

**DEVELOPMENT OF APP FOR DESIGNING OF  
LOCATION SPECIFIC DRIP IRRIGATION SYSTEM**

**M. Tech. Thesis**

**by**

**Priyanka Netam**

**DEPARTMENT OF SOIL AND WATER ENGINEERING  
SWAMI VIVEKANAND COLLEGE OF AGRICULTURAL  
ENGINEERING AND TECHNOLOGY & RESEARCH STATION  
FACULTY OF AGRICULTURAL ENGINEERING  
INDIRA GANDHI KRISHI VISHWAVIDYALAYA  
RAIPUR (Chhattisgarh)  
2020**

**DEVELOPMENT OF APP FOR DESIGNING OF  
LOCATION SPECIFIC DRIP IRRIGATION SYSTEM**

**Thesis**

**Submitted to the**

**Indira Gandhi Krishi Vishwavidyalaya, Raipur**

**by**

**Priyanka Netam**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR  
THE DEGREE OF**

**Master of Technology**

**in**

**Agricultural Engineering**

**(SOIL AND WATER ENGINEERING)**

College ID-220118037

University ID-20131418310


**2020**

**// CERTIFICATE //**

This is to certify that the thesis Viva-voce in respect of **Miss. Priyanka Netam** student of **M. Tech. (Agril. Engg.)**, has been conducted under the chairmanship of Head of the Department/ Dean (in case of out campii) along with Advisory Committee on **“Development of App for Designing of Location Specific Drip Irrigation System”**. The necessary corrections have also been made as per comments/ suggestion made by the Advisory Committee and Head of the Department/ Dean.


Two copies of her thesis (in final bound form) are being forwarded separately to the Dean through Head of Department for its onward transmission to the Vishwavidyalaya.

Date: 02/11/2020

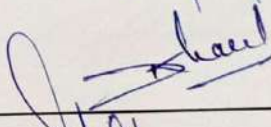
  
Signature of Major Advisor  
Name. Dr. Narendra Agrawal

**THESIS APPROVED BY THE STUDENT'S ADVISORY COMMITTEE**

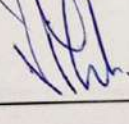
Chairman (Dr. Narendra Agrawal)

  
\_\_\_\_\_

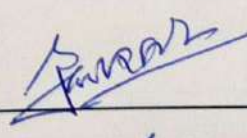
Member (Dr. Dhiraj Khalkho)

  
\_\_\_\_\_

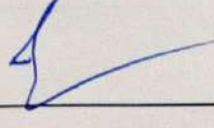
Member (Dr. V. M. Victor)

  
\_\_\_\_\_

Member (Dr. Samir Kumar Tamrakar)

  
\_\_\_\_\_

Member (Dr. M. P. Tripathi)

  
\_\_\_\_\_

## CERTIFICATE – II

This is to certify that the thesis entitled “**Development of App for Designing of Location Specific Drip Irrigation System**” submitted by **Priyanka Netam** to the Indira Gandhi Krishi Vishwavidyalaya, Raipur, in partial fulfillment of the requirements for the degree of **Master of Technology in Agricultural Engineering** in the Department of **Soil and Water Engineering** has been approved by the external evaluator and Student's Advisory Committee after oral examination, under the chairmanship of Head of the Department/Dean (in case of out campii).



Signature of Head of the Department/  
Dean (in case of out campii)

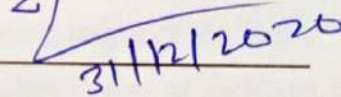
(Name *Dr. M. P. Tripathi*)

Date: *02/11/2020*

**Major Advisor**



**Faculty Dean**



*31/12/2020*

**Approved/Not approved**

**Director of Instructions**

## **ACKNOWLEDGEMENT**

*I wish to express my deep sense of respect and indebtedness to my Major Adviser Dr. Narendra Agrawal , Assistant Professor of Soil and Water Engineering, Faculty of Agricultural Engineering, IGKV, Raipur for his valuable, talented, inspiring, constructive criticism, and ceaseless encouragement provided during the entire project work. Indeed, the words at my command are not adequate, either in form of spirit or strength, to convey the depth of my feelings of gratitude for his most valuable guidance, suggestions, useful criticism and whole hearted support during the study.*

*I have a great pleasure in expressing my sincere thanks to my Advisory Committee Dr. Dhiraj Khalkho, Dr. V. M. Victor, Dr. Samir Kumar Tamrakar, and Dr. M. P. Tripathi for their priceless guidance, worthy suggestions and constant encouragement throughout the project.*

*I am very thankful to Dr. M. P. Tripathi, Dean, SVCAET & RS, IGKV, Raipur for his constant encouragement during project completion. I like to express my sincere thanks to Dr. M. P. Tripathi Head of the Department of Soil and Water Engineering, Dr. A. K. Dave, Head of the Department of Farm Machinery and Power Engineering and Dr. M. P. Tripathi Head of the Department of the Agricultural Processing and Food Engineering for their kind support and help at various stages of the study.*

*I am also thankful to faculty members of the Department of soil and Water Engineering Dr. M. P. Tripathi (Head of department), Dr. V.K. Pandey, Dr. R. K. Sahu (Professor), Dr. Dhiraj Khalkho (Associate Professor), Dr. Jitendra Sinha (Associate Professor) and Dr. Prafull Katre (Assistant Professor) including for their support and co-operation during the course of study.*

*I am especially thankful to Er. Visharad Diwakar (student & App developer), Er. Gajendra Sahu (student) and Er. Kuldeep Kashyap (student) to help to Develop Android App.*

*I am thankful to all the technical and clerical staff members of Dept. of Soil and Water Engg. and SVCAET & RS, Faculty of Agricultural Engineering for their kind support and help during entire study.*

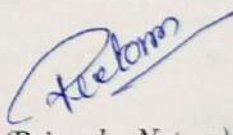
*I avail this pleasant opportunity to express my sincere thanks to my all senior, Er. Shrutika Verma, Er. Manju Dhrow, Er. Vijay Kumar, Er. Amrit Ekka, Er. Shudhir Verma and my batch mate Er. Aekesh Kumar, Er. Shreeya Baghel, Er. Anupama Lakra, Er. Divyanshu Gangeer, Er. Harsha Markam, Er. Kamal Kant Sahu, Er. Sandeep Kumar Sing, Er. Sakshi Thakur Er. Nomita Varma, Er. Tripti Temariya, Er. Sonal Sharma, Director of Instruction, IGKV, Raipur for providing necessary facilities for conducting the project.*

*Last but not the least, words run short to express my heartfelt gratitude to my beloved and my strength mother Smt. Maaya Netam and father Shri. Laxminath Netam for their boundless, generosity, inspiration, encouragements, everlasting blessing and abundant love for me for successful completion of this achievement. It is very difficult for me to express my feelings and gratitude for love and encouragement by my beloved brother and my elder sister.*

*All my sincere gratitude goes to almighty "The God" for his unfailing mercies over my life he has been kind to me.*

Date: 02/11/2020

Place: IGKV, Raipur

  
(Priyanka Netam)

## TABLE OF CONTENTS

| Chapter    | Title  | Page  |
|------------|--|-------|
|            | <b>ACKNOWLEDGEMENT</b>   | i     |
|            | <b>TABLE OF CONTENTS</b>   | iii   |
|            | <b>LIST OF TABLES</b>  | vii   |
|            | <b>LIST OF FIGURES</b>   | viii  |
|            | <b>LIST OF NOTATIONS / SYMBOLS</b>                                 | ix    |
|            | <b>LIST OF ABBREVIATIONS</b>                                       | x     |
|            | <b>ABSTRACT</b>  | xi    |
| <b>I</b>   | <b>INTRODUCTION</b>  | 1-4   |
|            | 1.1 Drip irrigation system design                                  | 2     |
|            | 1.2 Necessity of App development                                   | 3     |
| <b>II</b>  | <b>REVIEW OF LITRATURATURE</b>                                     | 5-14  |
|            | 2.1 Drip irrigation designing parameters                           | 5     |
|            | 2.2 Water requirement, crop coefficient and pan coefficient        | 5     |
|            | 2.3 Design of Laterals, Sub-main and Main                          | 8     |
|            | 2.4 Software's related to irrigation system design                 | 10    |
|            | 2.5 Android application (App) in Agriculture and Irrigation System | 12    |
| <b>III</b> | <b>MATERIALS AND METHODS</b>                                       | 15-37 |
|            | 3.1 Designing of Drip Irrigation system                            | 15    |
|            | 3.1.1 Steps involved in the designing of drip system of irrigation | 16    |
|            | 3.1.1.1 Designing inputs   | 17    |
|            | 3.1.1.2 Survey or field information                                | 17    |

---

|  |    |
|--|----|
| 3.1.1.3 Water Requirement (WR)                       | 18 |
| 3.1.1.4 Selection of drippers and number of plant    | 19 |
| 3.1.1.5 Water application rate and irrigation timing | 20 |
| 3.1.1.6 Selection and design of lateral              | 20 |
| 3.1.1.6.1 Head loss calculation in lateral           | 21 |
| 3.1.1.7 Selection and design of sub-main             | 22 |
| 3.1.1.8 Selection and design of main                 | 23 |
| 3.1.1.9 Total head -loss                             | 24 |
| 3.1.1.10 Selection of pump                           | 25 |
| 3.1.1.10.1 HP of pump                                | 25 |
| 3.1.1.10.2 Capacity of filter                        | 25 |
| 3.2 App Development Process                          | 26 |
| 3.2.1 System analysis                                | 26 |
| 3.2.2 Feasibility/ Probability analysis              | 27 |
| 3.2.2.1 Technology feasibility                       | 27 |
| 3.2.2.2 Operational feasibility                      | 28 |
| 3.2.2.3 Economic feasibility                         | 28 |
| 3.2.3 Requirement Analysis                           | 28 |
| 3.2.4 Coding   | 29 |
| 3.2.4.1 Strings. xml. file                           | 29 |
| 3.2.5 Testing  | 30 |
| 3.2.5.1 Unit testing                                 | 30 |
| 3.2.5.2 Integration testing                          | 30 |

---

---

|   |              |
|---|--------------|
| 3.2.5.3 Program testing   | 31           |
| 3.2.5.4 System testing  | 31           |
| 3.2.6 Implementation  | 31           |
| 3.2.6.1 Flow diagram of Drip Irrigation App                           | 31           |
| 3.3 Proposed software tool  | 32           |
| 3.3.1 Android   | 32           |
| 3.3.2 Application   | 33           |
| 3.4 Software used for the development of App                          | 34           |
| 3.4.1 Android Studio  | 34           |
| 3.4.2 Workflow of Android App   | 35           |
| 3.5.3 Java Script Object Notation (JSON)                              | 36           |
| <b>IV RESULTS AND DISCUSSION</b>                                      | <b>38-55</b> |
| 4 App include following step for the design of drip irrigation system | 38           |
| 4.1 User information activity   | 38           |
| 4.2 Field information activity  | 38           |
| 4.3 Water requirement(WR)   | 40           |
| 4.4 Selection of drippers   | 40           |
| 4.5 Selection and design of lateral                                   | 42           |
| 4.6 Selection and design of sub main                                  | 44           |
| 4.7 Selection and design of main                                      | 46           |
| 4.8 Total head loss   | 48           |
| 4.9 Selection of pump and filter                                      | 48           |

---

---

|                                  |        |
|----------------------------------|--------|
| 4.10 Output                      | 50     |
| 4.11 Validation of developed App | 51     |
| <b>V SUMMARY AND CONCLUSIONS</b> | 56-58  |
| <b>REFERENCES</b>                | 59-61  |
| <b>APPENDICES</b>                | 62-104 |
| <b>RESUME</b>                    | 105    |

---

## LIST OF TABLES

| <b>Table</b> | <b>Title</b>               | <b>Page</b> |
|--------------|----------------------------|-------------|
| 3.1          | Value of pipe outlets      | 22          |
| 3.2          | Description of head losses | 24          |
| 3.3          | Filter capacity            | 25          |

## LIST OF FIGURES

| Figure | Title   | Page |
|--------|---|------|
| 3.1    | String.xml file   | 30   |
| 3.2    | Flow diagram of Drip Irrigation App                         | 32   |
| 3.3    | Work flow of App Development                                | 35   |
| 4.1    | User information  | 39   |
| 4.2    | User information with input                                 | 39   |
| 4.3    | Field information   | 39   |
| 4.4    | Field information with input                                | 39   |
| 4.5    | Water requirement   | 41   |
| 4.6    | Water requirement with input                                | 41   |
| 4.7    | Selection of dripper  | 41   |
| 4.8    | Selection of dripper with input                             | 41   |
| 4.9    | Selection of lateral  | 43   |
| 4.10   | Selection of lateral with input                             | 43   |
| 4.11   | Selection of lateral with input show (not acceptable case)  | 43   |
| 4.12   | Selection of sub main                                       | 45   |
| 4.13   | Selection of sub main with input                            | 45   |
| 4.14   | Selection of sub main with input show (not acceptable case) | 45   |
| 4.15   | Selection of main   | 47   |
| 4.16   | Selection of main with input                                | 47   |
| 4.17   | Selection of mainline with input show (not acceptable case) | 47   |
| 4.18   | Total head loss   | 49   |
| 4.19   | Total head loss with input                                  | 49   |
| 4.20   | Selection of pump   | 49   |
| 4.21   | Selection of pump with input                                | 49   |
| 4.22   | Output  | 50   |

## LIST OF NOTATIONS/ SYMBOLS

---

| Notations/ Symbols |   | Description                        |
|--------------------|---|------------------------------------|
| m                  | : | Meter                              |
| m <sup>2</sup>     | : | Meter square                       |
| %                  | : | Percent                            |
| mm                 | : | Millimeter                         |
| lph                | : | Liter per hour                     |
| lps                | : | Liter per second                   |
| hr                 | : | Hour                               |
| hp                 | : | Horse power                        |
| m <sup>3</sup> /hr | : | Meter cube per hour                |
| mm/day             | : | Millimeter per day                 |
| lit/d/pt           | : | Liter per day per plant            |
| ×                  | : | Multiplication                     |
| =                  | : | Equal                              |
| <                  | : | Less than                          |
| >                  | : | Greater than                       |
| ÷                  | : | Division                           |
| &                  | : | And                                |
| ( )                | : | Calculated expression inside first |
| a                  | : | Alfa                               |
| <i>et al.</i> ,    | : | And other                          |
| Fig.               | : | Figure                             |

---

## LIST OF ABBREVIATIONS

---

---

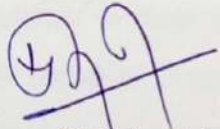
| <b>Abbreviations</b> | <b>Full forms</b>   |
|----------------------|---|
| Agril.               | : Agricultural  |
| Agril. Engg.         | : Agricultural Engineering  |
| Dept.                | : Department  |
| M .Tech              | : Master of Degree  |
| Engg.                | : Engineering   |
| FAE                  | : Faculty of Agriculture Engineering  |
| IGKV                 | : Indira Gandhi Kridhi Vishvidhyaly   |
| SVCAET&RS            | : Swami Vivekanand College of Agricultural<br>Engineering and technology & Research station |
| App                  | : Application   |
| PE                   | : Pan-Evaporation   |
| K <sub>C</sub>       | : Crop coefficient  |
| P <sub>C</sub>       | : Pan coefficient   |
| E <sub>u</sub>       | : Emission uniformity of drip   |
| S <sub>l</sub>       | : Lateral spacing   |
| S <sub>d</sub>       | : Dripper spacing   |
| WR                   | : Water Requirement   |
| HDPE                 | : High density polyethylene   |
| LDPE                 | : Low density polyethylene  |
| ET <sub>O</sub>      | : Evapotranspiration  |
| OS                   | : Operating system  |
| TV                   | : Television  |
| APK                  | : Android package   |
| SDK                  | : Software development kit  |
| IDE                  | : Instigated development environment  |
| JSON                 | : Java script object nation   |

---

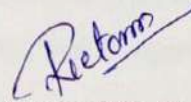
## THESIS ABSTRACT

---

- (a) Title of the Thesis: Development of App for Designing of Location Specific Drip Irrigation System
- (b) Full Name of the Student: Priyanka Netam
- (c) Major Subjects: Soil and Water Engineering
- (d) Name and Address of the Major Advisor: Dr. Narendra Agrawal  
Assistant Professor  
Department of Soil and Water Engineering,  
Swami Vivekanand College of Agriculture  
Engineering and Technology & Research  
Station, Faculty of Agricultural Engineering,  
IGKV, Raipur (CG)
- (e) Degree to be awarded: Master of Technology  
Agricultural Engineering  
(Soil and Water Engineering)



Signature of Major Advisor



Signature of the Student



Date : 02/11/2020

Signature of Head of the Department

---

## ABSTRACT

Drip irrigation is a system in which the water and plant nutrients applied by drippers at low pressure and frequent intervals into the plant root zone in a very precise and regulated manner. This system is also known as trickle irrigation. This is the latest technology of irrigation in which the water application rate is very low and

limited pressure i.e. drop by drop. Drip irrigation is an eco-friendly irrigation system which not only saves precious irrigation water but also increases productivity to the tune of 30-40 per cent over traditional methods of irrigation. This system is very popular in areas where, water scarcity and salt related problems.

The drip irrigation system mainly consists of filtration unit, fertigation unit, main line, sub-main line, laterals and drippers. Main lines are made of either high density polyethylene (HDPE) or rigid polyvinyl chloride (PVC) with 20 to 100 mm in diameters. These lines are used to carry the water from water source to the sub-main. Main line is attached to sub-main line through different PVC fittings. Sub-main is also made of PVC and supplies water to the laterals on one or both sides of it. Ball valves are provided in sub-mains to maintain the required pressure and flow in the pipe line. Flush valves are provided at the tail end of main and sub-main lines for cleaning and removing the dirt.

The main line delivers water to sub-main line and sub-main line to laterals. Laterals are either line source tubing or drip laterals and are made up of linear low density polyethylene (LLDPE), which is quite flexible and strong. These laterals are used to carry the water from sub-main lines to root zone of the plants through drippers. They are available in plain laterals and dripper inbuilt laterals with different spacing and sizing of 12mm to 20mm. Proper size of laterals are laid in open field and their spacing is based on the row to row distance of the crop.

Proper sizes of drippers are inbuilt into the laterals and the water from laterals spread into the plant root zone through the drippers. Soil, crop type and most importantly water requirement of crops are the major factors for deciding the spacing of drippers on the lateral pipe. Drippers are either pressure compensating or non pressure compensating depending upon topography of the field. These are well designed to provide precise amount of water to the plant as per their requirement. It consists of different parts to dissipate the pressure in the different pipes by means of a long flow path and to allow only a few liters of water per hour discharge with decreased water pressure. After that this water is spread into the soil profile by its normal movement.

In terms of water use and labour, the drip irrigation system is the most effective method. For efficient use of water and labour, drip irrigation systems should be designed, installed, and maintained properly. The design of drip irrigation system is very tedious job for any farmer or extension worker. Now a days various distributors and dealers of drip irrigation companies are available in the market and large number of farmers and they need to design and install many systems within a different places of their jurisdiction. But, in the design of drip irrigation system it is very difficult to calculate crop water requirement, size and length of different pipes, pressure losses in the various pipes, discharge variation along the laterals, size, discharge and number of drippers, horse power and discharge requirement of the pump set etc., manually. Also there are numerous software's available to deal with the design related issues but the drawback is the hardware requirement for the same.

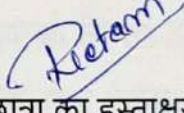
Presently with the advancement in Information and Communications Technology (ICT) and awareness among the users, android platform based mobile is being readily used and adopted by the farmers, extension workers, students and researchers. The present study was conducted with the objective to develop an Android Based App for design of drip irrigation system. This App was developed with the help of android studio software and java programming language. This App will provide proper platform that can help farmers, industrialist, marketing executive, extension specialist and researchers who design drip irrigation system. It can be extensively used by farmers or extension worker to overcome the issues of faculty design. It will also assist in the design of a system for various locations and crops. This App tackles the issue of accessibility of information and will customize the design based on the data input. The App would be a robust design tool for both small and large farmers and it has limited storage space and operates without internet access. The App has been tested with past researcher's design. The results obtained using developed App was compared with the results of researcher's design and the results were found satisfactory.

