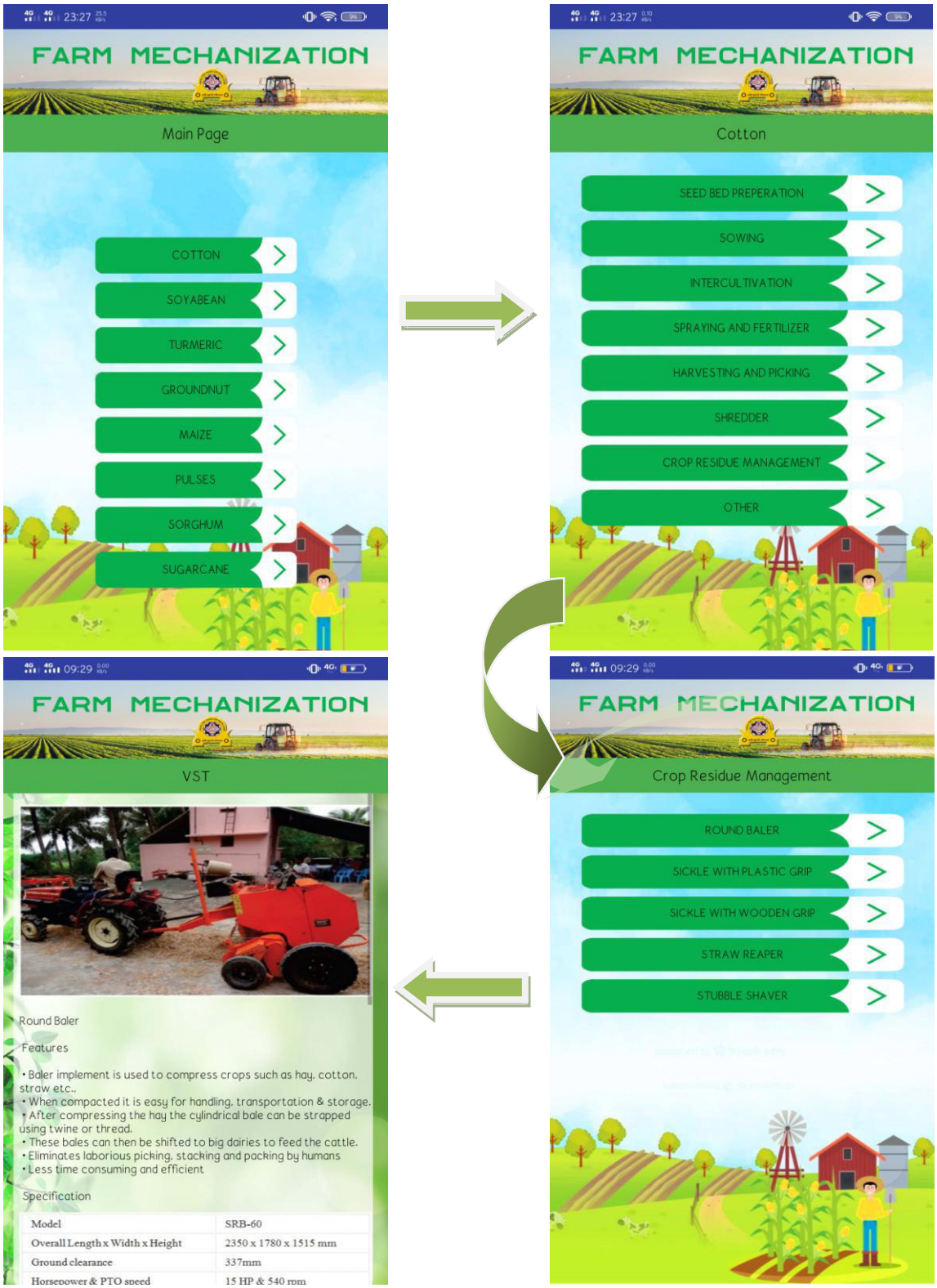
Now, Harvesting and picking as the another operation. Harvesting and picking categorized screen is displayed.

Now, shredding as another operation, by clicking on shredder following screen is displayed.