## शोध सारांश

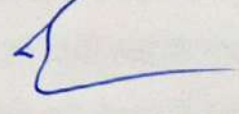
---

|                                  |  |
|----------------------------------|--|
| थीसिस का शीर्षक:                 | स्थान विशिष्ट ड्रिप सिंचाई प्रणाली के डिजाइन के लिए ऐप का विकास  |
| छात्र का पूरा नाम:               | प्रियंका नेताम   |
| प्रमुख विषय:                     | मृदा और जल अभियांत्रिकी  |
| प्रमुख सलाहकार का नाम और पता:    | डॉ. नरेंद्र अग्रवाल<br>सहायक प्रोफेसर<br>मृदा एवं जल अभियांत्रिकी विभाग कृषि<br>अभियांत्रिकी संकाय,<br>इंदिरा गांधी कृषि विश्वविद्यालय, रायपुर |
| डिग्री से सम्मानित किया जाने है: | कृषि अभियांत्रिकी में स्नातकोत्तर<br>(मास्टर ऑफ टैकनोलजी)  |

  
प्रमुख सलाहकार का हस्ताक्षर:

  
छात्रा का हस्ताक्षर

दिनांक : 02/11/2020

  
विभाग के प्रमुख का हस्ताक्षर

---

## सारांश

ड्रिप इरीगेशन एक ऐसी प्रणाली है जिसमें पानी और पौधों के पोषक तत्व ड्रिपर्स द्वारा कम दबाव और प्लांट रूट ज़ोन में लगातार अंतराल पर बहुत सटीक और विनियमित तरीके से

लगाए जाते हैं। इस प्रणाली को ट्रिकल सिंचाई के रूप में भी जाना जाता है। यह सिंचाई की नवीनतम तकनीक है जिसमें पानी के आवेदन की दर बहुत कम है और सीमित दबाव है यानी बूंद-बूंद करके। ड्रिप सिंचाई एक इको-फ्रेंडली सिंचाई प्रणाली है, जो न केवल कीमती सिंचाई के पानी को बचाती है, बल्कि सिंचाई के पारंपरिक तरीकों पर उत्पादकता को 30-40 प्रतिशत तक बढ़ाती है। यह प्रणाली उन क्षेत्रों में बहुत लोकप्रिय है, जहां पानी की कमी और नमक से संबंधित समस्याएं हैं।

ड्रिप सिंचाई प्रणाली में मुख्य रूप से निस्पंदन यूनिट, फर्टिगेशन यूनिट, मेन लाइन, सब-मेन लाइन, लेटरल और ड्रिपर्स शामिल हैं। मुख्य लाइनें व्यास में 20 से 100 मिमी के साथ या तो उच्च घनत्व पॉलीथीन (एच.डी.पी.ई.) या कठोर पॉलीविनाइल क्लोराइड (पी.वी.सी.) से बनी होती हैं। इन लाइनों का उपयोग जल स्रोत से पानी को उप-मुख्य तक ले जाने के लिए किया जाता है। विभिन्न पी.वी.सी. फिटिंग्स के माध्यम से मेन लाइन सब-मेन लाइन से जुड़ी हुई है। उप-मुख्य भी पी.वी.सी. से बना है और इसके एक या दोनों किनारों पर पार्श्व को पानी की आपूर्ति करता है। पाइप वाल्व को आवश्यक दबाव बनाए रखने और पाइप लाइन में प्रवाह करने के लिए उप-मेन में प्रदान किया जाता है। गंदगी को साफ करने और हटाने के लिए मुख्य और उप-मुख्य लाइनों के पूंछ के छोर पर फ्लश वाल्व प्रदान किए जाते हैं।

मुख्य लाइन पानी को सब-मेन लाइन और सब-मेन लाइन को लेटरल तक पहुंचाती है। पार्श्व या तो लाइन स्रोत ट्यूबिंग या ड्रिप लेटरल हैं और रैखिक कम घनत्व वाले पॉलीथीन (एलएलडीपीई) से बने होते हैं, जो काफी लचीला और मजबूत होता है। इन पार्श्वों का उपयोग उप-मुख्य लाइनों से पानी को ड्रिपर्स के माध्यम से पौधों के जड़ क्षेत्र तक ले जाने के लिए किया जाता है। वे अलग-अलग रिक्ति और आकार के साथ 12 मिमी से 20 मिमी तक के पार्श्व मैदानों और ड्रिपर इनबिल्ट लेटर्स में उपलब्ध हैं। पार्श्व के उचित आकार को खुले मैदान में रखा गया है और उनकी दूरी फसल की पंक्ति से पंक्ति की दूरी पर आधारित है।

ड्रिपर्स के उचित आकार पार्श्वों में इनबिल्ट होते हैं और पार्श्वों से पानी ड्रिपर्स के माध्यम से प्लांट रूट ज़ोन में फैलता है। मृदा, फसल के प्रकार और सबसे महत्वपूर्ण रूप से फसलों की पानी की आवश्यकता पार्श्व पाइप पर ड्रिपर्स के अंतर को तय करने के लिए प्रमुख कारक हैं। क्षेत्र की स्थलाकृति के आधार पर ड्रिपर्स या तो क्षतिपूर्ति या गैर-दबाव की भरपाई कर रहे

हैं। ये अच्छी तरह से उनकी आवश्यकता के अनुसार पौधे को पानी की सटीक मात्रा प्रदान करने के लिए डिज़ाइन किए गए हैं। इसमें एक लंबे प्रवाह पथ के माध्यम से अलग-अलग पाइपों में दबाव को अलग करने और कम पानी के दबाव के साथ प्रति घंटे केवल कुछ लीटर पानी के निर्वहन की अनुमति देने के लिए अलग-अलग हिस्से होते हैं। उसके बाद यह पानी अपने सामान्य आवागमन के द्वारा मिट्टी की प्रोफाइल में फैल जाता है।

पानी के उपयोग और श्रम के संदर्भ में, ड्रिप सिंचाई प्रणाली सबसे प्रभावी तरीका है। पानी और श्रम के कुशल उपयोग के लिए, ड्रिप सिंचाई प्रणाली को ठीक से डिजाइन, स्थापित और रखरखाव किया जाना चाहिए। ड्रिप सिंचाई प्रणाली का डिज़ाइन किसी भी किसान या विस्तार कार्यकर्ता के लिए बहुत थकाऊ काम है। अब एक दिन ड्रिप सिंचाई कंपनियों के विभिन्न वितरक और डीलर बाजार और बड़ी संख्या में किसानों के लिए उपलब्ध हैं और उन्हें अपने अधिकार क्षेत्र के विभिन्न स्थानों के भीतर कई प्रणालियों को डिजाइन और स्थापित करने की आवश्यकता है। लेकिन, ड्रिप सिंचाई प्रणाली के डिजाइन में फसल की पानी की आवश्यकता, आकार और विभिन्न पाइपों की लंबाई, विभिन्न पाइपों में दबाव के नुकसान, पार्श्व के साथ निर्वहन भिन्नता, आकार, निर्वहन और ड्रिपर्स, घोड़े की शक्ति और संख्या की गणना करना बहुत मुश्किल है मैन्युअल रूप से पंप सेट आदि की डिस्चार्ज आवश्यकता। इसके अलावा डिजाइन से संबंधित मुद्दों से निपटने के लिए कई सॉफ्टवेयर उपलब्ध हैं लेकिन खामी उसी के लिए हार्डवेयर की आवश्यकता है।

वर्तमान में सूचना और संचार प्रौद्योगिकी (आईसीटी) में उन्नति और उपयोगकर्ताओं के बीच जागरूकता के साथ, एंड्रॉइड प्लेटफॉर्म आधारित मोबाइल का उपयोग किसानों, विस्तार श्रमिकों, छात्रों और शोधकर्ताओं द्वारा आसानी से किया और अपनाया जा रहा है। ड्रिप सिंचाई प्रणाली के डिजाइन के लिए एंड्रॉइड आधारित ऐप विकसित करने के उद्देश्य से वर्तमान अध्ययन किया गया था। इस ऐप को एंड्रॉइड स्टूडियो सॉफ्टवेयर और जावा प्रोग्रामिंग लैंग्वेज की मदद से विकसित किया गया था। यह ऐप उचित मंच प्रदान करेगा जो किसानों, उद्योगपति, विपणन कार्यकारी, विस्तार विशेषज्ञ और ड्रिप सिंचाई प्रणाली को डिजाइन करने वाले शोधकर्ताओं की मदद कर सकता है। संकाय डिजाइन के मुद्दों को दूर करने के लिए किसानों या विस्तार कार्यकर्ता द्वारा इसका बड़े पैमाने पर उपयोग किया जा सकता है। यह विभिन्न

स्थानों और फसलों के लिए एक प्रणाली के डिजाइन में भी सहायता करेगा। यह ऐप सूचना की पहुंच की समस्या से निपटता है और डेटा इनपुट के आधार पर डिज़ाइन को कस्टमाइज़ करेगा। ऐप छोटे और बड़े दोनों किसानों के लिए एक मजबूत डिजाइन उपकरण होगा और इसमें सीमित भंडारण स्थान है और यह इंटरनेट एक्सेस के बिना काम करता है। एप्लिकेशन को पिछले शोधकर्ता के डिजाइन के साथ परीक्षण किया गया है। विकसित ऐप का उपयोग करके प्राप्त परिणामों की तुलना शोधकर्ता के डिजाइन के परिणामों से की गई और परिणाम संतोषजनक पाए गए।

# CHAPTER-I

## INTRODUCTION

---

Present world is in information and technology era. It is an era of accelerated technological progress. These technologies abruptly changes today's society. Technology has become the part of our daily life style. Today's most common way to transmit our voice, data, and services in the world with help of a mobile phone has become a common practice. Mobile phones make everyday life easy and convenient. It is not only for the communication of information but it creates miraculous changes in working atmosphere. Presently the whole world becomes closure and easy to accessible.

Now a day's mobile phones are very common and used by everyone. The development of mobile applications has increased the usability of mobiles beyond voice and text communications. App is a computer programming design to run on different electronic devices such as smart phone, laptop, tablet and computer. In few years ago mobile apps was developed mainly for entertainment and communication purpose only, but now trend has been change and these are focused on other sector also like economical based, daily routine based, activities based and agriculture or rural based mobile application. Mobile Apps for agriculture sector and rural development is one the most important field that focuses on the enhancement of both the fields.

Mobile application participates effectively in the development of agriculture and rural development by providing accurate and immediate information to the farmers and users. In present, lots of mobile app has been developed in the field of agriculture for the farmers. These mobile applications are helping the farmers to improve the crop productivity. There are many advantages of mobile apps like enhancement in own skills, reducing dependency in other technical persons, sell out their produce in open market etc. The most important in the mobile apps is that they are user friendly; one can easily operate it and get benefitted from it.

### **1.1 Drip irrigation system design**

Water is the most vital and important resource in agriculture and has substantial share in providing stability to all food grain production. Agriculture consumes more than 70% of global water and 90% water is consumed in developing countries.

The unexpected climatic change in today's world will have strongly marked adverse effects on the total availability of water and its quality. The apprehended major negative effect will be the decline in the quantum of rainfall, reduction in ground water recharge, high evaporation rate and rapidly melting of glaciers due to rise in temperature and increased water pollution. These other changes will have a lot bearing on limited water availability of acceptable quality. Under this circumstance, all need to have appropriate water management plan and adopt modern techniques of irrigation to increase the crop yield and water use efficiency.

Drip irrigation method is very popular in different countries and is instrumental in enhanced crop yield and assist in reducing water resource utilization. The drip irrigation system is the method of applying control and uniform amount of water directly in and around the root zone of the plant. Drip irrigation system should be incorporated whenever possible in scientific water management farm activities. The drip system maintains equilibrium between crop evapo-transpiration and water supplied over the crop growth period.

The limited ability of drip irrigation system operating over this short time period needs careful attention to be paid to estimating the crop water or calculate the status of soil water in the plant root zone (Michael and Ojha 2006). The drip irrigation system contains drippers, laterals, sub-main line, main line, which supplies water from the source to the plant root zone. The main line delivers the water to sub-main line and sub-main lines to the laterals. The emitters are connected to the laterals and these emitters deliver the water to the root zone of the plants. Laterals are usually made of low density polyethylene (LDPE) with diameters ranging from 12 to 20 mm, offering better flushing characteristics, simple installation and low maintenance. The main lines and sub-main lines are either high density polyethylene (HDPE) or rigid

polyvinyl chloride (PVC) with range of 20 to 100 mm in range. Emitters are plant point water sources and designed to provide a limited and equal amount of water as per requirement of plant. Emitters are of many kinds, such as nozzles, porous pipes, micro-tubes etc., to dissipate the pressure through a nozzle or a long flow path in the pipe distribution network and thus reduce the water pressure to allow just a few liters per hour to discharge. Then water is dispersed through the profile of the soil by its normal movement (Reddy *et al.*, 2017).

The overall capacity of the drip irrigation system is determined by the irrigation requirement, spacing, soil moisture characteristics, evapo-transpiration rate for the crop and area to be irrigated. As it is, difficult to calculate manually crop water requirement, water application rate, irrigation timing, various pipe discharge, pressure losses, size of pipes, and discharge requirement of the pump etc. The design of drip irrigation system is a very tough job and it requires skilled & trained persons. There for to overcome the issues of faulty design of drip irrigation system, proper design of the system is needed.

## **1.2 Necessity of App Development**

There are numerous software's available to deal with the design related issues but the drawback is that this computer related software's are not easily available to the users. Presently with the development of Information and Communications Technology (ICT) awareness regarding Smart Phones is increased. Therefore, the android based mobile applications is being readily used and easily adopted by the farmers, students and researchers. The aim of this project is to develop android based app for the designing of drip irrigation system for the different crops and different locations. This android app is an open source mobile operating system by Google and it can be easily installed by the farmers or extension workers through Google Play Store.

Most of the android application is written in programming language C++, C, Java, Kotlin etc, but in the present study for the design of drip irrigation system, Android Studio Software is used for the development of Mobile App. This App will

easily provide useful and accurate information about the design and optimization of drip irrigation area. It also deals with the issues of data availability and will customize the design based on the data input. In the total scenario, this App will be a comprehensive design tool both for small as well as big farmers. Therefore, keeping the above facts in the mind the present study is proposed with the following objectives.

### **Objectives**

1. To study the different designing steps of drip irrigation system based on available data.
2. To work out the appropriate design algorithms for development of android App for site specific drip irrigation system.
3. To validate developed drip irrigation system designing App.

## **CHAPTER-II**

### **REVIEW OF LITERATURE**

---

The chapter deals with the literature review relevant to the present research. A brief review pertaining to research work conducted on different supporting media for drip irrigation system design. Major fields related to drip irrigation system design include designing parameters, water requirement, crop factor, pan factor, evaporation, evapo-transpiration, design of laterals, sub-main and main, development of software related to designing of drip system and android based mobile applications is present in this chapter.

#### **2.1 Drip irrigation designing parameters**

Kumar and Palanisami (2010) studied the effect of drip irrigation on the farming system in terms of cropping patterns, resource usage and crop yield. The drip irrigation system has been found to have a major effect on resource savings, cultivation prices, crop yield and farm profitability. Therefore, the policy should be focused on promoting drip irrigation in the regions where there are concerns regarding water and labor shortages and where there is a transition towards larger crops.

Mane *et al.* (2019) described the different steps involved in the designing of drip irrigation system. Like type of crop, selection of dripper (emitter), selection and design of lateral, selection and design of sub main, selection and design of main line selection of pump.

#### **2.2 Water Requirement, crop-coefficient and pan- coefficient**

Sentelhas and Folegatti (2003) reported that the regression analysis contrasted the relationship between  $ET_0$  measured in a weighting lysimeter and FCA- measured in a class-A Pan compared by regression analysis. The same routine was adopted to evaluate  $ET_0$  estimates with different  $K_p$  values. They found the all methods to estimated  $K_p$  did not predict it well, with low correlation ( $R^2 < 0.2$ ), which resulted

in estimates of  $ET_0$  with high dispersion ( $R^2 < 0.8$ ). The best  $K_p$  methods to estimate  $ET_0$  were Pereira et al. (1995) and Cuenca (1989), both presenting high efficiency. The used of an arbitrary and constant  $K_p$  (0.71) to estimate  $ET_0$ , produced the same precision and accuracy as the estimates of  $K_p$  based on Pereira and Cuenca methods. This fixed value is a practical and simple option to convert ECA into  $ET_0$ , but this value must be calibrated for each place under different climatic conditions.

Kumar *et al.* (2011) studied the effect of ET and estimation of evapo-transpiration of mustard crop. They adopted Penman-Montheith equation because it is most reliable and accurate method for computing reference evapo-transpiration. They reported in their study that ET through FAO-56 showed 16.80% evapo-transpiration and modified FAO-56 showed 8.33% evapo-transpiration. They developed crop coefficient curve with the help of ratio of crop evapo-transpiration and reference evapo-transpiration with respect to time. Therefore, they concluded that the performance of modified FAO-56 curve method was found better than FAO-56 curve method for estimation of mustard evapo-transpiration.

Pradhan *et al.* (2013) observed that the  $K_p$  (Pan coefficient) showed difference during the year with value ranging between 0.72-.93 being higher in rainy and winter season and lower during the summer season. For the quantitative and qualitative estimation of  $K_p$  for the semi arid zone they used five methods .In this experiment it is recommended that temporal variation in  $K_p$  should be estimated for each station for computing representative  $ET_0$  using the Snyder method gave close agreement with the FAO-56 Pan-method (Panman-Montheith). So Snyder method is best for  $K_p$  calculation in semi- arid regions and this research according Snyder equation recommended for  $K_p$  value in semi-arid zone.

Gupta *et al.* (2017) conducted an experiment to estimate water productivity of mustard crop during Rabi season 2013-14 and 2014-15 using the lysimeter. The experiment was conducted in 1000 m<sup>2</sup> plot area. The research result observed in 2013-

14 for crop coefficient ( $K_c$ ) in initial stage was 0.3-0.72, at mid stage was 1.02 and at last stage was 0.5, during Rabi season. While the basal crop coefficient  $K_{cb}$  in the dual  $K_c$  was 0.19, 0.55, 0.91 and 0.24 for the four phases during the Rabi season. The single  $K_{cb}$  coefficient ( $K_c$ ) was 0.17, 0.55, 0.91 and .024 in 2014-15 and the dual condition  $K_c$  value of  $K_{cb}$  was 0.17, 0.46, 0.19 and 0.23 for four phases respectively. Water productivity was found to be 14.9 kg/ha-m, corresponding to grain yield of 2.34 t/ha with 157 mm total irrigation water applied in 2013-14 during the Rabi season and water productivity was 15.4 kg/ha-m with 2.89 t/ha grain yield in 2014-15. The coefficient was simpler and less cumbersome than the coefficient of basal crop  $K_c$  and the component of soil evaporation in the dual crop coefficient.

Silva *et al.* (2017) conducted a research with the aim to estimate the local crop coefficient ( $K_c$ ) for corn crop. The analysis of the research result showed that the  $K_c$  value as recommended by FAO for each studied town and corn cycle phase were different from each other.  $K_c$  value of Corn crop estimated for initial stage was 0.073 to 0.077, for the intermediate stage ranged between 1.00 to 1.31 and for final stage it ranged between 0.24 to 0.45 respectively.

Srinivas and Tiwari (2018) determined the crop coefficient ( $K_c$ ) and crop evapo-transpiration ( $ET_c$ ) for different stage of green gram crop. Crop coefficient was obtained from experiment for each growth stage as the ratio of evapo-transpiration ( $ET_c$ ) and reference crop evapo-transpiration ( $ET_0$ ) and three dimensional area (1m×1m×1.2m) used to determine the daily crop evapo-transpiration ( $ET_c$ ) of green gram crop. The crop ( $ET_c$ ) was determined by water balance equation and ( $ET_0$ ) was determined by Penman-Monteith equation. The  $ET_c$  and  $ET_0$  ratio was taken as  $K_c$  of crop at different crop growth stage and  $ET_c$  taken at time step of seasonal green gram crop. The length of plant growth stage identified as, 15, 30, 45, and 60 days for initial, mid and end growth stage. The result of  $ET_c$  for green gram was found to be 210.54 in initial stage, 36.0-86.78 in mid stage and the 16.52 in end stage of crop respectively.  $K_c$  value is estimated to be 0.34-0.98 initial stage, 1.13 mid stages and 0.17 end stage

of the crop. The Kc value very high in mid stage compares to initial or end stage of crop growth.

### **2.3 Design of Lateral, Sub-main and Main**

Amer and Gomaa (2003) determined uniformity of flow rate along drip lateral line depending on the average of absolute deviation and (CV) coefficient of variation of dripper flow rate. A field experiment was conducted for (80 m) of lateral line length (inside diameter 16 mm) with 160 dripper. They found that difference, Difference between measured and determined friction loss was less than ( $< 3.5\%$ ) when the equivalent length coefficient ( $a$ ) was determined as 1.26. It was reduced to 1% when ( $a$ ) was integrated to be 1.36. High correlation ( $r^2=0.97$ ) between field coefficient of variation and determined coefficient of variation with a high 3% difference for determined CV when turbo dripper CV(m) equals 6.5% was used. Emission uniformity was determined as 83.66, 87.62, and 78.94% for zero, (+2% to -2%) slopes of 80 m lateral, respectively, when using turbo dripper with manufacturing CV equals 6.5%.