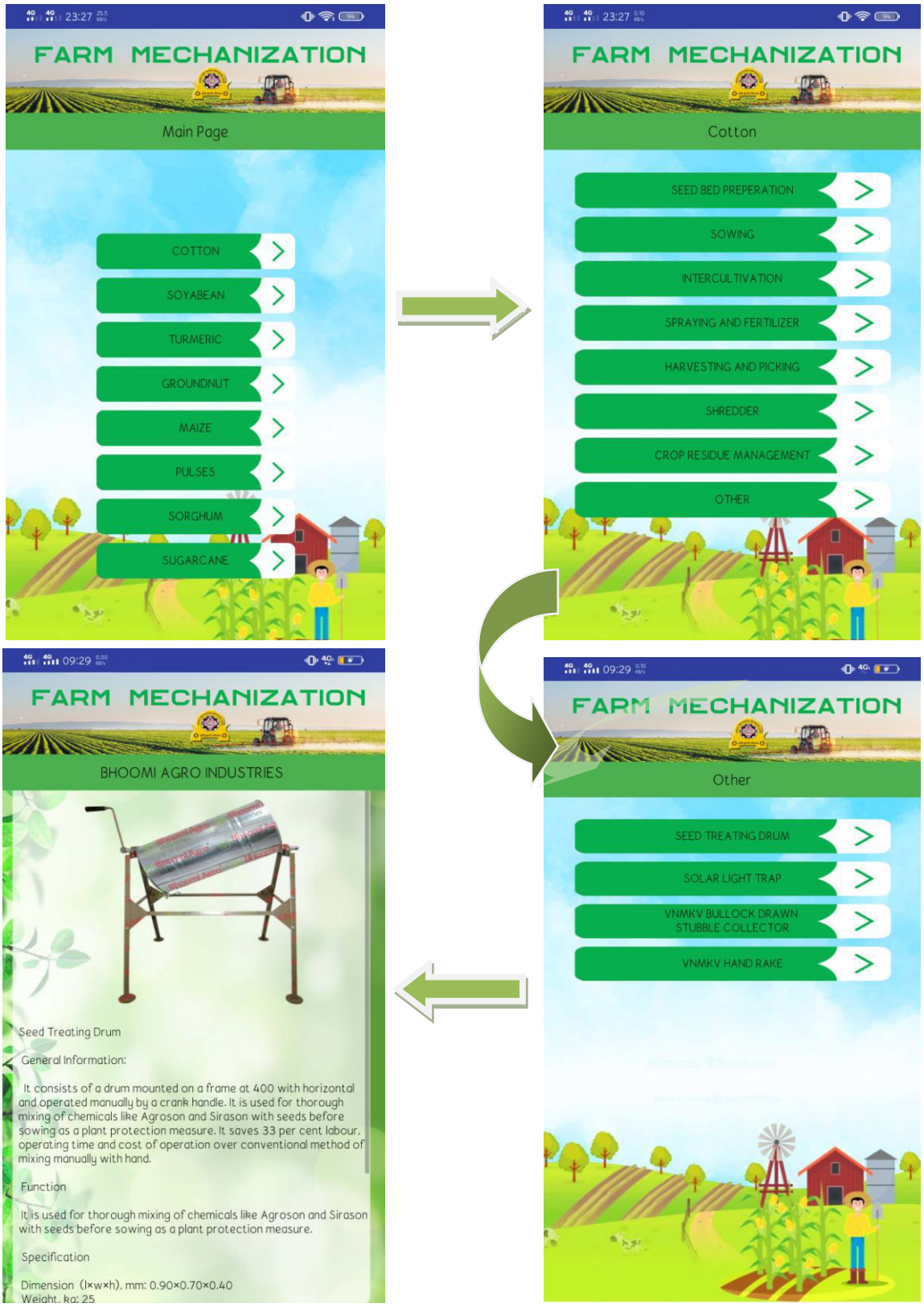




Now, Residue management, by clicking on Crop Residue Management following screen is displayed.



Now, Remaining gadgets or implements categorized in others, by clicking on others following screen is displayed.



CHAPTER -V

SUMMARY AND CONCLUSION

CHAPTER-V

SUMMARY AND CONCLUSIONS

5.1 Summary

Farm Mechanization play lead role in Indian economy to achieve modernization in agriculture sector to give substantial growth in agriculture by the means of adopting the modern technology. Farmers are well equipped with android source as compared iOS operating system smartphones. Although Android is an open and free source OS, an abundance of resources, development tools, guidelines and data can be accessed for free on the Android open - sourced community forums. Looking towards the issues in agriculture, developed mobile application to overcome the problem of accepting suitable machinery for desired crop. Mainly, Android is a truly open source and free development platform, taking into consideration Android based Mobile Application for Farm Mechanization has been developed.

The developed app of farm mechanization contains the different farm machinery for cultivation practices of major eight crops which are grown in this region categorized in four modes of operation viz. Manual, Animal, Tractor, Self Propelled. The fields of machine contain information about specification, feature, Image and video link. In this app, it also provides availability of machinery or implement by showing nearer authorized dealer details by contacting them.

5.2 Conclusion

Agriculture is the mainstay of Indian economy as more than over 70% of the rural households that depends on agriculture. Agriculture is the primary source of income to India's rural population. The goal of this research is to design and develop a mobile application in consideration of need base technology to farmer. To overcome with the farmer issue and problem we investigate farmers are not acquaint with the machinery all over them. They usually prefer traditional method whereas, to reduce the drudgery and strenuous condition work, mechanized machinery is useful.

The foremost aim of this thesis was to design and develop Android based mobile application for Farm Mechanization. Hence, android based mobile app for evaluating agricultural operation machinery to crop cultivation practices for different

operations, which contribute to reduces the mechanization gap. This app only took forward the same to reduce the mechanization gap it shows all type of machinery from reasonable to high cost with respect to power. This app have all type of machines for four mode of operation i.e manually, bullock, tractor and self propelled.

Moreover, the software was tested on an Android emulator and an Android smartphone. The application operated properly and the UI components on the mobile-phone respond quickly.

1. Mobile app is easy to use and can be simply modified to add more features.
2. This app is compatible in every smartphone above 5.1 lollipop version.
3. It is developed based on three tier architecture and this architecture has three layers which are client side layer, sever application layer and database layer.
4. This app gives the machinery for cultivation practices for selected eight crops.
5. The app has been developed based on three objectives in mind. According the first objective to study the existing machinery unknown and know to the farmer.

   Then in the second objective, to develop android based mobile app for farm mechanization and last objective to test the developed mobile app using emulator.
6. No any registration part in the app which adheres to farmer for use the app.
7. This app provides different type of machinery to the user for variety of crop with different sizes of machinery with respect to power.
8. Farmers or user can see the dynamic app timely updated with loads of features and more package of practices of different crop.
9. In this app it shows availability of machinery with the address and contact number of authorized dealers.

The application is finished in itself and prepared to be actualized yet every task needs to experience changes according to the further research necessities and time. The same is valid for this application.

SUGGESTION FOR FUTURE WORK

1. Developed application may have “Chatbot” facility for the farmer to overcome their queries.
2. Registration form or sign up process for farmer and fabricator to be include in it for creating database of famer and manufacturer.
3. Mail to Dealers or Manufacturer about the charges for putting forward your machinery in the application and similarity addition for other dealers or manufacture in the list.
4. Copywrite for the developed application is suggested.
5. Excess of data may stored in cloud storage i.e data regarding machinery that have to displayed for users are stored in cloud storage.
6. Make application to be upgraded everytime over different crops even after commercialization.

LITERATURE CITED

LITERATURE CITED

Abishek, A. G., Bharathwaj, M., & Bhagyalakshmi, L. (2016). Agriculture marketing using web and mobile based technologies. *Proceedings - 2016 IEEE International Conference on Technological Innovations in ICT for Agriculture and Rural Development, TIAR 2016*, *Tiar*, 41–44.

Ansari, M. A., & Pandey, N. (2013). Assessing the potential and use of mobile phones in agriculture Department of Agriculture Communication. *Karnataka Journal of Agricultural Science*, *26*(3), 388–393.

Baktha, K. (2017). Mobile Application Development: All the Steps and Guidelines for Successful Creation of Mobile App. *International Journal of Computer Science and Mobile Computing*, *6*(9), 15–20.

Barh, A., & Balakrishnan, M. (2018). Smart phone applications: Role in agri-information dissemination. *Agricultural Reviews*, *38*(01), 82–85.

Belakeri, P., Prasad, C. K., Bajantri, S., Mahantesh, M. T., Maruthi, S. T., & Rudresh, G. N. (2017). Trends of Mobile Applications in Farming. *International Journal of Current Microbiology and Applied Sciences*, *6*(7), 2499–2512.

Bhave, A., Joshi, R., & Fernandes, R. (2014). MahaFarm – An Android Based Solution for Remunerative Agriculture. *International Journal of Research in Advent Technology*, *2*(4), 439–443.