Mohanty *et al.* (2015) conducted a study for the impact of five distinct single and double inlet lateral connect with three distinct diameter sub-main pipe sizes 40 mm, 50mm and 63mm on water use efficiency (WUE), yield and growth characteristic of drip irrigated Brinjal crop for the year 2011, 2012 and 2013. For the selected plants, biometric characteristics of the Brinjal crop, such as plant height, base circumference, number of branches per plant, number of leaves per plant, base root diameter, leaf area, root volume, fruit length and weight, etc., was observed under different treatments. This performance was also expressed in the case of yield and water efficiency in the cultivation of the Brinjal crop. Maximum yield (399.48 q/ha) and water efficiency (880.58 kg/ha-cm) were observed in the case of a double inlet system with a diameter of two sub-mains of 63mm and a lateral link to both sides of the sub-mains. Similarly, it is known that the growth parameters and performance of the system increase when single inlet systems with laterals laid on one side or both sides of the sub-main are converted to the corresponding double inlet systems by looping

the laterals (L1 to L2 and L3 to L4), which is very simple to achieve just by adding some minimal cost to in-line laterals and connectors for achieving higher yield and water use efficiency.

Kumar *et al.* (2016) stated that drip irrigation is a form of irrigation to conserve water and fertilizer by allowing water to drip slowly through a network of drippers, valves, pipes, and pumps to the plant root, either to the soil surface or directly to the root region. In the study the overall maximum crop yield of Brinjal crop reported as 900 gm/plant and the minimum yield are 600 gm/ plant and the cumulative yield is 52270 gm. The highest efficiency of field water usage ( $6148.31 \text{ kg ha}^{-1} \text{ cm}^{-1}$ ) was found at 65 percent irrigation level among the drip irrigation levels, suggested a comparatively more effective use of irrigation water with the probability of 35 percent water saving through the adoption of 1.58 liter/day/plant.

Reddy *et al.* (2017) studied to assess and evaluate the performance of the drip irrigation system. The study was conducted to evaluate unequal distribution of water discharge of dripper in drip system and to assess the maintenance of installed system. Farmers mainly facing installation problem of drip system in the sloppy land. The results of the study were found variation of drip system ranges from (3- 8%) and was accepted (as it was less than  $< 10\%$ ) and in some cases it ranged from (25- 44%), which was not acceptable (as  $< 20\%$  is not acceptable).

Kumar *et al.* (2019) developed the design for the drip irrigation system, which consisted of 12 different crops. The design was based on the friction-loss method and was calculated using the Hazen-Williams formula. The 2 lph emitter's could be used for all of the crops, except tomato and mango crop, where 4 lph and 14 lph emitters were recommended, respectively. The diameter of the laterals used for all of the crops was calculated and was found to be 12 mm, except for the tomato crop, where 16 mm diameter laterals pipe was recommended. The diameter of the entire sub-main pipe was calculated and was found to be 50 mm. The mains diameter was recommended as

50, 75 and 90 mm. The efficiency of the water pump was assumed as 70% and using the values of the total dynamic head and the discharge at the final main line, the size of the pump was calculated and was found to be 22.5 hp. The next higher size of the pump available in the market was to be installed.

#### **2.4 Software's related to irrigation system design**

Perold (1977) designed a micro irrigation system by using computer program and graphical design technique in his study. In this design, adjustable micro tubes were used as feeders and the flow rate was assumed to be constant. He reported that simulation trials should be plotted to find out an optimal size for each laterals.

Narayanan *et al.* (2002) developed a computerized model to design and economically optimize the above-ground drip irrigation systems for carrots, onions, cabbage and sweet corn. The complete enumeration method was used to find the optimal lateral, sub-main and main line sizes, pipelines using hydraulic principles, details on fields, crops, water and soils, and the characteristics and costs of components of drip irrigation. Economic analysis was carried out for fields of 0.4, 0.8, 1.6, and 4.0 ha having one, two, or four irrigated zones and two potential well sites. In the design and optimization of drip irrigation systems for small areas, this model offers valuable details.

Rares (2009) reported that supplies of good water quality are diminishing day by day. Methods to increase water use performance in agriculture need to be sought. The most effective approach used in agriculture to mitigate decline in available water is drip irrigation. There was a template for the irrigation group with Hydrosols Irrigation Planning Tools. The output of such basic hydraulic data is a quick and simple calculation tool. The use of Hydrosol enables the manufacturer, dealer or end-user to determine the efficiency of in-field components of the micro-irrigation system, such as drip laterals and micro-sprinklers, sub-mains and manifolds, main lines, measurement of energy valves, etc.

Bombale *et al.* (2011) developed drip irrigation design software that offers connectivity at all stages of the design process and a solution based on individuals' own skills and the software's knowledge base. Not only the person fulfill his requirements, but his design can also be contrasted with many other alternative designs resulting from the application of the software with numerous possible inputs. This software can include tools to assist producers, industrialists, marketing executives, extension specialists, and researchers in the design of drip irrigation systems.

Reddy *et al.* (2017) developed software for the design of drip irrigation system with visual basic software based on all designing parameters. A sample run was made with assumed data for different crops. For Apple crop with a spacing of 15ft x 15ft requires 1 hp pump with total head of 38.5m should be operated for 2.5 hrs, flow in main line is 2.116402 lph costs about 97009 Rs./ ha. For Banana crop with spacing of 10ft x 10ft requires 3 hp pump with total head of 38.5m should be operated for 1.25 hrs, flow in main line is 4.938272 lph costs about 120990 Rs./ ha. For Coconut crop with spacing of 20ft x 20ft requires 1 hp pump with total head of 38.5m and should be operated for 5 hrs, flow in main line is 1.20120 lph and costs about 83417 Rs. /ha. For Mango crop with spacing of 25ft x 25ft requires 1 hp pump with total head of 38.5m and should be operated for 4.06 hrs, flow in main line is 1.532567 lph and costs about 77573 Rs./ha. For Pomegranate crop with spacing of 15ft x 15ft requires 1 hp pump with total head of 38.5m and should be operated for 2.5 hrs, flow in main line is 2.116402 lph and costs about 120990 Rs/ ha. For Orange crop with spacing of 18ft x 18ft requires 2 hp pump with total head of 38.5m and should be operated for 1.88 hrs, flow in main line is 2.962963 lph and costs about 92940 Rs. / ha.

Gowtham *et al.* (2017) developed a windows based computer application for designing efficient drip irrigation facilities using Visual Basic 6. The concern for an arrangement of this type is justified by the superior advantages offered by drip

irrigation compared to other methods. This software is user-friendly and provides interaction at all stages of the design process and a solution based on individual's requirements and the information base available within the software. The individual not only can suit to his requirement but can also compare his design with several other alternate designs with different inputs. Design of system arrived by this software was tested with manual calculations at developer's level and results were found satisfactory.

### **2.5 Android application (App) in Agriculture and Irrigation system**

Lomotey *et al.* (2013) developed a cloud-enabled mobile application known as MobiCrop that helps the farmer. Mobile technology has been gradually adopted as a measure in the agricultural space to assist farmers in decision-making. Scholars from the College of Agriculture at the University of Saskatchewan, Canada, have begun the MobiCrop project. The purpose of the project was to enable farmers to easily access up-to - date pesticide information and to make more decisions about which pesticide to apply, how and when to apply, etc. MobiCrop was developed as a mobile distributed system that followed a three-layered implementation because of its complexity. The caching technique on the mobile was suggested to assist the pesticide information offline comfort as the data was transferred to reside on the database server.

Ghogare and Monga (2015) performed the introduction and figuration of E-Agriculture applications. E-Agriculture is an emerging field focusing on the upgrading of rural and agricultural development through advanced information and communication processes. Some issues with agriculture and rural development were discussed. The main aim of their research was to reach farmers for their alertness, usage and perception in E-Agriculture. E-Agriculture is a platform for supporting marketing of products related to agriculture. The work on E- Agriculture transported the information regarding agricultural details to farmers via message and hereby projected to switch over E-Agriculture. The details such as daily alert, seasonal alert and other additional details could be sent to farmers. The daily alert could be sent to

all farmers whereas seasonal alert could be sent to farmers only for selected farmers based on clustering result. Finally the other or additional detail which was proclaimed by agriculture could be sent to all farmers. Experimental result founded better results when compared with the existing work.

Sharma *et al.* (2015) developed an E-Agro Android Program. E-Agro supported farmers with experience in crop production, pricing, fertilizers and diseases, etc., and even advice on modern cultivation methods and the use of bio-fertilizers. Their main goal was to provide the remote farmers with modern farming techniques. A new area called E-Agro, which focuses on enhancing agricultural and rural growth through a variety of technologies, has given birth to the persistently rising importance and application of information technology in agriculture. The user interface for the device is in English.

Ghanshyam *et al.* (2016) developed an Android Application for agricultural assistance to the farmers in their farming activities. This software was basically for sustainable development of farmers. This system combines modern Internet and mobile communication systems with GPS that are efficient and smooth for farming. Aim of this research to bring the spark of 21<sup>st</sup> century to that 70% population who are land worshipers. This paper is about an android application, using ICT, promoting e-governance by providing continuous information relevant to agriculture like 7/12 Extract, weather forecast, crop prices, news, Krishi Offices, Fertilizers, Pesticides, Insecticides and even enquiries about the particular information. This was one farmer's friend kind of application.

Kaur (2016) developed a software application which is basically for dissemination of information to farmers about the proper way of performing various farming activities in regional language Punjabi, of two main crops of Punjab- wheat and paddy. This application will assists farmers in performing these activities in proper manner and help them in improving their productivity of crops. Based on date

of sowing, farmers will get notifications about farming activities. Additional advice was provided along with the notification on the basis of weather conditions. This system is an android based application developed in regional language Punjabi for overcoming lingual barrier of farmers. The application is developed using Android Studio which is the official Integrated Development Environment (IDE) for developing an Android application.

Sharma *et al.* (2018) develop android based mobile application name Kisan-Sevak for agriculture sector. This application aim to diverse ranging from cropping information, online shopping market rate for farmer to weather for-casting and daily agriculture news are available in single screen. Their application helps farmers to get their food actual praise. Farmers their agriculture product shells by him and bay market actual praise and get profit.

## **CHAPTER-III**

### **MATERIALS AND METHODS**

---

This chapter deals with the concise description of formula used, to write algorithms and working process techniques of Information and Communication Technology (ICT) employed during this research work. Information of the design parameters and formula chosen for the irrigation drip system as proposed by Mane *et al.*, (2014), Michael and Ojha (2006), different books, research papers, some Indian and other country irrigation system designing companies manual. This literature helped to select suitable formula for design parameters. Various factors such as economic, topographic (location, elevation, field boundary, shape/slope, area, location of pumping unit etc.), soil (type, soil moisture holding capacity, depth and intake rate), water supply (quantity, quality, temporal variation) and crop factors (crop, row-to-row and plant-to-plant spacing) were considered while developing the Drip Irrigation App for designing the drip system of irrigation.

#### **3.1. Designing of Drip Irrigation System**

A new system must be designed on the basis of detailed review of many designs of drip irrigation system and requirement of the users. Hydraulic principles govern the flow of water through irrigation pipes. A basic requirement of these principles is necessary for understanding the design and operation of irrigation systems (Boman and Shukla, 2004). For the design of the system, crop water requirement, friction head losses in different components and operating pressure of the system are very important factors. There are several methods, formulas and tools were used for describing the design of the drip irrigation system. It is necessary to design a suitable and economically viable system to deliver a predefined amount of water at the root zone of each plant at regular intervals. This is to ensure that the plants do not suffer from stress or strain of less and over watering. Totally customized, efficient and long-life system ensures saving in water, early maturity and a bountiful harvest, season after season, year after year. Apart from all this, savings in labor and fertilizer costs.

### **3.1.1. Steps involved in the designing of drip system of irrigation**

Effective design should be reflected in its operation as far as flow variation is concerned. There should be minimum variation in emission of drippers. Overall irrigation efficiency should be more than 90 %. The design of drip irrigation differs from crop to crop, plot to plot, soil to soil and climatic conditions. In general, following steps are involved in design of drip irrigation system:

1. Designing inputs
2. Survey or field information
  - Total area of field
3. Water requirement
4. Selection of Dripper and plant
  - Total number of drippers
  - Total number of plants
5. Water application rate and irrigation timing
6. Selection and design of lateral
  - Length of lateral for one side of sub-main
  - Number of dripper on one lateral
  - Discharge rate of one lateral
  - Head loss in lateral
7. Selection and design of sub-main
  - Number of laterals on sub-main
  - Discharge rate of sub-main
  - Head loss in sub-main
8. Selection and design of main
  - Discharge of main
  - Head loss in main
9. Total head-loss
10. Selection of pump and filter
  - HP of pump

- Capacity of filter

### 3.1.1.1 Designing inputs

Compared to other irrigation systems, the Drip System is designed for high irrigation efficiency and standardized distribution of water and nutrients with a high crop value. The aim at a precise quantity and uniform application of water for each and every plant, collection of data is a prerequisite for designing of an efficient drip irrigation system. The certain input parameters needed to build an efficient system of drip irrigation are as follows:

1. Layout of the area.
2. Details of the soil type and available water source.
3. Agronomic details like crop, spacing, age, peak water requirement, crop factor etc.
4. Climatic data such as temperature, humidity, rainfall, evaporation etc.

### 3.1.1.2 Survey or field information

The survey feedback is needed for the design of the drip irrigation system to prepare an accurate layout of any field (size, shape and slope).

1. Regular size of the field distance between points of the corners. With the help of a tape in a straight line with corner points of the field, it can be measured. Total area of the field calculates by the formula.

$$\text{Total area} = A \times B \quad (3.1)$$

Where,

A = Length of plot (m)

B = Width of plot (m)

2. Elevation - Wherever possible, the slope of the ground surface can be measured with the naked eye for small plot and taken into account when designing the drip irrigation system. It is difficult to determine if the surface of the ground is sloping and undulating. In order to make a right design of the drip system, levels should mostly be taken with a leveling instrument and contour plot on the chart.

3. Soil details - The soil details includes the details of soil quality and type of soil available on field. Soil quality depends on the texture of the soil (clay, silt & sand proportion). The soil type with high sand content is categorized as light soil where as with high clay content as heavy soil. It should be enclosed with the survey report if a soil analysis report is available.
4. Water source - Water source location (well, reservoir, river pond, pump, etc.) should be properly marked on the design map and the following information noted.
  - 1) Size, volume, low rate and height from ground level or water supply above ground level or depth.
  - 2) Pump specifics for the current pump, including suction, transmission, actual, discharge and head operating time, pump, HP, planned discharge and attention.
  - 3) Water quality, impurities in water (sand, silt, algae etc).

### 3.1.1.3 Water Requirement (WR)

For the calculation of the water requirement, water needed for the plant during its lifecycle is considered. It can be defined as the quantity of water needed by the diversified crop patter for the normal growth of the crop in a given period of time under field conditions at the site. Water demand for crops depends on many variables, such as temperature, soil type, stage of plant growth, canopy influences, evapo-transpiration, etc. The accuracy of determining the requirement for crop water largely depends on the availability of records of climatic data. The need for water can be determined (Mane *et al.*, 2019).

$$WR = \frac{PE \times Pc \times Kc \times \text{crop area} \times \text{wetted area \%}}{Eu} \quad (3.2)$$

Where,

- |    |   |  |
|----|---|--|
| WR | = | Water requirement (mm/day)                     |
| PE | = | Maximum pan evaporation of the region (mm/day) |
| Pc | = | Pan coefficient                                |

|             |   |  |
|-------------|---|--|
| Kc          | = | Crop coefficient, the value of which for any crop depends upon foliage characteristics, stage of growth of crop, geographical location or environment. |
| Crop area   | = | Row to row spacing and plant to plant spacing of the crop, (m <sup>2</sup> )   |
| Wetted area | = | It is the area which is shaded due to its canopy cover when sun is over head, which depends upon stage of growth of plant.                             |
| Eu          | = | Emission uniformity of drip system (0.9)   |

#### 3.1.1.4 Selection of drippers and number of plants

Dripper is the major element of a drip irrigation system because it ensures water application uniformity over the entire irrigated area by supplying water at the uniform rate. Selection the type of drippers and number of drippers per plant is depends on type of crops, peak water requirement of the crops and the infiltration rate of the soil. Proper size of emitters must supply enough water to the root zone of the plants to meet the crop water requirement. The spacing of emitters should be according to intra row spacing of the plants. The selection of drippers and number of plants calculate (Mane *et al.*, 2019).

$$\text{Total no of dripper} = \frac{AB(m^2)}{Sl(m) \times Sd(m)} \times \text{no. of dripper per plant} \quad (3.3)$$

$$\text{Total no of plant} = \frac{AB(m^2)}{Sl(m) \times Sd(m)} \quad (3.4)$$

Where,

|                |   |                              |
|----------------|---|------------------------------|
| AB             | = | Total area (m <sup>2</sup> ) |
| S <sub>l</sub> | = | Lateral spacing (m)          |
| S <sub>d</sub> | = | Dripper spacing (m)          |

### 3.1.1.5 Water application rate and irrigation timing

Water application rate means the amount of water supplied per hectare of irrigated crop for single operation of water application. Water application rate is adjusted by using drippers which discharge at the desired application rates. Irrigation period is the length of the operation of the irrigation system that provides the plants with the appropriate amount of water. The required emitter flow can be calculated based on the water requirement of the crop, number of emitters per plant, and irrigation application efficiency of drip irrigation system. Similarly the duration of irrigation can be calculated on the basis of water requirement of the crop and dripper discharge. This can be computed (Mane *et al.*, 2019).

$$\text{Water application rate} = \text{Dripper discharge} \times \text{Number of dripper per plant} \quad (3.5)$$

$$\text{Irrigation timing} = \frac{\text{water requirement (lit/day/plant)}}{\text{dripper discharge(lps)}} \quad (3.6)$$

### 3.1.1.6 Selection and design of lateral

The lateral carries the water from the sub-main pipe and feeds to the individual drippers. Emitting devices or drippers can be fitted at determined spacing as per the requirement of the crops. Length and diameter of the lateral is decided by the discharge rate of drippers and number of drippers on single lateral. The laterals pipe are small in diameter, flexible pipes or tube is made of low-density polyethylene (LDPE) or linear low-density polyethylene (LLDPE) with a diameter of 12 mm to 32 mm. Black color laterals is to prevent the growth of algae and to avoid the effect of ultraviolet radiation. They are connected to the sub-main pipe through grommet take off and can withstand the maximum pressure of 1.5 to 2.0 kg/cm<sup>2</sup>. This can be computed (Mane *et al.*, 2019).

$$\begin{aligned} &\text{Number of dripper on one lateral} \\ &= \frac{\text{length of lateral(m)}}{\text{dripper spacing (m)}} \times \text{no. of dripper per plant} \end{aligned} \quad (3.7)$$

$$\begin{aligned} & \text{Discharge rate of one lateral (lps)} \\ & = \text{No. of dripper on one lateral} \times \text{drripper discharge(lps)} \end{aligned} \quad (3.8)$$

### 3.1.1.6.1 Head loss calculation in lateral

As the laterals are significant with regards to pressure variation, proper evaluation of frictional head losses in the laterals is essential to achieve higher uniformity. The diameter of lateral is usually selected that the difference in the discharge between two extreme emitters operating simultaneously should not exceed 10%. For having 10% variation in the flow, the pressure head difference between emitter should not exceed 10 to 15% of the average operating head, where long path emitter are used and 20% for turbulent flow emitters. Pressure variation 5 to 20% and discharge variation 5 to 10% throughout the system should be maintained in a range. The Hazen-Williams formula used for the calculation of head loss is given as follows:

$$\Delta H_{(\text{lateral})} = K \times \left( \frac{Q_{\text{lateral}}}{c} \right)^{1.852} \times (D_{\text{in}(\text{lateral})})^{-4.871} \times (L_1 + L_e) \times F \quad (3.9)$$

Where,

- K =  $1.21 \times 10^{10}$  (constant value)
- $Q_{(\text{lateral})}$  = Discharge rate of one lateral (lps)
- C = (LLDPE=LDPE=140) (Default value)
- $D_{\text{in}(\text{lateral})}$  = Inner diameter of pipe (in mm) (Diameter of lateral-2.2)
- $L_1$  = Length of lateral for one side of sub-main (m)
- F = Outlet Factor (The values of outlet factor for lateral is shown in Table( 3.1)
- $L_e$  = Outlet  $\times$  F-value (outlet = number of dripper on one lateral)

Table No. 3.1: Value of pipe outlets

| S. No | No. of Outlets | F- value |
|-------|----------------|----------|
| 1.    | 9              | 0.42     |
| 2.    | 10-11          | 0.41     |
| 3.    | 12-15          | 0.40     |
| 4.    | 16-20          | 0.39     |
| 5.    | 21-30          | 0.38     |
| 6.    | 31-36          | 0.37     |
| 7.    | 37-40          | 0.36     |
|       | More than 40   | 0.36     |

(Source: Mane *et al.*, 2019)

### 3.1.1.7 Selection and design of sub-main

Sub-main pipes usually used for drip irrigation system are within 32 mm to 75 mm diameters. This pipe can be of PVC (poly vinyl chloride) and HDPE (high density poly ethylene) material. The sub-main design is based on both capacity and uniformity. Capacity means the sub-main size should be large enough to supply the necessary amount of water to irrigate the subsequent part of the field. Uniformity means the sub-main should be designed to maintain an allowable pressure variation, so the flow into all lateral lines taking from it will have little variation. Sub-main supplies water to individual lateral. Design of sub-main is similar to that of lateral; however it differs in that the spacing between outlets is greater and larger flow rates are involved. The size and length of sub-main is determined by the number of laterals and distance between the laterals. The position of the sub-main should be at the centre of the plot usually in a plain field. On the sloping field, the lateral line should be placed along the contour and the sub-main along the slope. Sub-main discharge and head loss calculate (Mane *et al.*, 2019).

Number of laterals of sub-main =

$$\frac{\text{length of submain(m)}}{\text{lateral spacing(m)}} \times \text{number of side of submain} \quad (3.10)$$

$$\begin{aligned} \text{Discharge rate of sub-main } (Q_{\text{submain}}) = \\ \text{Discharge rate of one lateral} \times \text{No. of lateral on submain} \end{aligned} \quad (3.11)$$

$$\begin{aligned} \text{Head loss } (\Delta H_{\text{sub-main}}) \text{ (m)} = \\ K \times \left(\frac{Q_{\text{submain}}}{c}\right)^{1.852} \times (D_{\text{in (submain)}})^{-4.871} \times (L_s + L_e) \times F \end{aligned} \quad (3.12)$$

Where,

|                           |   |   |
|---------------------------|---|---|
| K                         | = | $1.21 \times 10^{10}$ (constant value)                          |
| $Q_{\text{(submain)}}$    | = | Discharge rate of sub-main ( $Q_{\text{submain}}$ ) (lps)       |
| C                         | = | (PVC=HDPE =150)(Default)  |
| $D_{\text{in (submain)}}$ | = | Inner diameter of pipe (in mm) (Diameter of Submain-4)          |
| $L_s$                     | = | Length of sub-main ( $L_s$ )(m)                                 |
| F                         | = | F-value (same as earlier)                                       |
| $L_e$                     | = | Outlet $\times$ F-value (outlet= Number of lateral on sub-main) |

### 3.1.1.8 Selection and design of main

The main line design is based on topography, operating pressure, field layout of laterals and sub-mains and the required discharge from each outlet along main line. The main line system has changing flow capacity with respect to length. The main line design is to select the proper diameter for each section to deliver water at the required rate to all sub-mains and distribution lines in the system. There are a number of pipe sizes to meet the hydraulic requirement of a given layout. The diameter of the main line pipe is one size greater than the sub-main. The size of mainline is decided by flow rate of the entire sub-main. The sizes of main line are available in 40mm, 50mm, 63mm, 75mm, 90mm and 110mm etc. The allowable variation in head is less than 20% is acceptable for the safe design. The head loss through pipe was calculated by Hazen-Williams formula:

$$\begin{aligned} \text{Head loss of main } (\Delta H_{\text{main}}) \text{ (m)} \\ = K \times \left(\frac{Q_{\text{main}}}{c}\right)^{1.852} \times (D_{\text{in (main)}})^{-4.871} \times L \text{ (m)} \end{aligned} \quad (3.13)$$

Where,

- K =  $1.21 \times 10^{10}$  (constant)  
 $Q_{(main)}$  = Discharge of main ( $Q_{main}$ ) (lps) = Discharge of sub-main (lps)  
 C = (PVC=HDPE =150) (Default)  
 $D_{in (main)}$  = Inner diameter of pipe (in mm) (Diameter of main -4)  
 $L_{(m)}$  = Length of main ( $L_m$ ) (m)

### 3.1.1.9 Total head-loss

Table No. 3.2 shows the description of different types of head losses occurred in the pipe system. As follows, the head (pressure) required at the inlet of the mainline calculation (Mane *et al.*, 2019).

$$\text{Total head required} = \text{suction head} + \text{delivery head} + \text{filter loss} + \text{mainline loss} + \text{operating loss} + \text{fitting loss} + \text{venture head loss} + \text{elevation difference} \quad (3.14)$$

Table No.3.2: Description of head losses

| S. No. | Head               | Description   |
|--------|--------------------|---|
| 1.     | Delivery head      | Vertical distance between centre of pump to ground level  |
| 2.     | Friction losses    | Frictional head losses in different type of filter, it is assumed 2m for each type.                                   |
| 3.     | Main line losses   | Frictional head losses occur in mainline  |
| 4.     | Operating pressure | In general 10 m pressure is required for non pressure compensating dripper and 15 m for pressure compensating dripper |
| 5.     | Fitting losses     | Loss in fitting like bends , elbows, tees, reducers, and valves etc.  |
| 6.     | Ventury head       | Pressure required to operate the ventury (or) fertilizer applicator and is assumed to be 5m                           |
| 7.     | Elevation          | Vertical distance between ground level near to water source the highest level of ground                               |

(Source: Reddy *et al.*, 2017)

### 3.1.1.10 Selection of pump

#### 3.1.1.10.1 HP of pump

Size of pump depends upon the flow of water required and total pressure required at the pump to operate the irrigation system efficiently. While designing the system from drippers to mainline, finalize the system flow. The pump should have the capacity to deliver this flow. After finalizing the discharge and total head of pump, then calculate the approximate HP of pump using (Mane *et al.*, 2019).

$$HP = \frac{Q \times H}{75 \times a \times b} \quad (3.15)$$

Where,

- HP = Horse power
- Q = Main line discharge
- a = Efficiency of motor (assumed %)
- b = Efficiency of motor (assumed %)

#### 3.1.1.10.2 Capacity of filter

The filtration requirement depends on the size of the emitter's flow direction, water quality, and mainline flow. Filter capacity was calculated by using following (Mane *et al.*, 2019).

$$\text{Filter capacity} = Q_{(\text{main})} \times 3.6 \quad (3.16)$$

Where,

- $Q_{(\text{main})}$  = Main line discharge

Table No. 3.3: Filter capacity

| S. No. | Flow rate (m <sup>3</sup> /hr) | Size of filter (inch) | Size of filter (mm) |
|--------|--------------------------------|-----------------------|---------------------|
| 1.     | 3.5-5.0                        | 1                     | 25                  |
| 2.     | 5.0-9.0                        | 1.5                   | 40                  |
| 3.     | 10-15                          | 2.0                   | 63                  |
| 4.     | 15-25                          | 2.0                   | 63                  |
| 5.     | 25-40                          | 3                     | 90                  |

(Source: Mane *et al.*, 2019)

### **3.2 App Development Process**

The App development process is a system consisting of a comprehensive process plan outlining how particular Apps can be planned, created, run, managed, replaced or enhanced. The phase cycle defines a procedure for establishing the overall process of software development and improving software quality. From its beginning to its working estimate, App development process contains all the steps of an App. It is a technique for planning and managing the development, testing, and development of an App. The App development process is a series of phases that provides a blueprint for an application's development and process cycle management. This process describes all phases of the life cycle of development that are used, but research, system design and development to prepare and execute the sequence of events needed at scheduled time to achieve App or system efficiency. The following procedure involves in the App development process:

3.2.1 System analysis

3.2.2 Feasibility analysis

3.2.3 Requirement analysis

3.2.4 Coding

3.2.5 Testing

3.2.6 Implementation

#### **3.2.1 System analysis**

This is the initial stage of App development process and very significant step. This stage is vital to the App's success. The need to create new Apps is taken into account at this point and issues are also found in the current framework. The initial system analysis involves planning a system proposal that lists the problem, studying the goal and reference for research, predicting, restricting system benefits, etc. The device plan is prepared for approval by the location and the device prior to the customer. The consumer approves the request for the next stage of the loop. Some might even refuse the proposal or request some alteration of the proposal from the user. Via these steps, the device research phase:

- 1) Existing system- The design of drip irrigation system is a tedious job for any farmer or extension worker. There are numerous software's available to deal with the design

related issues but the drawback is the hardware requirement for the same. Presently with the development of Information and Communications Technology (ICT) and awareness among the farmers, android platform based mobile is being readily used and adopted by the farmers.

- 2) Proposed system- The android based App for the designing of drip irrigation system that can be easily installed by the farmers or extension workers and can be used extensively to overcome the issues of faulty design of drip irrigation system.

### **3.2.2 Feasibility/ Probability analysis**

Analysis of feasibility is used to evaluate the issue and solution of a planned project to increase the efficiency of a project and the consistency of the desired outcome. The design and components of feasibility studies are primarily dependent on the region in which the project under review is implemented. Feasibility is characterized as the realistic level at which a project can be successfully achieved. A feasibility analysis is conducted to estimate feasibility, which determines if the solution considered is realistic and feasible in the App. Accessibility of information services, cost approximation for the production of the software, benefits of the software to the association after it has been created and costs to be gained during the feasibility study are calculated during its maintenance. The goal of the feasibility study is to determine the motive for the creation of an App that is adequate for the consumer, versatile for modification and consistent with existing standards. Technology feasibility, operational feasibility and economic feasibility are various types of feasibility that are manually considered.

#### **3.2.2.1 Technology feasibility**

The technological element points out if the viability of the project is beyond the limits of current technology and if the technology exists at all or if it can be achieved under such resource constraints such as budget, schedule plan. The scheduling software is based on Android, which is very stable and technologically developed. In terms of technological viability, device experts take a look at the organization's requirements, such as:

- Input system which can enter in the effective time a quantity of data.
- Output system that can generate output in a bulk in an efficient time.
- Ease to understand.

### **3.2.2.2 Operational feasibility**

Operational feasibility is a measure of how safe the problems are solved by a proposed method, which takes effective advantage of the opportunities recognized during the definition of scope which fulfils the requirements recognized in the creation of the system requirement review process. Operational viability depends on the project's user criteria and requires predicting whether the system will be used if it is developed and implemented.

### **3.2.2.3 Economic feasibility**

Economic analysis may also be viewed as an analysis of costs or benefits. It is the most commonly used tool for assessing a new system's usefulness. The technique is to control the gains and savings that are anticipated from an aspirant scheme in economic analysis and compare them to costs. If sales compensate for costs, the decision to design and implement the system is taken. The App enhances the user's economic status by enhancing crop productivity.

### **3.2.3 Requirement Analysis**

Analysis of requirements involves identifying the needs and goals of users in the sense of expected consumer use, drip irrigation design system framework and accepted system features to evaluate system function requirements. Previous studies are studied and restructured; mission development and description of the method are maintained. The purpose of Requirements Analysis is to:

- Enhance designing objectives and requirements.
- Define primary performance objectives and refine them into requirements;
- Identify and outline constraints that bound solutions.

- Define functional and performance requirements based on customer provided measures of effectiveness.

### **3.2.4 Coding**

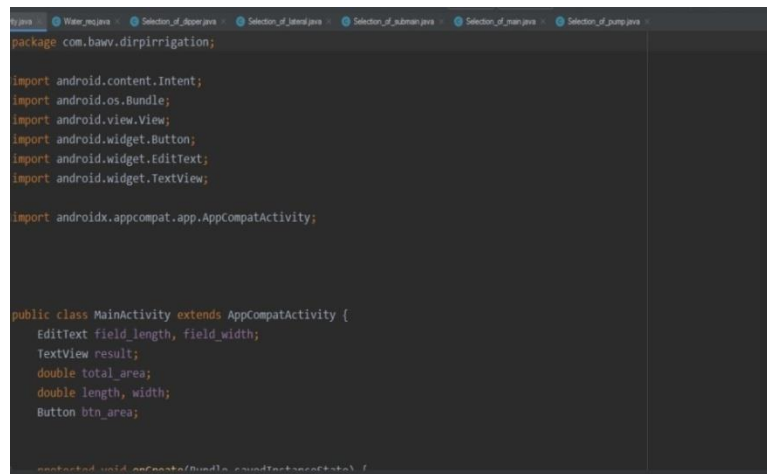
The system design needs to be worked in order to make it a workable system. This leads to the coding of design into computer language, i.e., programming language. This is always related to as the programming stage, in which the programmer converts the requirements of the software into machine instructions, known as programme. It is an important step where with the assistance of a computer language, the specified processes are changed into control provisions. The programme connects the movements of data and manages in a device the complete operation. The testing and maintenance effort is minimized by a well transcribed code. Generally, it is assumed that the services must be segmental in design. This helps, if required, in fast development, maintenance and further variations. The scheduling programme is a framework based on Android. For coding Android Studio Integrated development environment is used in whose backend core Java language.

#### **3.2.4.1 Strings.xml file**

Some information is string. Xml file of android studio. Format of string .xml file is:

```
<string= “field of string”>test </string >
```

Where attribute “field information” specifies the unique name of string by which string identified and “text string” is date store under that name.



```

package com.bawv.dirirrigation;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText field_length, field_width;
    TextView result;
    double total_area;
    double length, width;
    Button btn_area;
}

```

Fig. 3.1: String.xml file

### 3.2.5 Testing

It is the device evaluation procedure and its module with the aim of evaluating whether or not it meets the requirement defined. In order to detect any mistake, holes, or lacking necessities contrary to the actual necessity, testing is performing a method in the method. A test run of the system is performed until the new system is fully put into operation, removing all the bugs. An effective scheme is a critical step. A test plan should be created and run on a given collection of test data after checking the entire programme of the scheme. The test run performance should meet the anticipated outcomes. Effective test results for the new system are carried out:

#### 3.2.5.1 Unit testing

The function of the unit is to segregate each programming component and to show that the individual component is correct in terms of specifications and functionality.

#### 3.2.5.2 Integration testing

After unit integration evaluation, the evaluation of the collective component of an application is conducted to decide if the property works to ensure that there is no fault in the flow of correct information from one module to another.

### **3.2.5.3 Program testing**

They must be openly checked with the fitted test data when the programmes have been encoded and compiled and brought to working conditions. Any variation and validation must be tested and any uncomfortable pattern must be noted and debugged.

### **3.2.5.4 System testing**

Finally, the whole system is mutually reviewed to ensure the specifications of the customer. One real date is performed at this point of the exam. The findings are reviewed to make sure that the app is error free. It is corrected according to a further test for planned performance if any error is observed. The final app is checked for the correct feature as a whole and has the desired outcome and meets the specifications of the customer.

## **3.2.6 Implementation**

The implementation process starts after creating the App. Implementation is the stage of a project in which theory is converted into functional practice. Installing the App will be easy to run on any Android based smart phone consumer. In this process, the major steps involved are: App installation, Conversion, Design user field and Documentation.

### **3.2.6.1 Flow diagram of Drip Irrigation App**

Flow chart 3.1 shows a two-dimensional flow diagram, which explains how the data is stored and transmitted into the system. The graphical representation recognizes each source of data and how it interacts with other data sources to reach a final output. It is a documentation support which is understood by both programmers and non programmers. Progressive farmers, field staff, technical staff and research scholars can easily understand the design of drip system of irrigation through this flow diagram and can also easily operate and design their system by themselves.

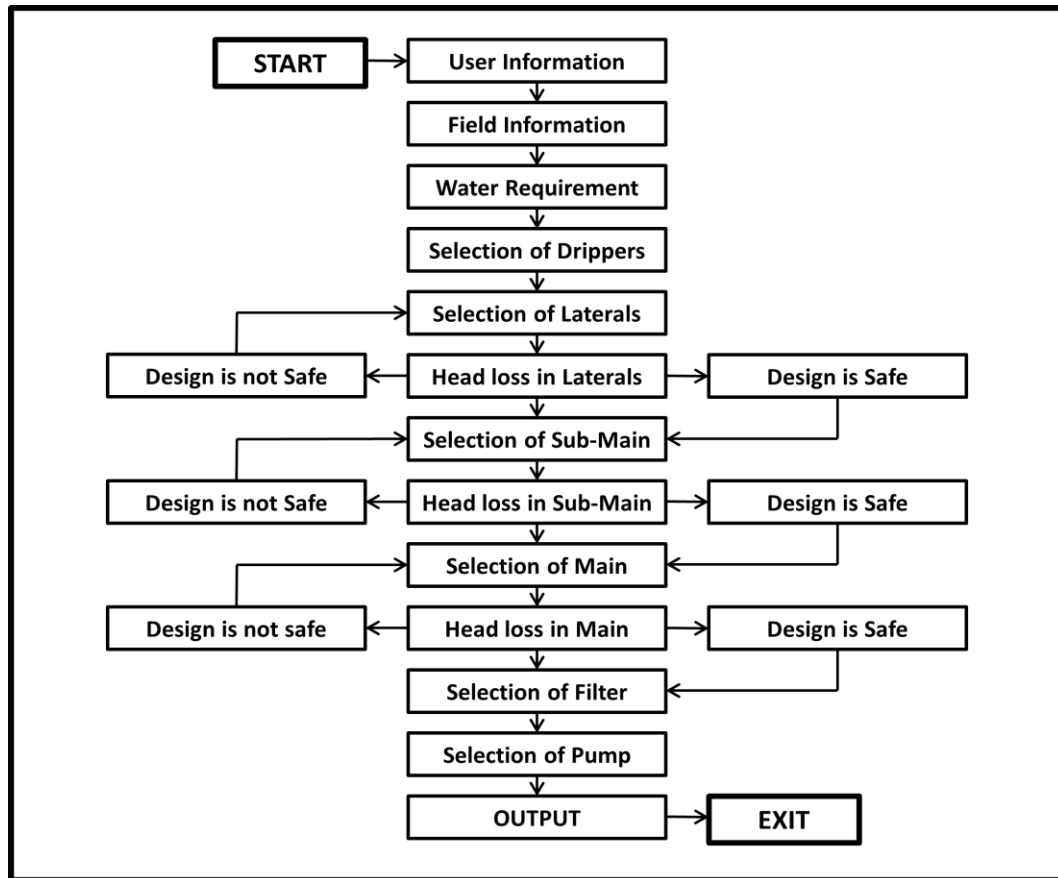


Fig. 3.2: Flow diagram of Drip Irrigation App

### 3.3 Proposed software tool

The specifications of the proposed system are defined in the requirement analysis process and then translated into the system design stage, the programming language and the hardware and software framework in which the new system will run are also determined in the design phase.

#### 3.3.1 Android

Android is a mobile operating system (OS) based on the Linux kernel and developed by Google. Android is developed mainly for handheld touch screen devices such as smart phones and tablets, with specialized user interfaces for televisions (Android TV), vehicles (Android Auto), and wrist watches with a user interface based on direct manipulation (Android Wear). The default user interface of Android is based on direct manipulation, using touch inputs that closely fit real-world behavior, such as swiping, clicking, pinching,

and reverse pinching, and a virtual keyboard to manipulate on-screen objects. The user input response is designed to be immediate and offers a fluid touch interface, often using the device's vibration features to provide the user with haptic feedback. Some applications use internal hardware such as accelerometers, gyroscopes and proximity sensors to respond to extra user behavior. Android comes with an online app store for the Android market. It allows Android users to select, and download applications developed by third party developers and use them.

Android is open source and it can be used for creating apps for anybody. Android provides a rich platform for applications that enables one to create innovative mobile device apps and games in a Java language environment. Android tools are developed as a mixture of different components that can be independently invoked. An user activity, for example, offers a single screen for a user interface and a service performs work in the background separately. One may start another component from one component with a purpose. In another app, one can also start a part, such as an operation to view an address in a map app. This model offers many entry points for a single app and allows any app to behave as the "default" of a user for an action that may be invoked by other apps.

### **3.3.2 Application**

There are Android applications written in the Java programming language. The Android Software Development Kit (SDK) tools compile code into an Android package (APK), which is an archive file with a .apk suffix, along with any data and resource files. All the contents of an Android app are contained in one APK file and it is the file used by Android users to install the app. Every Android app lives in its own security sandbox once installed on a device:

- The Android operating system is a Linux multi-user system where each software's is a different user.
- The framework assigns each app a special Linux user identity by default (ID). The framework sets permissions for all files in an app, so that they can only be accessed by the user ID assigned to that app.
- Each process has its own virtual machine, so the code of an app runs from other

software in isolation.

- Each app runs its own Linux process by default. When some of the components of the app need to be run, Android starts the process, then shuts down the process when it is no longer necessary or when the device needs to recover memory for other applications.

The theory of least privilege is applied by the Android framework in this process. That is, each app only has access to the components it needs to do its job, by default, and no more. This provides a very safe environment where an app is unable to access parts of the system for which permission is not required.