Chaurasia, H., Islam, S. N., Priyadarshini, S., Kumar, R., & Khan, M. A. (2020). Development of Android based Crop Advisory Application for Seed Spices. *International Journal of Current Microbiology and Applied Sciences*, *9*(3), 904–913.

Chirmade, M., Tayade, K., Bankar, G. S., & Sugave, S. (2015). Agriculture Supply Chain Management Based Android Application. *International Journal of Advanced Research in Computer and Communication Engineering*, *4*(4), 86–90.

Choudhary, M., Dhone, S., Jadhav, A., Dhandal, C., & Nighot, P. J. M. (2015). Scheduling , Controlling And Monitoring of Agricultural Devices Using Android Application. *International Journal of Advanced Research in Computer Engineering & Technology, 4*(4), 1512–1515.

Confalonieri, R., Foi, M., Casa, R., and others (2013). Development of an app for estimating leaf area index using a smartphone. Trueness and precision determination and comparison with other indirect methods. *Computers and Electronics in Agriculture*, *96*, 67–74.

Delgado, B., Paredes, M., & Martínez, M. (2015). Software application for calculating solar radiation and equivalent evaporation in mobile devices. *Agricultural Water Management*, *151*, 30–36.

Gaikwad, R., Gawade, A., Sinha, P., & Chopade, P. S. (2017). *Secure Android application for Farmers based on Ratings and reviews (SAFR). 05*, 194–197.

Gyanappa A . Walikar , Ankita Suhas Kadam, and others (2018). Mobile Applications Used For Farmers : A Survey. *International Journal of Engineering Science Invention (IJESI), 7*(4), 60–66.

Intaravanne, Y., & Sumriddetchkajorn, S. (2012). BaiKhao (rice leaf) app: a mobile device-based application in analyzing the color level of the rice leaf for nitrogen estimation. *Optoelectronic Imaging and Multimedia Technology II*, *8558*, 85580F.

Islam, R., & Mazumder, T. (2010). Mobile application and its global impact. *International Journal of Engineering & Technology*, 1*0(06)*, 72–78.

Jadhav, M., & Jagtap, V. (2019). Android Application for Farmers. *International Research Journal of Engineering and Technology (IRJET)*, Volume: 06, Issue: 04. 4200–4202.

Karetsos, S., Costopoulou, C., & Sideridis, A. (2014). Developing a smartphone app for m-government in agriculture. *Journal of Agricultural Informatics*, *5*(1).

Karkhile, S. ., & Ghuge, S. . (2015). A Modern Farming Techniques using Android Application. *International Journal of Innovative Research in Science, Engineering and Technology*, *4*(10), 10499–10506.

Kavitha S & B. G. Girish. (2019). GREENDROID : Energy Efficient Mobile Application Processor Using Android System. *International Journal of Scientific Research & Engineering Trends, 5*(1), 11–13.

Kokate, K., & Singh, A. (2013). Use of Mobile Technologies for Empowering Small holder farmers in India. In *http://www.e-agriculture.org/content/use-mobile-technologies-empowering-small-holder-farmers-india (accessed 2 June 2014)* (pp. 1–9). Mobile extension for empowering smallholder farmers in India - ICAR, India.pdf

Madhumathi, R., Radhakrishnan, R., Suresh Kumar, S., Abineshkumar, K., Karthi, M., & Manojkrishna, M. (2016). Bidding application in Amazon web services for the sales of agricultural products. *2016 International Conference on Recent Trends in Information Technology, ICRTIT 2016*.

Mane, M. S. G., & Kulkarni, R. V. (2019). *Fostering Innovation , Integration and Inclusion Through Interdisciplinary Practices in Management Review on : Design and Development of Mobile App for Farmers*.

Méndez-Porras, A., Quesada-López, C., & Jenkins, M. (2015). Automated testing of mobile applications: A systematic map and review. *CIBSE 2015 - XVIII Ibero-American Conference on Software Engineering*, 195–208.

Muccini, H., Di Francesco, A., & Esposito, P. (2012). Software testing of mobile applications: Challenges and future research directions. *7th International Workshop on Automation of Software Test, AST 2012 - Proceedings*, 29–35.

Murakami, Y., Utomo, S. K. T., Hosono, K., Umezawa, T., & Osawa, N. (2013). IFarm: Development of cloud-based system of cultivation management for precision agriculture. *IEEE 2nd Global Conference on Consumer Electronics, GCCE 2013*, 233–234.