### **3.4 Software used for the development of App**

#### **3.4.1 Android Studio**

The proposed scheduling app has been developed using Android Studio Software. For Android app development, Android Studio Software is the official Integrated Development Environment (IDE). Android Studio offers even more features that improve one's productivity when creating Android apps, in addition to IntelliJ's powerful code editor and developer tools, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where one can app develop for all Android devices
- Instant Run to push changes to running app without building a new APK
- Code templates and GitHub integration to help build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and Native Development Kit (NDK) support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

**Project Structure:** Each Android Studio project includes one or more source code files and resource file modules. Modulus forms include:

- Modules of Android applications
- Modules for library
- Modules of Google App Engine

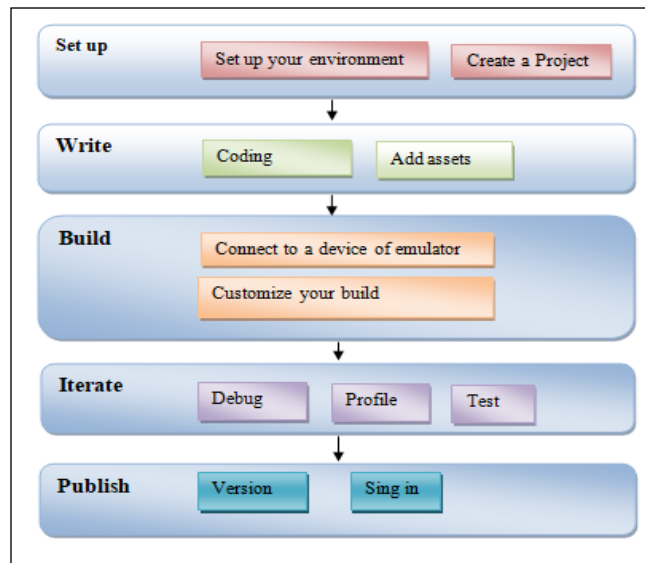
Android Studio shows the project files in the Android project view by default. This view is arranged by modules to provide easy access to the main source files of your project. Under Gradle Scripts, all build files are available at the top level and each app module includes the following folders:

- Manifests: Includes a file called AndroidManifest.xml.
- Java: Includes source code files for Java, including test code for JUNIT.
- Res: Includes all non-code tools, such as XML layouts, strings for the UI, and images for bitmaps.

In order to concentrate on particular aspects of app creation, one may also configure the view of the project files.

### 3.4.2 Workflow of Android App

Some specialized tools are needed in the workflow for developing an Android app. An overview of that cart is given in the following list:



Flow chart 3.3: Work Flow of App Development

1. **Setup:** Download the Android Studio Software and create a project.
2. **Write app:** In order to help function faster, write better code, build a UI and develop resources for various types of devices, Android Studio provides a number of tools and knowledge. In this step, the logic of the application is implemented.
3. **Develop and Run:** The project is integrated into a debug gable APK package during this stage that can be installed and run on the emulator or an Android-powered computer.
4. **Debug, profile and test:** This is the iterative stage in which app writing begins, but with an emphasis on removing bugs and improving the output of the app. In those endeavors, of course, producing tests will help. Use the Android Monitor to display and evaluate different performance metrics such as memory use, network traffic, central processing unit effect, and more. The designed app is then checked and debugged until all goals are met.
5. **Publish:** Then, by registering, the installed app is available and then registered users can login and use the app.

### 3.4.3 Java Script Object Notation (JSON)

JSON is an open standard data exchange format based on a JavaScript syntax subset. JSON is text-based, lightweight and easily readable/writable, usually considered. JSON is language-independent, although closely related to JavaScript. JSON uses conventions close to other languages (e.g., C, C++, Java, Perl and Python), but independent, making JSON an ideal language for data exchange. The following include the JSON characteristics:

- Flexibility, allowing keys to be developed by the programmer.
- Less overhead, as content is mostly information.
- Portable information.
- Non-ownership.
- Popular and convenient Web services format.

JSON is used as a data exchange format by some over XML because it is less verbose, functions easily, reduces the size of data and simplifies the processing of documents. In

web development, it is used extensively, especially because it transfers information seamlessly across potentially incompatible technologies. A Java application running on a UNIX box or a C# application running on Windows, for instance, may be affected.

JSON is a way of storing information in a standardized format in which each piece of data typically has an identifier (known as a key) and a value (which can be in multiple formats). It can also have a nested structure where multiple children (keys) etc. can have a key. It is also referred to as a variant of XML for JavaScript. The main reason for using JSON is based on the principle of key value pairs (KVPs), which are suitable because they have very detailed control over the particular information you want to search for or extract and manipulate. In order to share web resources, JSON uses a subset of JavaScript.

## CHAPTER-IV

### RESULTS AND DISCUSSION

---

This chapter deals with the results related to the development of App for design of drip system of irrigation. This Drip Irrigation App was developed with the help of Android Studio software and for coding Java language was used. The Drip Irrigation App can be used to build drip system design for vegetables and fruits crop grown at the different locations. This App needs only limited data and can measured a large number of data outputs for small and large area for the design of drip system. The result of this Drip Irrigation App include following steps for design of drip system are explain in this chapter.

#### **4 App include following step for the design of drip irrigation system**

##### **4.1 User information Activity**

This is the first activity that opens when user installs the app. User needs to provide a name, institute, village, correct contact number and farm khasra number, which user enters in the App while registering. The information provided by user's matches with the data saved in the output database table or PDF form. This is static App; therefore, it is not requiring internet connection. Filling all the desired information by the user, touch on next button to get next form. Fig. 4.1 shows the blank form of user related information and information filled by the users is shown in Fig. 4.2.

##### **4.2 Field information Activity**

To plan and design a drip irrigation system, a cadastral area map with system layout is needed. After registration of user in the App, select the name of crop and enter the length of plot, width of plot and slope of the field. After putting all the desired values, touch on calculate button to get the total area in square meter and then touch on next button to get next form. Blank form of field related information is shown in Fig. 4.3 and the information related with the particular field is filled by the users is shown in Fig. 4.4.

**Drip Irrigation**

**User Information**  
Name of Farmer/Scholar

\_\_\_\_\_

Institute  
\_\_\_\_\_

Village/City  
\_\_\_\_\_

Contact No.  
\_\_\_\_\_

Khasra No.  
\_\_\_\_\_

NEXT

Fig. 4.1 User information

**Drip Irrigation**

**User Information**  
Name of Farmer/Scholar

Priyanka netam

Institute  
IGKV

Village/City  
Raipur

Contact No.  
9039497899

Khasra No.  
258

NEXT

Fig. 4.2 User information with input

**Drip Irrigation**

**Field information**  
Name of Crop

\_\_\_\_\_

Length of plot (m)  
\_\_\_\_\_

Width of plot (m)  
\_\_\_\_\_

Slope (%)  
\_\_\_\_\_

CALCULATE

Total Area  
Result m<sup>2</sup>

NEXT

Fig. 4.3: Field information

**Drip Irrigation**

**Field information**  
Name of Crop

Banana

Length of plot (m)  
100

Width of plot (m)  
100

Slope (%)  
.02

CALCULATE

Total Area  
10000.0 m<sup>2</sup>

NEXT

Fig. 4.4: Field information with input

### **4.3 Water requirement (WR)**

When the design of drip irrigation system is planned, the peak water requirement is considered for the crop during its lifecycle. Pan evaporation is taken into account when measuring the peak water requirement. Data of pan evaporation is obtained from the nearest weather station. Crop water requirement can vary depending on the type of soil, the agro-climate situation, the variety of crops, etc. Then enter the value of crop coefficient ( $K_c$ ) according to the stage of the crop. Then enter the value of percentage wetted area, row to row and plant to plant spacing according with the selected crop, pan coefficient ( $P_c$ ) and emission uniformity ( $E_u$ ). After putting all the desired values, click on calculate button to get the crop water requirement in liter per day per plant and then click on next button to go to the next page. Blank form of water requirement in Drip Irrigation App is shown in Fig. 4.5 and the values related with the water requirement is filled by the users is shown in Fig. 4.6.

### **4.4 Selection of drippers**

The selection of drippers as well as the total design of drip system is depends on peak water requirement. There are lots of inline and online drippers or emitters are available in the market. It is very important to select the proper size of drippers. Based on peak water requirement, soil type, water availability, electricity availability etc., it emits enough water to fulfill the water requirement within predefined time. Blank form of dripper related information is shown in Fig. 4.7 and the values related with the selection of dripper is filled by the users is shown in Fig. 4.8. First of all, select the drippers from the available list of dripper sizes viz., 2 lph, 4 lph, 8 lph, 10 lph respectively. Then enter the number of drippers per plant which are required in App and touch on calculate button to get total number of plants, total number of drippers, water application rate in lph and irrigation timing in hrs. And then touch on next button to get next form.

**Drip Irrigation**

**Water Requirement**  
Pan evaporation (PE)(mm/day)

\_\_\_\_\_

Crop coefficient (Kc)

\_\_\_\_\_

Wetted area (%)

\_\_\_\_\_

Lateral spacing /Row to Row spacing (Sl) (m)

\_\_\_\_\_

Dripper spacing / Plant to plant spacing (Sd) (m)

\_\_\_\_\_

Pan coefficient (Pc)

\_\_\_\_\_

Emission uniformity of drip (Eu)

\_\_\_\_\_

**CALCULATE**

Water requirement  
Result (liter/day/plant)

**NEXT**

Fig. 4.5 Water requirement

**Drip Irrigation**

**Water Requirement**  
Pan evaporation (PE)(mm/day)

10

Crop coefficient (Kc)

.7

Wetted area (%)

.25

Lateral spacing /Row to Row spacing (Sl) (m)

6

Dripper spacing / Plant to plant spacing (Sd) (m)

6

**Pan coefficient (Pc)**

.7

Emission uniformity of drip (Eu)

.90

**CALCULATE**

Water requirement  
49.0 (liter/day/plant)

**NEXT**

Fig. 4.6: Water requirement with input

**Drip Irrigation**

**Selection of dripper**  
Dripper discharge (lph)

\_\_\_\_\_

Select from 2,4,8,10

Number of dripper per plant

\_\_\_\_\_

Select from 1 to 6

**CALCULATE**

Total number of Plant  
Result

Total number of Dipper  
Result

Water application rate  
Result lph

Irrigation timing  
Result hr

**NEXT**

Fig. 4.7: Selection of dripper

**Drip Irrigation**

**Selection of dripper**  
Dripper discharge (lph)

4

Select from 2,4,8,10

**Number of dripper per plant**

4

Select from 1 to 6

**CALCULATE**

Total number of Plant  
277.77777

Total number of Dipper  
1111.1111

Water application rate  
16.0 lph

Irrigation timing  
4.0 hr

**NEXT**

Fig. 4.8: Selection of dripper with input

#### 4.5 Selection and design of lateral

Lateral carries water from sub-main and feeds to the individual drippers. The size and length of lateral is decided by the discharge of the drippers and number of drippers on one lateral. Design the layout of the system considering maximum lateral running length so as to obtain higher uniformity. Laterals are available in various sizes in the market. The Hazen-Williams formula is used for determining the friction head loss in laterals. The permissible head loss due to friction by taking into account 10% head variation or will be 1.0 m. After finalizing the information regarding drippers, select the pipe material that is used for the design of the system and lateral diameter from the available list viz., 12mm, 16mm and 20mm. And then enter the length of lateral for one or both side of sub-main (select one side for 1 both side for 2). Then click on calculate button after putting all the values in the App. Immediately the App will calculate the length of lateral for one side of sub-main, number of drippers on one lateral, discharge rate of one lateral and head loss in lateral.

After getting the result, if the head loss in lateral is less than 1.0 meter then message box will appear and show “Lateral size is Acceptable”, the calculated head loss is sufficient to carry the flow or head loss is within the permissible limit and go to the next and if the head loss is more than 1.0 meter than message box will display “Lateral size is not Acceptable”, the calculated head loss is not sufficient to carry the discharge and select another diameter of lateral. The App shows the selected size of pipe is the internal diameter of lateral. After completing the selection and design of laterals click on next button to get next form and to go to previous form click the back button of your mobile. Blank form of lateral related information is shown in Fig. 4.9 and the values related with the selection of lateral is filled by the users is shown in Fig. 4.10. Similarly Fig. 4.11 shows the size of lateral is not acceptable, change the diameter and repeat the procedure.

Fig. 4.9 Selection of lateral

Fig. 4.10 Selection of lateral with input

Fig. 4.11 Selection of lateral with input (Not acceptable case)

#### 4.6 Selection and design of sub-main

After finalizing the size and number of emitters and design of laterals, decide the number of sections, for the entire area, so that irrigation cycle can be completed in the available time for operation. Design of sub main for the particular section is based on both capacity and uniformity. Sub-main pipes are available in various sizes in the market. The Hazen-Williams formula is used for determining the friction head loss in sub-main. The permissible head loss due to friction by taking into account is 20% head variation or will be 2.0 m. After getting the information regarding drippers and laterals, select the pipe material that is used for the design of sub-main and enter the whether the lateral is connected either one side of sub-main or on two sided of sub-main. After that, select the diameter of sub-main from the available list viz., 50mm, 63mm, 75mm, 90mm and 110mm. And then click on calculate button after putting all the values in the App. Immediately the App will calculate the number of laterals on sub-main, discharge rate of sub-main and head loss in sub-main pipe.

After getting the result, if the head loss in sub-main is less than 2.0 meter then message box will appear and show “Sub-main size is Acceptable”, the calculated head loss is sufficient to carry the flow or head loss is within the permissible limit and go to the next and if the head loss is more than 2.0 meter than message box will display “Sub-main size is not Acceptable”, the calculated head loss is not sufficient to carry the discharge and select another diameter of sub-main. The App shows the selected size of pipe is the internal diameter of sub-main. After completing the selection and design of sub-main, click on next button to get next form and to go to previous form touch the back button of your mobile. Blank form of sub-main related information is shown in Fig.4.12 and the values related with the selection of sub-main is filled by the users is shown in Fig.4.13. Similarly Fig.4.14 shows the size of sub-main is not acceptable, change the diameter and repeat the procedure.

**Drip Irrigation**

**Selection of Sub-main**

Pipe material  
Diameter of sub-main(mm)

Select from 50,63,75,90,110

Number of lateral on sub-main

Select one side for 1 or both side for 2

Length of sub-main(Ls)(m)

**CALCULATE**

Number of lateral on sub-main  
Result

Discharge rate of sub-main (Q submain)  
Result lps

Head loss of sub-main ( $\Delta H$  submain)  
Result m

**NEXT**

Fig. 4.12: Selection of sub-main

**Drip Irrigation**

**Selection of Sub-main**

PVC  
Diameter of sub-main(mm)

50

Select from 50,63,75,90,110

Number of lateral on sub-main

2

Select one side for 1 or both side for 2

Length of sub-main(Ls)(m)

100

**CALCULATE**

Number of lateral on sub-main  
33.333336

Discharge rate of sub-main (Q submain)  
1.234568 lps

Head loss of sub-main ( $\Delta H$  submain)  
0.535029 m

Head loss < 1m Acceptable

**NEXT**

Fig.4.13: Selection of sub-main with input

**Drip Irrigation**

**Selection of Sub-main**

PVC  
Diameter of sub-main(mm)

40

Select from 50,63,75,90,110

Number of lateral on sub-main

2

Select one side for 1 or both side for 2

Length of sub-main(Ls)(m)

100

**CALCULATE**

Number of lateral on sub-main  
33.333336

Discharge rate of sub-main (Q submain)  
1.234568 lps

Head loss of sub-main ( $\Delta H$  submain)  
1.7657175 m

Head loss > 1m Not Acceptable (in case select next size of diameter)

**NEXT**

Fig. 4.14 Selection of sub-main with input (Not acceptable case)

#### 4.7 Selection and design of main

After finalizing drippers, design of laterals and sub-main lines, connect all the sub-mains to the water source using the main line. Size of main line is decided by considering the quantity of water flowing through the entire sub-mains. It's very important to decide how many sub-mains are to be operated at one time and based on that flow through the main line is calculated and size of mainline is designed for that flow. There are a number of pipe sizes to meet the hydraulic requirement of a particular layout. Generally the size of main line is one size higher than sub-main pipe.

The Hazen-Williams formula is used for determining the friction head loss in main. The permissible head loss due to friction by taking into account is 20% head variation or will be 2.0 m. After getting the information regarding drippers, laterals and sub-main pipe line, select the pipe material that is used for the design of main line and the diameter of main line from the available list viz., 50mm, 63mm, 65mm, 75mm, 90mm, 100mm, 110mm, 125mm and 140mm. And then touch on calculate button after putting all the values in the App. Immediately the App will calculate the head loss in main pipe.

After getting the result, if the head loss in main line is less than 2.0 meter then message box will appear and show "Main pipe size is Acceptable", the calculated head loss is sufficient to carry the flow or head loss is within the permissible limit and go to the next and if the head loss is more than 2.0 meter than message box will display "Main pipe size is not Acceptable", the calculated head loss is not sufficient to carry the discharge and select another diameter of main pipe. The App shows the selected size of pipe is the internal diameter of main line. After completing the selection and design of main line, touch on next button to get next form and to go to previous form touch the back button of your mobile. Blank form of main pipe related information is shown in Fig. 4.15 and the values related with the selection of main pipe is filled by the users is shown in Fig. 4.16. Similarly Fig.4.17 shows the size of sub-main is not acceptable, change the diameter and repeat the procedure.

**Drip Irrigation**

**Selection of Main**

Pipe material  
Diameter of Main line(mm)

Select from 50,63,65,75,90,100,110,125,140

Length of main (Lmain)(m)

CALCULATE

Head loss of main ( $\Delta H$  main)  
Result m

NEXT

Fig. 4.15: Selection of main

**Drip Irrigation**

**Selection of Main**

PVC  
Diameter of Main line(mm)

50

Select from 50,63,65,75,90,100,110,125,140  
Length of main (Lmain)(m)

50

CALCULATE

Head loss of main ( $\Delta H$  main)  
0.6634784 m

NEXT

Head loss < 2m Acceptable

Fig.4.16:Selection of main with input

**Drip Irrigation**

**Selection of Main**

PVC  
Diameter of Main line(mm)

50

Select from 50,63,65,75,90,100,110,125,140  
Length of main (Lmain)(m)

1000

CALCULATE

Head loss of main ( $\Delta H$  main)  
13.2695675 m

NEXT

Head loss > 2m Not Acceptable (in case select next size of diameter)

Fig. 4.17 Selection of main with input (Not acceptable case)

#### **4.8 Total head loss**

Size of pump depends upon the flow of water required and total pressure required at the pump to operate the irrigation system efficiently. While designing the system from drippers to mainline, first to finalized the system flow. The pump should have the capacity to deliver this flow. The required total head of pump is the sum of all heads viz., suction head, delivery head, head loss in venture, head loss in filters, head loss in main pipe line, head loss in fittings and operating pressure head. Enter the values of all losses in the system, touch on calculate button on the Drip Irrigation App to get the total head loss and then touch on next button to get next form and to go to previous form touch the back button of your mobile. Fig. 4.18 shows the blank form of head loss related information and the values related with head loss in the system is filled by the users is shown in Fig. 4.19.

#### **4.9 Selection of pump and filter**

The HP of the appropriate pump set is mainly depending on the discharge of the system and the total operating head. The duty of pump in terms of discharge and head is determined after the diameter, length and discharge of all the emitters, laterals, sub-mains, and mains are decided and the friction losses are estimated. After getting the total head loss, putting the values of efficiency of motor and efficiency of pump in the Drip Irrigation App and then touch the calculate button to get horse power of pump and filter capacity to be required for the system. Then touch the “create PDF” button to get all details and dimensions of designed system of drip irrigation. Fig. 4.20 shows the blank form of selection of pump and the values related with pump and filter selection for the system is filled by the users is shown in Fig.4.21.

**Drip Irrigation**

**Total Headloss**  
Suction+Delivery (head)(m)

Operating pressure(m)

Fitting loss(m)

Filter loss(m)

Venture loss(m)

Elevation difference(m)

CALCULATE

Total head  
Result m

NEXT

Fig. 4.18: Total head loss

**Drip Irrigation**

**Total Headloss**  
Suction+Delivery (head)(m)

12

Operating pressure(m)

10

Fitting loss(m)

2

Filter loss(m)

2

Venture loss(m)

5

Elevation difference(m)

0

CALCULATE

Total head  
32.198505 m

NEXT

Fig. 4.19 Total head loss with input

**Drip Irrigation**

**Selection of Pump**  
Efficiency of motor(%)

Efficiency of pump(%)

CALCULATE

Pump  
Result HP  
Selection Of Filter  
Result m<sup>3</sup>/hr

CREATE PDF

Fig. 4.20: Selection of pump

**Drip Irrigation**

**Selection of Pump**  
Efficiency of motor(%)

.80

Efficiency of pump(%)

.75

CALCULATE

Pump  
0.883361 HP  
Selection Of Filter  
4.4444447 m<sup>3</sup>/hr

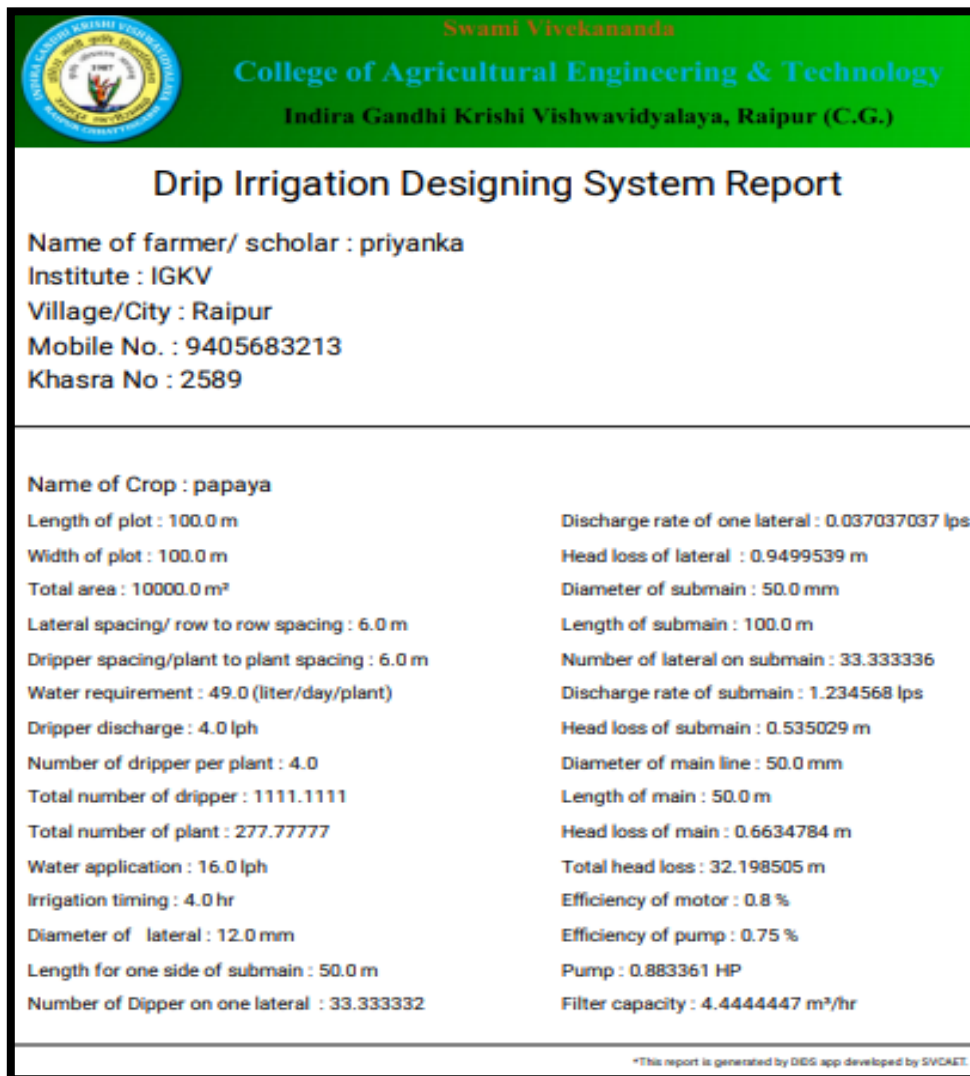
CREATE PDF

Use 1 inch (25mm) filter

Fig. 4.21: Selection of pump with input

#### 4.10 Output –

Fig. 4.22 shows the output of all calculated data in single PDF page. Based on all the objectives and executing of the methodology the Drip Irrigation App is developed or create a PDF file i.e. Drip Irrigation Designing System Report.



**Swami Vivekananda**  
**College of Agricultural Engineering & Technology**  
**Indira Gandhi Krishi Vishwavidyalaya, Raipur (C.G.)**

**Drip Irrigation Designing System Report**

Name of farmer/ scholar : priyanka  
 Institute : IGKV  
 Village/City : Raipur  
 Mobile No. : 9405683213  
 Khasra No : 2589

---

|  |   |
|--|---|
| Name of Crop : papaya                          |   |
| Length of plot : 100.0 m                       | Discharge rate of one lateral : 0.037037037 lps |
| Width of plot : 100.0 m                        | Head loss of lateral : 0.9499539 m              |
| Total area : 10000.0 m <sup>2</sup>            | Diameter of submain : 50.0 mm                   |
| Lateral spacing/ row to row spacing : 6.0 m    | Length of submain : 100.0 m                     |
| Dripper spacing/plant to plant spacing : 6.0 m | Number of lateral on submain : 33.333336        |
| Water requirement : 49.0 (liter/day/plant)     | Discharge rate of submain : 1.234568 lps        |
| Dripper discharge : 4.0 lph                    | Head loss of submain : 0.535029 m               |
| Number of dripper per plant : 4.0              | Diameter of main line : 50.0 mm                 |
| Total number of dripper : 1111.1111            | Length of main : 50.0 m                         |
| Total number of plant : 277.77777              | Head loss of main : 0.6634784 m                 |
| Water application : 16.0 lph                   | Total head loss : 32.198505 m                   |
| Irrigation timing : 4.0 hr                     | Efficiency of motor : 0.8 %                     |
| Diameter of lateral : 12.0 mm                  | Efficiency of pump : 0.75 %                     |
| Length for one side of submain : 50.0 m        | Pump : 0.883361 HP                              |
| Number of Dipper on one lateral : 33.333332    | Filter capacity : 4.4444447 m <sup>3</sup> /hr  |

\*This report is generated by DIES app developed by SVCAET.

Fig. 4.22: Output in PDF file

#### 4.11 Validation of developed App-

Design of the system obtained by Drip Irrigation App was tested with different researcher's design, different irrigation companies design viz., Gowtham, *et al.*, (2017), Reddy, *et al.*, (2017), Mane, *et al.*, (2014), Bombale, *et al.*, (2011), Design of drip system of irrigation of EPC Irrigation, NETAFIM Irrigation and JAIN Irrigation and results were found satisfactory.

Various researcher's and companies are designing the drip system of irrigation but any one of them like Mane *et al.* (2019) has been selected for testing or validation of the Drip Irrigation App.

#### Model Example Design of drip irrigation system for 1 ha fruit crop (Papaya) using available data:

|                |             |                           |                        |
|----------------|-------------|---------------------------|------------------------|
| 1.Type of crop | = Papaya    | 7.Maximum evaporation     | = 10mm/day             |
| 2.Spacing      | = 6m×6m     | 8. Assume crop factor     | = 0.7                  |
| 3.Size of land | = 100m×100m | 9. Pan coefficient        | = 0.7                  |
| 4.Land slop    | = 0.2%      | 10.Hazen-William constant | = PVC=150,<br>LDPE=140 |
| 5.Static head  | = 12m       | 11. Dripper per plant     | = 4                    |
| 6. Wetted area | = 25%       |                           |                        |

#### Step-1: Total area -

$$\begin{aligned}
 \text{➤ Total area} &= \text{Length of plot (m)} \times \text{Width of plot (m)} \\
 &= 100(\text{m}) \times 100(\text{m}) \\
 &= \mathbf{10000 \text{ m}^2}
 \end{aligned}$$

#### Step -2: Determination of water requirement -

$$\begin{aligned}
 \text{➤ Water requirement} &= \frac{PE \times Pc \times Kc \times \text{wetted area}(\%) \times \text{crop area (m}^2\text{)}}{Eu} \\
 &= \frac{10 \times 0.7 \times 0.7 \times 0.25 \times 6 \times 6}{0.90} \\
 &= \mathbf{49 \text{ liter/day/plant}}
 \end{aligned}$$

**Step- 3: Selection of dripper**

$$\begin{aligned}
 \text{➤ Total number of dripper} &= \frac{\text{Total area (m}^2\text{)} \times \text{no. of dripper per plant}}{S_l \text{ (m)} \times S_d \text{ (m)}} \\
 &= \frac{10000(\text{m}^2) \times 4}{6 \times 6} \\
 &= \mathbf{1112 \text{ Nos.}}
 \end{aligned}$$

$$\begin{aligned}
 \text{➤ Total number of plant} &= \frac{\text{Total area (m}^2\text{)}}{S_l \text{ (m)} \times S_d \text{ (m)}} \\
 &= \frac{10000 \text{ (m}^2\text{)}}{6 \times 6} \\
 &= \mathbf{278}
 \end{aligned}$$

**Step-4: Water application rate and irrigation timing-**

$$\begin{aligned}
 \text{➤ Water application (lph)} &= \text{Dripper discharge(lph)} \times \text{number of dripper per plant} \\
 &= 4 \text{ lph} \times 4 \\
 &= \mathbf{16 \text{ lph}}
 \end{aligned}$$

$$\begin{aligned}
 \text{➤ Irrigation timing (hr)} &= \frac{\text{Water requirement (liter /day/plant)}}{\text{Water application (lph)}} \\
 &= \frac{49}{16} \\
 &= \mathbf{3.062 \text{ hr} = 4 \text{ hr}}
 \end{aligned}$$

**Step-5: Selection and design of lateral-**

$$\begin{aligned}
 \text{➤ Number of dripper on one lateral} &= \frac{\text{Length of lateral (m)} \times \text{number of dripper per plant}}{\text{Dripper spacing (m)}} \\
 &= \frac{50 \times 4}{6} \\
 &= \mathbf{33.33 = 34}
 \end{aligned}$$

➤ **Discharge rate of one lateral (Q<sub>lateral</sub>) (lps) =**

$$\begin{aligned} & \frac{\text{Dripper discharge (lph)} \times \text{number of dripper on one lateral}}{60 \times 60} \\ &= \frac{4 \text{ (lph)} \times 34}{60 \times 60} \\ &= \mathbf{0.035 \text{ lit/sec}} \end{aligned}$$

➤ **Lateral head loss calculation-**

**Head loss of lateral ( $\Delta H$  Latral) = (Hazen and William equation)**

$$\Delta H_{(\text{lateral})} = K \times \left(\frac{Q_{\text{lateral}}}{c}\right)^{1.852} \times (D_{\text{in (lateral)}})^{-4.871} \times (L_1 + L_e) \times F$$

Where,

|                           |   |  |
|---------------------------|---|--|
| K                         | = | 1.21 × 10 <sup>10</sup> (constant value)                                       |
| Q <sub>(lateral)</sub>    | = | Discharge rate of one lateral (lps)  |
| C                         | = | (LLDPE=LDPE=140) (Default value)   |
| D <sub>in (lateral)</sub> | = | Inner diameter of pipe (in mm) (Diameter of lateral-2.2)                       |
| L <sub>1</sub>            | = | Length of lateral for one side of sub-main (m)                                 |
| F                         | = | Outlet Factor (The values of outlet factor for lateral is shown in Table( 3.1) |
| L <sub>e</sub>            | = | Outlet × F-value (outlet= number of dripper on one lateral)                    |

$$= 1.21 \times 10^{10} \times \left(\frac{0.037}{140}\right)^{1.852} \times (9.8)^{-4.871} \times (61.84) \times 37$$

$$= \mathbf{0.87m < 1.1m}$$

**Step 6: Selection and design of sub-main-**

➤ **Number of lateral on sub-main=**

$$\begin{aligned} & \frac{\text{Length of submain (m)} \times \text{No. of side of sub-main}}{\text{Lateral spacing (m)}} \\ &= \frac{100 \times 2}{6} \\ &= \mathbf{33.33 = 34} \end{aligned}$$

➤ **Discharge rate of sub-main (Q<sub>submain</sub>) (lps)=**

$$\begin{aligned} & \text{Discharge rate of one lateral (lps)} \times \text{Number of laterals on sub main} \\ & = 0.035 \times 34 \\ & = \mathbf{1.21 \text{ liter/sec}} \end{aligned}$$

➤ **Head loss of sub-main( $\Delta H_{\text{submain}}$ )(m)= (Hazen and William equation)**

$$\Delta H_{(\text{sub-main})} = K \times \left(\frac{Q_{\text{submain}}}{c}\right)^{1.852} \times (D_{\text{in}(\text{subman})})^{-4.871} \times (L_s + L_e) \times F$$

Where,

$$\begin{aligned} K & = 1.21 \times 10^{10} \text{ (constant value)} \\ Q_{(\text{submain})} & = \text{Discharge rate of sub-main (} Q_{\text{submain}} \text{) (lps)} \\ C & = (\text{PVC=HDPE =150})(\text{Default}) \\ D_{\text{in}(\text{submain})} & = \text{Inner diameter of pipe (in mm) (Diameter of Submain-4)} \\ L_s & = \text{Length of sub-main (} L_s \text{)(m)} \\ F & = \text{F-value (same as earlier)} \\ L_e & = \text{Outlet} \times \text{F-value (outlet= Number of lateral on sub-main)} \\ & = 1.21 \times 10^{10} \times \left(\frac{1.21}{150}\right)^{1.852} \times (47.6)^{-4.871} \times (112.58) \times .37 \\ & = \mathbf{0.53 \text{ m} < \mathbf{1m}} \end{aligned}$$

**Step-7: Selection and design of main line -**

➤ Discharge of sub-main(lps)= Discharge of main(lps)

$$= \mathbf{1.21 \text{ liter/sec}}$$

➤ Head loss of main ( $H_{\text{main}}$ )(m)= (Hazen and William equation)

$$\Delta H (\text{main}) = K \times \left(\frac{Q_{\text{main}}}{c}\right)^{1.852} \times (D_{\text{in}(\text{main})})^{-4.871} \times L(\text{m})$$

Where,

$$\begin{aligned} K & = 1.21 \times 10^{10} \text{ (constant)} \\ Q_{(\text{main})} & = \text{Discharge of main (} Q_{\text{main}} \text{) (lps) = Discharge of sub-main (lps)} \\ C & = (\text{PVC=HDPE =150}) \text{ (Default)} \\ D_{\text{in}(\text{main})} & = \text{Inner diameter of pipe (in mm) (Diameter of main -4)} \\ L_{(\text{m})} & = \text{Length of main (} L_m \text{) (m)} \end{aligned}$$

$$= 1.21 \times 10^{10} \times \left(\frac{1.21}{150}\right)^{1.852} \times (47.6)^{-4.871} \times 50$$

$$= \mathbf{0.5 \text{ m}}$$

**Step-8: Total head loss-**

- **Total head required** = (suction + delivery) + filter loss + mainline loss  
 +operating loss +fitting loss + venture head loss +elevation difference  
 = 12+10+.5+2+2+5  
 = **31.5 m**

**Step-9: Selection and design of pump and filter-**

- **The horsepower is calculated as follows:**

$$\text{HP} = \frac{Q \times H}{75 \times a \times b}$$

Where,

- HP = Horse power  
 Q = Main line discharge  
 a = Efficiency of motor (assumed %)  
 b = Efficiency of motor (assumed %)

$$= \frac{1.21 \times 31.5}{75 \times 0.75 \times 0.85}$$

$$= \mathbf{0.79 = 1 \text{ HP}}$$

- **Selection of filter-**

$$\text{Filter capacity} = Q_{(\text{main})} \times 3.6$$

$$= 1.21 \times 3.6$$

$$= \mathbf{4.356 \text{ m}^3/\text{hr.}}$$

Where,

$$Q_{(\text{main})} = \text{main line discharge}$$

Model example of Mane *et al.* (2019) published book “Principle of Drip Irrigation System” is used to validate the present developed App and found the similar results.

## CHAPTER-V

### SUMMARY AND CONCLUSIONS

---

In the agricultural sector, the amount of water used for irrigation purposes for crop production is high. Water availability problem is being faced in the entire world. Under such situation we need to minimize the wasting of water and develop water saving techniques. For precise utilization of water various techniques such as micro irrigation method, sprinkler method, drip irrigation method etc should be used. However the drip irrigation method are very effective for water saving because drip irrigation system apply controlled and uniform amount of water directly near to the root zone of the plant through emitters at required interval over a long period of time. It improves water use efficiency and reduces water evaporation.

Proper design and installation of drip irrigation system is very important. For proper installation we require basic information, such as length and width of field, slope, soil, metrological data, crop spacing, etc. The design of drip irrigation system is a tedious job for any farmer, scholar and extension worker. There is number of software available to deal with the design related issues but the drawback is the hardware requirement for the same. Presently with the development of Information and Communications Technology (ICT) and awareness among the farmers, android platform based mobile is being readily used and adopted by the farmers.

The aim of the project titled “Development of App for designing of location specific drip irrigation system” was to develop drip irrigation designing system for farmers and extension workers using mobile technology. It is an android based user friendly App that could be used by farmer and extension workers for design drip irrigation system. The first version of android App was designed considering for some selective fruit and vegetable crops. App will calculate some basic designing parameters for drip irrigation system as per user requirement and input data. The developed App helps user to design their field and install whole drip irrigation system and minimizes their extra designing cost and can get higher return.

**Conclusions**

1. Many drip designing method were studied based on the available literature and proper design sequence was establish based on available input data.
2. Appropriate design algorithm was created as blue print for the development of App. The App is user friendly, easy to use, workable with minimum input data and is capable of providing step wise drip designing procedure to all the stakeholders .The App was developed using Android Studio Software and Java language was used for its coding.
3. Design of the system obtained by Drip Irrigation App was tested with past researcher's design and results. From developed App and all the researchers result were found to be perfectly matching.

### **Suggestions for future work**

The developed android App is completed and ready for implementation, but some work has needed to be done for future improvement. So future suggestion of the working is as follows:

#### **1. More crops add**

This App is developed for some fruit and vegetable. More crop cash, flowers, oil seed, forest trees can be added to provide farmers information farming activities of this crop also.

#### **2. Design for sprinkler and micro irrigation system**

This App develops for drip irrigation system designing only, in future suggested App develops for sprinkler or more micro irrigation system.

#### **3. Another language**

This App is developing in English language so it use full for technical people working in drip irrigation companies, scholars and farmers who know English. Similar App cab be developed in Hindi any other regional language.

#### **4. Develop for other platform**

This App is an Android based App. so it operates only on android mobile phone. Similar App can be developing by changed platform that is instead of android windows, IOS or any other platform.

#### **5. Add more information**

This App only for dip design calculation, add more micro irrigation calculation, add verity of soil, emitter information , material quality information , more crop, venture, pump and filter quality etc .

## REFERENCES

---

- Amer, K. H. and Gomaa, A. H. 2003. Uniformity determination in drip irrigation lateral design. *Misr. J. Ag. Engg.*, 20(2): 405-434.
- Boman, B. and Shukla, S. 2004. Hydraulic considerations for citrus microirrigation systems. <http://edis.ifas.ufl.edu>.
- Bombale, V.T., Popale, P.G. and Magar, A. P. 2011. Development of software design of drip irrigation system. *International Journal of Agricultural Engineering*, 4 (2): 170-175.
- Ghanshyam, K., Pooja, K. Pooja, N. and Yogita, G. 2016. AGRONOMY-An android application regarding farmer utility. *J. Engg. Tech. Innov. Res.*, 3(4): 86-89.
- Ghogare, S. A. and Monga, P. M. (2015). E- Agriculture introduction and figuration of its application. *International Journal of Advanced Research in Computer Science and Software Engineering*, 5(1): 44-47.
- Gowtham Deekshithulu, N. V., Babu, G. R., Kumar H. V. H. and Babu, G. R. 2017. Design of computer aided drip irrigation system software. *International Journal of Current Microbiology and Applied Sciences*, 6(12): 108-118.
- Gupta, A., Sarangi, A. and Singh, D. K. 2017. Estimation of crop coefficients and water productivity of mustard (*brassica juncea*) under semi-arid conditions. *Current Science*, 113(2): 264-271.
- Kaur, N. 2016. Development of mobile based agriculture scheduling system for farmers in regional language (Punjabi) using weather conditions. Doctoral dissertation, Punjab Agricultural University, Ludhiana.
- Kumar, A., Kumari, A., Jain, S. K. and Lalita. 2019. Precision farming of different horticultural crops by the application of drip irrigation for the Hi-tech horticulture unit at Pusa. *Journal of Pharmacognosy and Phytochemistry*, 8(4): 1330-1338.
- Kumar, D. S. and Palanisami, K. 2010. Impact of drip irrigation on farming system: evidence from southern India. *Agricultural Economics Research Review*, 23(2): 265-272.