Nimbalkar R. R. (2013). Mobile Application Testing Matrix and Challenges. *International Journal of Science and Research (IJSR)*, *2*(7), 27–40.

Oinas-Kukkonen, H., & Kurkela, V. (2003). Developing successful mobile applications. *Proceedings of the IASTED International Conference on Computer Science and Technology*, 50–54.

Patel, H., & Patel, D. (2016). Survey of Android Apps for Agriculture Sector. *International Journal of Information Sciences and Techniques*, *6*(1/2), 61–67.

Patel, V. B., & Thakkar, R. G. (2014). Agricultural Android Application. *International Journal of Computer Science And Technology, 1 1 2*. *8491*(c), 326–328.

P. Tamil Selvi et al., P. T. S. et al. . (2019). The Usage of Mobile Apps in Agriculture. *International Journal of Agricultural Science and Research*, *9*(4), 221–228.

Prasad, S., Peddoju, S. K., & Ghosh, D. (2013). AgroMobile: A Cloud-Based Framework for Agriculturists on Mobile Platform. *International Journal of Advanced Science and Technology*, *59*, 41–52.

Reddy, S., Pawar, A., Rasane, S., & Kadam, S. (2015). A Survey on Crop Disease Detection and Prevention using Android Application. *International Journal of Innovative Science, Engineering & Technology*, *2*(4), 621–626.

Sharma, Shubham, Patodkar, V., Simant, S., & Shah, C. (2015). *E-Agro Android Application (Integrated Farming Management Systems for sustainable development of farmers)*. *3*(1), 4–8.

Sharma, Sunidhi, Sharma, D. K., & Sharma, S. (2008). Overview of Mobile Android Agriculture Applications. *International Research Journal of Engineering and Technology*, *225*, 225–231.

Shweta Sharan, Kamini and Neha Mahajan. (2013). Tech Productivity - An Android Based Solution for Indian Agriculture. *Oriental Journal of Computer Science and Technology*, 06 (01). 125-129.

Sopegno, A., Calvo, A., Berruto, R., Busato, P., & Bocthis, D. (2016). A web mobile application for agricultural machinery cost analysis. *Computers and Electronics in Agriculture*, *130*, 158–168.

Srija, S., Chanda, R. G., Lavanya, S., & Kalpana, D. M. (2016). AgroNutri Android Application. *International Journal of Advanced Research in Computer and Communication Engineering*, *5*(5), 808–810.

Uchinuno, T., Yasunaga, Y., Keiichi, M., Sugimoto, N., & Aoqui, S. I. (2013). Development of knowledge sharing system for agriculture application. *Proceedings - 2nd IIAI International Conference on Advanced Applied Informatics, IIAI-AAI*, *3*, 108–111.

Wedpathak, G. S. (2017). Agriculture System Using Android Device-A Review Paper. *World Journal of Technology, Engineering and Research*, *2*(1), 227–231.

Wu, Y., & Chang, K. (2013). An empirical study of designing simplicity for mobile application interaction. *19th Americas Conference on Information Systems, AMCIS - Hyperconnected World: Anything, Anywhere, Anytime*, *1*, 331–338.

CURRICULUM VITAE

CURRICULUM VITAE

Name of candidate	:	KAKDE ASHUTOSH CHANDRAKANT
Date of Birth	:	05/01/1998
Nationality	:	INDIAN
Department	:	FARM MACHINERY & POWER ENGINEERING
Permanent address	:	Old Mhada Colony Q.No. LIG 80 Post Manas Mandir, Dist.- Wardha-442001
Mobile No	:	8308894659
Email id	:	ashukakde125@gmail.com
Title of thesis	:	DESIGN AND DEVELOPMENT OF ANDROID BASED MOBILE APPLICATION FOR FARM MECHANIZATION

Academic qualification

Course/ Degree	Name of college/Institute	University/ Board	Year of passing	Percentage (%)/ CGPA	Class/ Grade
SSC	Gandhi City Public School, Wardha	Central Board of Secondary Education	2013	7.2	Second class
HSC	New English Junior College, Wardha	RTMNU, Nagpur	2015	64.31%	Second class
B. Tech (Agricultural Engineering)	College of Agricultural Engineering and Technology, Jalgaon (Ja.)	Dr. Panjabrao Deshmukh Krishi Vidhyapeeth, Akola	2019	77.80%	First class

Place: Parbhani

Date : 16/ 02/2021

Signature of the candidate

Kakde Ashutosh Chandrakant