- Kumar, R., Shankar, V. and Kumar, M. 2011. Development of crop coefficients for precise estimation of evapotranspiration for mustard in mid hill zone-India. *Universal Journal of Environmental Research & Technology*, 1(4): 531-538.
- Kumar, R., Trivedi, H., Yadav, R., Das, B. and Bist, A. S. 2016. Effect of drip irrigation on yield and water use efficiency on brinjal (*Solanum melongena*) cv. Pant samrat. *Int. J. Engg. Sci. Res. Tech*, 5(10): 7-17.
- Lomotey, R. K., Chai, Y., Jamal, S. and Deters, R. 2013. MobiCrop: Supporting crop farmers with cloud-enabled mobile app. *Proc 6<sup>th</sup> Service-Oriented Computing and Applications*, (1): 182-89.
- Mane, M. S., Ayare, B. L. and Magar, S. S. 2019. Principles of drip irrigation system. 4<sup>th</sup> Edition, Jain Brothers, New Delhi, pp 41-76.
- Michael, A. M. and Ojha, T. P. 2006. Principles of agricultural engineering, volume II. pp 449-452.
- Mohanty, B., Senapati, S. C. and Sahu, A. P. 2015. Growth, yield and water use efficiency in drip irrigated brinjal (*Solanum melongena L.*) as affected by single and double inlet drip laterals. *International Journal of Agricultural Engineering*, 8(2): 144-150.
- Narayanan, R., Steele, D. D. and Scherer, T. F. 2002. Computer model to optimize above ground drip irrigation systems for small areas. *Journal of Applied Engineering in Agriculture*, 18(4): 459-469.
- Perold, R. P. 1977. Design of irrigation pipe laterals with multiple outlets. *Journal of the Irrigation and Drainage Division*, 103(2): 179-195.
- Pradhan, S., Sehgal . V. K., Das, D. K. Bhandyopadhyay, K. K. and Singh R. 2013. Evaluation of pan coefficient method for estimating FAO-56 reference crop evapotranspiration in a semi arid environment. *Journal of Agrometeorology*, 15(1): 90-93.
- Rares, H. C. Z. 2009. Designing a drip irrigation system using hydrocalc irrigation planning. *Research Journal of Agricultural Science*, 41(1): 420-425.

- Reddy, K. H. N., Kumar, K. A. and Ramana, M. 2017. Design and development of drip irrigation system software in visual basic. *International Journal of Agricultural Science and Research*, 7(4): 339-346.
- Reddy, Y. K., Adamala, S. and Babu, B. H. 2017. Case study on performance evaluation of drip irrigation systems in selected villages of Guntur district, Andhra Pradesh, India. *Int. J. Curr. Microbiol. App. Sci.*, 6(2): 437-445
- Sentelhas, P. C. and Folegatti, M. V. 2003. Class A pan coefficients (Kp) to estimate daily reference evapotranspiration (ET<sub>o</sub>). *Revista Brasileira de Engenharia Agrícola e Ambiental*, 7(1): 111-115.
- Sharma, S. Sharma, D.K. and Sharma, S. 2018. Overview of mobile android agriculture applications. *International Research Journal of Engineering and Technology*, 5(8): 225-231.
- Sharma, S., Patodkar, V., Simant, S. and Shah, C. 2015. E-Agro Android Application (Integrated Farming Management Systems for sustainable development of farmers). *Int. J. Engg. Res. Gen. Sci.*, 3(1): 368-372.
- Silva, W. A. D., Santana, J. S., Silva, C. M. D. and Nunes, A. A. 2017. Crop Coefficient Regionalization for Irrigated Agriculture Planning in Maranhao State-Brazil, *Engenharia Agrícola*, 37(5): 953-960.
- Srinivas, B. and Tiwari, K. N. 2018. Determination of crop water requirement and crop coefficient at different growth stages of green gram crop by using non-weighting lysimeter. *Int. J. Curr. Microbiol. App. Sci.*, 7(9): 2580-2589.

## APPENDIX-I

### User\_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16sp"
    tools:context=".user">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="User Information"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textSize="18sp" />

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <EditText
            android:id="@+id/user_name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:hint="Name of Farmer/Scholar "
            android:inputType="textAutoComplete"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <EditText
            android:id="@+id/institute"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Institute"
            android:inputType="textAutoComplete"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <EditText
            android:id="@+id/city"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Village/City"
            android:inputType="textAutoComplete"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>
</LinearLayout>
```

```

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/mob_no"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Contact No."
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/khasra"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Khasra No."
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/next_user"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="270dp"
    android:text="Next"
    android:textSize="10dp" />

```

```
</LinearLayout>
```

### **User\_activity.java**

```

package com.bawv.dirp irrigation;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

public class user extends AppCompatActivity {
    EditText user_name,institute,city,mob_no,khasra;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_user);
    Button next_main = findViewById(R.id.next_user);

    ActivityCompat.requestPermissions(this, new String[]
        {Manifest.permission.WRITE_EXTERNAL_STORAGE},
        PackageManager.PERMISSION_GRANTED);

    user_name = findViewById(R.id.user_name);
    institute = findViewById(R.id.institute);
    city = findViewById(R.id.city);
    mob_no = findViewById(R.id.mob_no);
    khasra = findViewById(R.id.khasra);

    next_main.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            if ( (TextUtils.isEmpty(user_name.getText().toString()) ||
                (TextUtils.isEmpty(institute.getText().toString()) ||
                (TextUtils.isEmpty(city.getText().toString()) ||
                (TextUtils.isEmpty(mob_no.getText().toString()) ||
                (TextUtils.isEmpty(khasra.getText().toString())
            )
            { Toast.makeText(user.this, "Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
            }
            else{
                String u_name = user_name.getText().toString().trim();
                String inst_e = institute.getText().toString().trim();
                String cit_y = city.getText().toString().trim();
                int mob = Integer.parseInt(mob_no.getText().toString());
                int ks_ra = Integer.parseInt(khasra.getText().toString());
                Intent user = new Intent(user.this, MainActivity.class);
                user.putExtra("data_user", u_name);
                user.putExtra("data_inst", inst_e);
                user.putExtra("data_city", cit_y);
                user.putExtra("data_mob", mob);
                user.putExtra("data_ks_ra", ks_ra);
                startActivity(user);
            }
        }
    });
}

```

### Field informariom\_activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"

```

```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16sp"
tools:context=".MainActivity">

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Field information"
    android:textAppearance="@style/TextAppearance.AppCompat.Body2"
    android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/crop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Name of Crop"
        android:inputType="textAutoComplete"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/field_length"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Length of plot (m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/field_width"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Width of plot (m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"

```

```

    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/slope"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Slope (%)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_area"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Total Area" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/area_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" mÅ²" />
</LinearLayout>

<Button
    android:id="@+id/next_main"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="270dp"
    android:text="Next"
    android:textSize="10dp" />

</LinearLayout>

```

### Field information\_activity.java

```

package com.bawv.dirp irrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;

```

```

import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText field_length, field_width, crop;
    TextView result;
    double total_area;
    double length, width;
    Button btn_area;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Bundle area = getIntent().getExtras();
        final String data_user = area.getString("data_user");
        final String data_inst = area.getString("data_inst");
        final String data_city = area.getString("data_city");
        final int data_mob = area.getInt("data_mob");
        final int data_ks_ra = area.getInt("data_ks_ra");

        field_length = findViewById(R.id.field_length);
        field_width = findViewById(R.id.field_width);
        crop = findViewById(R.id.crop);
        btn_area = findViewById(R.id.btn_area);
        result = findViewById(R.id.area_result);

        btn_area.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if ( (TextUtils.isEmpty(field_length.getText().toString()) ||
                    (TextUtils.isEmpty(field_width.getText().toString())
                    )
                )
                {
                    Toast.makeText(MainActivity.this, "Fields can't be Empty",
                        Toast.LENGTH_LONG)
                        .show();
                }
                else
                {
                    length = Double.parseDouble(field_length.getText().toString());
                    width = Double.parseDouble(field_width.getText().toString());
                    // for total area
                    total_area = length * width;
                    result.setText(String.valueOf(total_area));
                }
            }
        });

        Button next_main = findViewById(R.id.next_main);

```

```

next_main.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if ( (TextUtils.isEmpty(field_length.getText().toString()) ||
            (TextUtils.isEmpty(field_width.getText().toString()) ||
            (TextUtils.isEmpty(crop.getText().toString())
            )
            {
                Toast.makeText(MainActivity.this,"Fields can't be Empty",
                    Toast.LENGTH_LONG)
                    .show();
            }
            else
            {
                String crop_ = crop.getText().toString().trim();
                double d_length = Double.parseDouble(field_length.getText().toString());
                double d_width = Double.parseDouble(field_width.getText().toString());
                double d_area = Double.parseDouble(result.getText().toString());
                Intent intent = new Intent(MainActivity.this, Water_req.class);
                intent.putExtra("data_user", data_user);
                intent.putExtra("data_inst", data_inst);
                intent.putExtra("data_city", data_city);
                intent.putExtra("data_ks_ra", data_ks_ra);
                intent.putExtra("data_mob", data_mob);
                intent.putExtra("data_length", d_length);
                intent.putExtra("data_width", d_width);
                intent.putExtra("data_area", d_area);
                intent.putExtra("data_crop", crop_);
                startActivity(intent);
            }
        }
    });
}
}

/*<Spinner
    android:id="@+id/veg"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:entries="@array/veg"
    android:scrollbarSize="2dp" />*/

```

### Water requirement\_activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16sp"
    tools:context=".Water_req">

    <TextView
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="@string/water_requirement"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/pe"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="21dp"
        android:hint="LPan evaporation (PE)(mm/day)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/kc"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Crop coefficient (Kc)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/wa"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/wetted_area"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/sl"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/sl"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout

```

```

    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/sd"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/sd"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/pc"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Pan coefficient (Pc)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/eu"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Emission uniformity of drip (Eu)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_water_req"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Water requirement" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/water_req_result"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=" (liter/day/plant)" />
</LinearLayout>

<Button
    android:id="@+id/next_water"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="270dp"
    android:text="Next"
    android:textSize="10dp" />

</LinearLayout>

```

### **Water requirement \_activity.java**

```

package com.bawv.dirprrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Water_req extends AppCompatActivity {

    TextView result;
    EditText pe, kc, wa, sl, sd, pc, eu;
    double pe_ ,kc_ ,sl_ ,sd_ ,wa_ ,pc_ ,eu_ ;
    double water_req;
    Button btn_water_req;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_water_req);

        Bundle water = getIntent().getExtras();
        final String data_user = water.getString("data_user");
        final String data_inst= water.getString("data_inst");
        final String data_city = water.getString("data_city");
        final int data_mob = water.getInt("data_mob");
        final int data_ks_ra = water.getInt("data_ks_ra");
        final double d_length = water.getDouble("data_length");
        final double d_width = water.getDouble("data_width");
        final double total_area = water.getDouble("data_area");
        final String data_crop = water.getString("data_crop");

```

```

pe = findViewById(R.id.pe);
kc = findViewById(R.id.kc);
wa = findViewById(R.id.wa);
sl = findViewById(R.id.sl);
sd = findViewById(R.id.sd);
pc = findViewById(R.id.pc);
eu = findViewById(R.id.eu);
btn_water_req = findViewById(R.id.btn_water_req);
result = findViewById(R.id.water_req_result);

btn_water_req.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if ( (TextUtils.isEmpty(pe.getText().toString()) ||
            (TextUtils.isEmpty(kc.getText().toString()) ||
            (TextUtils.isEmpty(wa.getText().toString()) ||
            (TextUtils.isEmpty(sl.getText().toString()) ||
            (TextUtils.isEmpty(sd.getText().toString()) ||
            (TextUtils.isEmpty(pc.getText().toString()) ||
            (TextUtils.isEmpty(eu.getText().toString())
            )
        ) {
            Toast.makeText(Water_req.this, "Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
        else {
            pe_ = Double.parseDouble(pe.getText().toString());
            kc_ = Double.parseDouble(kc.getText().toString());
            wa_ = Double.parseDouble(wa.getText().toString());
            sl_ = Double.parseDouble(sl.getText().toString());
            sd_ = Double.parseDouble(sd.getText().toString());
            pc_ = Double.parseDouble(pc.getText().toString());
            eu_ = Double.parseDouble(eu.getText().toString());
            // for Water requirement
            water_req = (pe_ * kc_ * wa_ * sl_ * sd_ * pc_) / eu_ ;
            result.setText(String.valueOf(water_req));
        }
    }
});

Button next_water = findViewById(R.id.next_water);
next_water.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if ( (TextUtils.isEmpty(pe.getText().toString()) ||
            (TextUtils.isEmpty(kc.getText().toString()) ||
            (TextUtils.isEmpty(wa.getText().toString()) ||
            (TextUtils.isEmpty(sl.getText().toString()) ||
            (TextUtils.isEmpty(sd.getText().toString()) ||
            (TextUtils.isEmpty(pc.getText().toString()) ||
            (TextUtils.isEmpty(eu.getText().toString())
            )
        ) {
            Toast.makeText(Water_req.this, "Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
    }
}

```

```

else {
double d_sl = Double.parseDouble(sl.getText().toString());
double d_sd = Double.parseDouble(sd.getText().toString());
double d_wr = Double.parseDouble(result.getText().toString());
Intent wr_intent = new Intent(Water_req.this,Selection_of_dipper.class);
wr_intent.putExtra("data_user", data_user);
wr_intent.putExtra("data_inst", data_inst);
wr_intent.putExtra("data_city", data_city);
wr_intent.putExtra("data_ks_ra", data_ks_ra);
wr_intent.putExtra("data_mob", data_mob);
wr_intent.putExtra("data_water",d_wr);
wr_intent.putExtra("data_sl",d_sl);
wr_intent.putExtra("data_sd",d_sd);
wr_intent.putExtra("total_area",total_area);
wr_intent.putExtra("d_length",d_length);
wr_intent.putExtra("d_width",d_width);
wr_intent.putExtra("data_crop",data_crop);
startActivity(wr_intent);
}}});
}
}

```

#### Selection of dripper\_activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16sp"
tools:context=".Selection_of_dipper">

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Selection of dripper"
android:textAppearance="@style/TextAppearance.AppCompat.Body2"
android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
app:helperText="Select from 2,4,8,10"
app:helperTextEnabled="true">

<EditText
android:id="@+id/lph"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Dripper discharge (lph)"
android:inputType="numberDecimal"
android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout

```

```

    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:helperText="Select from 1 to 6"
    app:helperTextEnabled="true">

    <EditText
        android:id="@+id/dpp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Number of dripper per plant"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_dipper1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Total number of Plant"

    />

<TextView
    android:id="@+id/no_of_plant_result"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Total number of Dipper" />

<TextView
    android:id="@+id/no_of_dipper_result"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Water application" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/water_app_result"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=" lph" />
</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Irrigation timing" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/irri_timing_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" hr" />
</LinearLayout>

<Button
    android:id="@+id/next_dipper"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="270dp"
    android:text="Next"
    android:textSize="10sp" />

</LinearLayout>

```

### **Selection of dripper\_activity.java**

```

package com.bawv.dirprrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Selection_of_dipper extends AppCompatActivity {

```

```

Button dip1;
TextView res1,res2,res3,res4;
EditText lph,dpp;
double lph_,dpp_;
double nop,n_dpp, wa, it;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_selection_of_dripper);
    Bundle dipper = getIntent().getExtras();
    final String data_user = dipper.getString("data_user");
    final String data_inst= dipper.getString("data_inst");
    final String data_city = dipper.getString("data_city");
    final int data_mob = dipper.getInt("data_mob");
    final int data_ks_ra = dipper.getInt("data_ks_ra");
    final double sl = dipper.getDouble("data_sl");
    final double sd = dipper.getDouble("data_sd");
    final double total_area = dipper.getDouble("total_area");
    final double data_length = dipper.getDouble("d_length");
    final double data_Width = dipper.getDouble("d_width");
    final double data_water = dipper.getDouble("data_water");
    final String data_crop = dipper.getString("data_crop");

    lph = findViewById(R.id.lph);
    dpp = findViewById(R.id.dpp);
    dip1 =findViewById(R.id.btn_dipper1);
    res1 = findViewById(R.id.no_of_plant_result);
    res2 = findViewById(R.id.no_of_dipper_result);
    res3 =findViewById(R.id.water_app_result);
    res4 = findViewById(R.id.irri_timing_result);

    dip1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if ( (TextUtils.isEmpty(lph.getText().toString()) ||
                (TextUtils.isEmpty(dpp.getText().toString())
            )
            )
            {
                Toast.makeText(Selection_of_dipper.this,"Fields can't be Empty",
                    Toast.LENGTH_LONG)
                    .show();
            }

            else
            {
                // for Total number of plant
                nop = total_area/(sl * sd);
                res1.setText(String.valueOf(nop));

                // for Total number of dripper
                dpp_ = Double.parseDouble(dpp.getText().toString());
                n_dpp = nop * dpp_;
                res2.setText(String.valueOf(n_dpp));

                // for Water application

```

```

lph_ = Double.parseDouble(lph.getText().toString());
wa = lph_ * dpp_;
res3.setText(String.valueOf(wa));

// Irrigation timing
it = wa / lph_;
res4.setText(String.valueOf(it));

}
});

Button next_dipper = findViewById(R.id.next_dipper);
next_dipper.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(lph.getText().toString()) ||
            (TextUtils.isEmpty(dpp.getText().toString())
        )
        {
            Toast.makeText(Selection_of_dipper.this,"Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();

        }

        else
        {
            double d_dpp = Double.parseDouble(dpp.getText().toString());
            double d_lph = Double.parseDouble(lph.getText().toString());
            double d_tnop = Double.parseDouble(res1.getText().toString());
            double d_tnod = Double.parseDouble(res2.getText().toString());
            double d_wa = Double.parseDouble(res3.getText().toString());
            double d_it = Double.parseDouble(res4.getText().toString());
            Intent dipper = new Intent(Selection_of_dipper.this,Selection_of_lateral.class);
            dipper.putExtra("data_user", data_user);
            dipper.putExtra("data_inst", data_inst);
            dipper.putExtra("data_city", data_city);
            dipper.putExtra("data_ks_ra", data_ks_ra);
            dipper.putExtra("data_mob", data_mob);
            dipper.putExtra("data_length",data_length);
            dipper.putExtra("data_Width",data_Width);
            dipper.putExtra("total_area",total_area);
            dipper.putExtra("data_crop",data_crop);
            dipper.putExtra("data_sl",sl);
            dipper.putExtra("data_sd",sd);
            dipper.putExtra("data_water",data_water);
            dipper.putExtra("data_dpp",d_dpp);
            dipper.putExtra("data_lph",d_lph);
            dipper.putExtra("data_tnop",d_tnop);
            dipper.putExtra("data_tnod",d_tnod);
            dipper.putExtra("data_wa",d_wa);
            dipper.putExtra("data_it",d_it);

            startActivity(dipper);
        }
    }
});
}
}
}

```

**Selection of lateral\_activity.xml**

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16sp"
    tools:context=".Selection_of_lateral">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Selection of Lateral"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/spinner_pipe_material"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:entries="@array/spinner_pm" />

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:helperText="Select from 12,16,20,25,32,40"
        app:helperTextEnabled="true">

        <EditText
            android:id="@+id/dia_lat"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Diameter of lateral(mm)"
            android:inputType="numberDecimal"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:helperText="Select 1 for End or 2 for Centre">

        <EditText
            android:id="@+id/length_lat"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Length of lateral"
            android:inputType="numberDecimal"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>

    <Button
        android:id="@+id/btn_lateral1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Calculate"

```

```

        android:textSize="8dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Length of later for one side
of sub-main" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/one_side_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" m" />
</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Number of dripper on one lateral" />

<TextView
    android:id="@+id/no_dip_lat_result"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Discharge rate of one lateral (Q lateral)" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/discharge_lat_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" lps" />
</LinearLayout>

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Head loss of lateral (â†‡H Latral)" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/head_loss_lat_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" m" />
</LinearLayout>

<Button
    android:id="@+id/next_lateral"
    android:layout_width="wrap_content"
    android:layout_height="38dp"
    android:layout_marginLeft="270dp"
    android:text="Next"
    android:textSize="8dp" />

</LinearLayout>

```

### Selection of lateral\_activty.java

```

package com.bawv.dirpirrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Selection_of_lateral extends AppCompatActivity {
    Button lat1;
    TextView res1,res2,res3,res4;
    EditText dia_lat, side;
    double dia_lat_, len_lat_, side_, no_dip_lat,dis_rate, hl_lat;
    double dsd = 60 * 60;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_selection_of_lateral);
    }
}

```

```

Bundle lateral = getIntent().getExtras();
final String data_user = lateral.getString("data_user");
final String data_inst= lateral.getString("data_inst");
final String data_city = lateral.getString("data_city");
final int data_mob = lateral.getInt("data_mob");
final int data_ks_ra = lateral.getInt("data_ks_ra");
final double total_area = lateral.getDouble("total_area");
final double data_length = lateral.getDouble("data_length");
final double data_Width = lateral.getDouble("data_Width");
final String data_crop = lateral.getString("data_crop");
final double data_sl = lateral.getDouble("data_sl");
final double data_sd = lateral.getDouble("data_sd");
final double data_water= lateral.getDouble("data_water");
final double data_dpp = lateral.getDouble("data_dpp");
final double data_lph = lateral.getDouble("data_lph");
final double data_tnop = lateral.getDouble("data_tnop");
final double data_tnod= lateral.getDouble("data_tnod");
final double data_wa = lateral.getDouble("data_wa");
final double data_it = lateral.getDouble("data_it");

dia_lat =findViewById(R.id.dia_lat);
side = findViewById(R.id.length_lat);
lat1 =findViewById(R.id.btn_lateral1);
res1 = findViewById(R.id.one_side_result);
res2 = findViewById(R.id.no_dip_lat_result);
res3 = findViewById(R.id.discharge_lat_result);
res4 = findViewById(R.id.head_loss_lat_result);

lat1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(dia_lat.getText().toString()) ||
            (TextUtils.isEmpty(side.getText().toString())
        )
        {
            Toast.makeText(Selection_of_lateral.this,"Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
        else{
            side_ = Double.parseDouble(side.getText().toString());
            // for Length of later
            len_lat_ = data_length / side_;
            res1.setText(String.valueOf(len_lat_));

            // for Number of dripper on one lateral
            no_dip_lat = (len_lat_/data_sl) * data_dpp;
            res2.setText(String.valueOf(no_dip_lat));

            dis_rate = (data_lph * no_dip_lat) / dsd;
            // for Discharge rate of one lateral
            res3.setText(String.valueOf(dis_rate));

            dia_lat_ = Double.parseDouble(dia_lat.getText().toString());
            double k = 1.21*Math.pow(10,10);
            double dia_in = dia_lat_-2.2;
            double f = 0.36;
            double le = no_dip_lat * f;

```

```

double out = len_lat_ + le;
double ql1 = dis_rate/140;
double ql2 = Math.pow(ql1,1.852);
double din2 = Math.pow(dia_in,-4.871);

// for Head loss of lateral
hl_lat = k*ql2*din2*out*f;

res4.setText(String.valueOf(hl_lat));
if ( hl_lat < 1)
{
    Toast.makeText(Selection_of_lateral.this,"Head loss < 1m Acceptable",
        Toast.LENGTH_LONG)
        .show();
}
else{ Toast.makeText(Selection_of_lateral.this,"Head loss > 1m Not Acceptable in case(change
diameter of lateral)",
    Toast.LENGTH_LONG)
    .show();

}
}}});

Button next_lateral = findViewById(R.id.next_lateral);
next_lateral.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(dia_lat.getText().toString()) ||
            (TextUtils.isEmpty(side.getText().toString())
        )
        {
            Toast.makeText(Selection_of_lateral.this,"Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
        else{
            double data_dia_lat = Double.parseDouble(dia_lat.getText().toString());
            double data_one_side_sm = Double.parseDouble(res1.getText().toString());
            double data_nod_lat = Double.parseDouble(res2.getText().toString());
            double data_dis_rate = Double.parseDouble(res3.getText().toString());
            double data_dis_hl_lat = Double.parseDouble(res4.getText().toString());
            Intent lateral =new Intent(Selection_of_lateral.this,Selection_of_submain.class);
            lateral.putExtra("data_user", data_user);
            lateral.putExtra("data_inst", data_inst);
            lateral.putExtra("data_city", data_city);
            lateral.putExtra("data_ks_ra", data_ks_ra);
            lateral.putExtra("data_mob", data_mob);
            lateral.putExtra("data_length",data_length);
            lateral.putExtra("data_Width",data_Width);
            lateral.putExtra("total_area",total_area);
            lateral.putExtra("data_crop",data_crop);
            lateral.putExtra("data_sl",data_sl);
            lateral.putExtra("data_sd",data_sd);
            lateral.putExtra("data_water",data_water);
            lateral.putExtra("data_dpp",data_dpp);
            lateral.putExtra("data_lph",data_lph);
            lateral.putExtra("data_tnop",data_tnop);

```

```

        lateral.putExtra("data_tnod",data_tnod);
        lateral.putExtra("data_wa",data_wa);
        lateral.putExtra("data_it",data_it);
        lateral.putExtra("data_dia_lat",data_dia_lat);
        lateral.putExtra("data_one_side_sm",data_one_side_sm);
        lateral.putExtra("data_nod_lat",data_nod_lat);
        lateral.putExtra("data_dis_rate",data_dis_rate);
        lateral.putExtra("data_dis_hl_lat",data_dis_hl_lat);
        startActivity(lateral);
    }
    });
}
}
}

```

### Selection of sub-main\_ activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16sp"
    tools:context=".Selection_of_submain">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Selection of Sub-main"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/spinner_sub_pipe_material"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:entries="@array/spinner_pmsm" />

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:helperText="Select from 40,63,75,90,110"
        app:helperTextEnabled="true"
        >

        <EditText
            android:id="@+id/dia_sm"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Diameter of sub-main(mm)"
            android:inputType="numberDecimal"
            android:textSize="10dp" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

app:helperTextEnabled="true"
app:helperText="Select 1 for End or 2 for Centre">

<EditText
    android:id="@+id/dia_lat_sm"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Number of lateral on sub-main"
    android:inputType="numberDecimal"
    android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/sm_side"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Length of sub-main(Ls)(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_sml"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Number of lateral on sub-main"

    />

<TextView
    android:id="@+id/no_lat_sm_result"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Discharge rate of sub-main (Q submain)" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/no_dip_lat_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

        android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=" lps" />
</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Head loss of sub-main (â†H submain)" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/hl_sm_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" m" />
</LinearLayout>

<Button
    android:id="@+id/next_submain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="270dp"
    android:text="Next"
    android:textSize="10dp" />

</LinearLayout>

```

### Selection of sub-main\_activity.java

```

package com.bawv.dirp irrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Selection_of_submain extends AppCompatActivity {

    Button sml;

```

```

TextView res1,res2,res3;
EditText dia_sm,len_sm,side;
double dia_sm_len_sm_side_;
double no_lat_sm, dis_rate_sm,hl_sm;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_selection_of_submain);
    final Bundle sub_main = getIntent().getExtras();
    final String data_user = sub_main.getString("data_user");
    final String data_inst= sub_main.getString("data_inst");
    final String data_city = sub_main.getString("data_city");
    final int data_mob = sub_main.getInt("data_mob");
    final int data_ks_ra = sub_main.getInt("data_ks_ra");
    final double total_area = sub_main.getDouble("total_area");
    final double data_length = sub_main.getDouble("data_length");
    final double data_Width = sub_main.getDouble("data_Width");
    final String data_crop = sub_main.getString("data_crop");
    final double data_sl = sub_main.getDouble("data_sl");
    final double data_sd = sub_main.getDouble("data_sd");
    final double data_water= sub_main.getDouble("data_water");
    final double data_dpp = sub_main.getDouble("data_dpp");
    final double data_lph = sub_main.getDouble("data_lph");
    final double data_tnop = sub_main.getDouble("data_tnop");
    final double data_tnod= sub_main.getDouble("data_tnod");
    final double data_wa = sub_main.getDouble("data_wa");
    final double data_it = sub_main.getDouble("data_it");
    final double data_dia_lat = sub_main.getDouble("data_dia_lat");
    final double data_one_side_sm = sub_main.getDouble("data_one_side_sm");
    final double data_nod_lat = sub_main.getDouble("data_nod_lat");
    final double data_dis_rate = sub_main.getDouble("data_dis_rate");
    final double data_dis_hl_lat = sub_main.getDouble("data_dis_hl_lat");
    dia_sm =findViewById(R.id.dia_sm);
    len_sm =findViewById(R.id.dia_lat_sm);
    side =findViewById(R.id.sm_side);
    sm1 =findViewById(R.id.btn_sm1);
    res1 = findViewById(R.id.no_lat_sm_result);
    res2 = findViewById(R.id.no_dip_lat_result);
    res3 = findViewById(R.id.hl_sm_result);

    sm1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            if ( (TextUtils.isEmpty(dia_sm.getText().toString()) ||
                (TextUtils.isEmpty(len_sm.getText().toString()) ||
                (TextUtils.isEmpty(side.getText().toString())
            )
            {
                Toast.makeText(Selection_of_submain.this,"Fields can't be Empty",
                    Toast.LENGTH_LONG)
                    .show();
            }
            else{
                len_sm_ = Double.parseDouble(len_sm.getText().toString());
                side_ = Double.parseDouble(side.getText().toString());
            }
        }
    });
}

```

```

// for Number of lateral on sub-main
no_lat_sm = (len_sm_/ data_sl) * side_;
res1.setText(String.valueOf(no_lat_sm));

// for Discharge rate of sub-main
dis_rate_sm = data_dis_rate * no_lat_sm;
res2.setText(String.valueOf(dis_rate_sm));

dia_sm_ = Double.parseDouble(dia_sm.getText().toString());
double k = 1.21 * Math.pow(10,10);
double f = .36;
double q = dis_rate_sm/150;
double qlt = Math.pow(q,1.852);
double dia = dia_sm_ - 4;
double dia_in = Math.pow(dia,-4.871);
double le = no_lat_sm * f;
double out = side_ + le;

// for Head loss of sub-main
hl_sm = k * qlt * dia_in * out * f;

res3.setText(String.valueOf(hl_sm));
if ( hl_sm < 1)
{
    Toast.makeText(Selection_of_submain.this,"Head loss < 1m Acceptable",
        Toast.LENGTH_LONG)
        .show();
}
else{ Toast.makeText(Selection_of_submain.this,"Head loss > 1m Not Acceptable",
    Toast.LENGTH_LONG)
    .show();
}
}
});

Button next_submain = findViewById(R.id.next_submain);
next_submain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(dia_sm.getText().toString()) ||
            (TextUtils.isEmpty(len_sm.getText().toString()) ||
            (TextUtils.isEmpty(side.getText().toString())
        )
        {
            Toast.makeText(Selection_of_submain.this,"Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
        else{
            double data_dia_sm = Double.parseDouble(dia_sm.getText().toString());
            double data_len_sm = Double.parseDouble(side.getText().toString());
            double data_nol_sm = Double.parseDouble(res1.getText().toString());
            double data_dis_rate_sm = Double.parseDouble(res2.getText().toString());
            double data_hl_sm = Double.parseDouble(res3.getText().toString());
            Intent sub_main =new Intent(Selection_of_submain.this,Selection_of_main.class);

```

```

        sub_main.putExtra("data_user", data_user);
        sub_main.putExtra("data_inst", data_inst);
        sub_main.putExtra("data_city", data_city);
        sub_main.putExtra("data_ks_ra", data_ks_ra);
        sub_main.putExtra("data_mob", data_mob);
        sub_main.putExtra("data_length",data_length);
        sub_main.putExtra("data_Width",data_Width);
        sub_main.putExtra("total_area",total_area);
        sub_main.putExtra("data_crop",data_crop);
        sub_main.putExtra("data_sl",data_sl);
        sub_main.putExtra("data_sd",data_sd);
        sub_main.putExtra("data_water",data_water);
        sub_main.putExtra("data_dpp",data_dpp);
        sub_main.putExtra("data_lph",data_lph);
        sub_main.putExtra("data_tnop",data_tnop);
        sub_main.putExtra("data_tnod",data_tnod);
        sub_main.putExtra("data_wa",data_wa);
        sub_main.putExtra("data_it",data_it);
        sub_main.putExtra("data_dia_lat",data_dia_lat);
        sub_main.putExtra("data_one_side_sm",data_one_side_sm);
        sub_main.putExtra("data_nod_lat",data_nod_lat);
        sub_main.putExtra("data_dis_rate",data_dis_rate);
        sub_main.putExtra("data_dis_hl_lat",data_dis_hl_lat);
        sub_main.putExtra("data_dia_sm",data_dia_sm);
        sub_main.putExtra("data_len_sm",data_len_sm);
        sub_main.putExtra("data_nol_sm",data_nol_sm);
        sub_main.putExtra("data_dis_rate_sm",data_dis_rate_sm);
        sub_main.putExtra("data_hl_sm",data_hl_sm);
        startActivity(sub_main);
    }
    });
}
}
}

```

#### Selection of main\_activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16sp"
    tools:context=".Selection_of_main">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Selection of Main"
        android:textAppearance="@style/TextAppearance.AppCompat.Body2"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/spinner_main_pipe_material"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:entries="@array/spinner_pmm" />

```

```

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:helperText="Select from 75,90,100,110,125,140"
    app:helperTextEnabled="true">

    <EditText
        android:id="@+id/dia_main"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Diameter of Main line(mm)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/main_length"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Length of main (Lmain)(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_main"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="Head loss of main ( $\hat{a}^{\dagger}H$  main)" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/hl_main_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" m" />
</LinearLayout>

```

```

<Button
    android:id="@+id/next_main"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="270dp"
    android:text="Next"
    android:textSize="10dp"

/></LinearLayout>

```

### Selection of main\_activity.java

```

package com.bawv.dirpirrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Selection_of_main extends AppCompatActivity {

    Button btn;
    TextView res;
    EditText dia_m,len_m;
    double dia_m_,len_m_;
    double hlm;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_selection_of_main);
        Bundle main = getIntent().getExtras();
        final String data_user = main.getString("data_user");
        final String data_inst= main.getString("data_inst");
        final String data_city = main.getString("data_city");
        final int data_mob = main.getInt("data_mob");
        final int data_ks_ra = main.getInt("data_ks_ra");
        final double total_area = main.getDouble("total_area");
        final double data_length = main.getDouble("data_length");
        final double data_Width = main.getDouble("data_Width");
        final String data_crop = main.getString("data_crop");
        final double data_sl = main.getDouble("data_sl");
        final double data_sd = main.getDouble("data_sd");
        final double data_water= main.getDouble("data_water");
        final double data_dpp = main.getDouble("data_dpp");
        final double data_lph = main.getDouble("data_lph");
        final double data_tnop = main.getDouble("data_tnop");
        final double data_tnod= main.getDouble("data_tnod");
        final double data_wa = main.getDouble("data_wa");
        final double data_it = main.getDouble("data_it");
        final double data_dia_lat = main.getDouble("data_dia_lat");
    }
}

```

```

final double data_one_side_sm = main.getDouble("data_one_side_sm");
final double data_nod_lat = main.getDouble("data_nod_lat");
final double data_dis_rate = main.getDouble("data_dis_rate");
final double data_dis_hl_lat = main.getDouble("data_dis_hl_lat");
final double data_dia_sm = main.getDouble("data_dia_sm");
final double data_len_sm = main.getDouble("data_len_sm");
final double data_nol_sm = main.getDouble("data_nol_sm");
final double data_dis_rate_sm = main.getDouble("data_dis_rate_sm");
final double data_hl_sm = main.getDouble("data_hl_sm");

dia_m = findViewById(R.id.dia_main);
len_m = findViewById(R.id.main_length);
btn = findViewById(R.id.btn_main);

res = findViewById(R.id.hl_main_result);

btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(dia_m.getText().toString()) ||
            (TextUtils.isEmpty(len_m.getText().toString())
        )
        )
        {
            Toast.makeText(Selection_of_main.this,"Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();

        }
        else{

            dia_m_ = Double.parseDouble(dia_m.getText().toString());
            len_m_ = Double.parseDouble(len_m.getText().toString());
            double k = 1.21 * Math.pow(10,10);
            double q = data_dis_rate_sm/150;
            double qlt = Math.pow(q,1.852);
            double dia = dia_m_ - 4;
            double dia_in = Math.pow(dia,-4.871);
            // for Head loss of main
            hlm = k * qlt * dia_in * len_m_;
            res.setText(String.valueOf(hlm));
            if ( hlm < 2)
            {
                Toast.makeText(Selection_of_main.this,"Head loss < 2m Acceptable",
                    Toast.LENGTH_LONG)
                    .show();

            }
            else{ Toast.makeText(Selection_of_main.this,"Head loss > 2m Not Acceptable",
                Toast.LENGTH_LONG)
                .show();

            }

        }
    }
});

Button next_main = findViewById(R.id.next_main);
next_main.setOnClickListener(new View.OnClickListener() {
    @Override

```

```

public void onClick(View v) {
    if ((TextUtils.isEmpty(dia_m.getText().toString()) ||
        (TextUtils.isEmpty(len_m.getText().toString())))
    ) {
        Toast.makeText(Selection_of_main.this, "Fields can't be Empty",
            Toast.LENGTH_LONG)
            .show();
    } else {
        double data_dia_m = Double.parseDouble(dia_m.getText().toString());
        double data_len_m = Double.parseDouble(len_m.getText().toString());
        double data_hl_m = Double.parseDouble(res.getText().toString());
        Intent main = new Intent(Selection_of_main.this, Selection_of_pump.class);
        main.putExtra("data_user", data_user);
        main.putExtra("data_inst", data_inst);
        main.putExtra("data_city", data_city);
        main.putExtra("data_ks_ra", data_ks_ra);
        main.putExtra("data_mob", data_mob);
        main.putExtra("data_length", data_length);
        main.putExtra("data_Width", data_Width);
        main.putExtra("total_area", total_area);
        main.putExtra("data_crop", data_crop);
        main.putExtra("data_sl", data_sl);
        main.putExtra("data_sd", data_sd);
        main.putExtra("data_water", data_water);
        main.putExtra("data_dpp", data_dpp);
        main.putExtra("data_lph", data_lph);
        main.putExtra("data_tnop", data_tnop);
        main.putExtra("data_tnod", data_tnod);
        main.putExtra("data_wa", data_wa);
        main.putExtra("data_it", data_it);
        main.putExtra("data_dia_lat", data_dia_lat);
        main.putExtra("data_one_side_sm", data_one_side_sm);
        main.putExtra("data_nod_lat", data_nod_lat);
        main.putExtra("data_dis_rate", data_dis_rate);
        main.putExtra("data_dis_hl_lat", data_dis_hl_lat);
        main.putExtra("data_dia_sm", data_dia_sm);
        main.putExtra("data_len_sm", data_len_sm);
        main.putExtra("data_nol_sm", data_nol_sm);
        main.putExtra("data_dis_rate_sm", data_dis_rate_sm);
        main.putExtra("data_hl_sm", data_hl_sm);
        main.putExtra("data_dia_m", data_dia_m);
        main.putExtra("data_len_m", data_len_m);
        main.putExtra("data_hl_m", data_hl_m);
        startActivity(main);
    }
}
}
}
}

```

### Selection of pump\_activity.xmt

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```

```

android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16sp"
tools:context=".Selection_of_pump">

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/total_hl"
    android:textAppearance="@style/TextAppearance.AppCompat.Body2"
    android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/suction"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Suction+Delivery (head)(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/op"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Operating pressure(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/fit_loss"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Fitting loss(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/filter_loss"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

        android:hint="Filter loss(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/v1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Venture loss(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/ed"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Elevation difference(m)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/btn_pump1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Total head" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/th_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" m" />
</LinearLayout>

```

```

<Button
    android:id="@+id/next_pump"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="270dp"
    android:text="Next"
    android:textSize="10dp" />

```

```

</LinearLayout>

```

### Selection of pump\_activity.java

```

package com.bawv.dirpirrigation;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class Selection_of_pump extends AppCompatActivity {
    Button pump1;
    TextView res1;
    EditText sd,op,fit_l,fil_l,vl,ed;
    double sd_,op_,fit_l_,fil_l_,vl_,ed_;
    double thl;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_selection_of_pump);

        Bundle pump = getIntent().getExtras();
        final String data_user = pump.getString("data_user");
        final String data_inst = pump.getString("data_inst");
        final String data_city = pump.getString("data_city");
        final int data_mob = pump.getInt("data_mob");
        final int data_ks_ra = pump.getInt("data_ks_ra");
        final double total_area = pump.getDouble("total_area");
        final double data_length = pump.getDouble("data_length");
        final double data_Width = pump.getDouble("data_Width");
        final String data_crop = pump.getString("data_crop");
        final double data_sl = pump.getDouble("data_sl");
        final double data_sd = pump.getDouble("data_sd");
        final double data_water = pump.getDouble("data_water");
        final double data_dpp = pump.getDouble("data_dpp");
        final double data_lph = pump.getDouble("data_lph");
        final double data_tnop = pump.getDouble("data_tnop");
        final double data_tnod = pump.getDouble("data_tnod");
        final double data_wa = pump.getDouble("data_wa");
        final double data_it = pump.getDouble("data_it");
        final double data_dia_lat = pump.getDouble("data_dia_lat");

```

```

final double data_one_side_sm = pump.getDouble("data_one_side_sm");
final double data_nod_lat = pump.getDouble("data_nod_lat");
final double data_dis_rate = pump.getDouble("data_dis_rate");
final double data_dis_hl_lat = pump.getDouble("data_dis_hl_lat");
final double data_dia_sm = pump.getDouble("data_dia_sm");
final double data_len_sm = pump.getDouble("data_len_sm");
final double data_nol_sm = pump.getDouble("data_nol_sm");
final double data_dis_rate_sm = pump.getDouble("data_dis_rate_sm");
final double data_hl_sm = pump.getDouble("data_hl_sm");
final double data_dia_m = pump.getDouble("data_dia_m");
final double data_len_m = pump.getDouble("data_len_m");
final double data_hl_m = pump.getDouble("data_hl_m");

sd =findViewById(R.id.suction);
op =findViewById(R.id.op);
fit_l =findViewById(R.id.fit_loss);
fil_l =findViewById(R.id.filter_loss);
vl =findViewById(R.id.vl);
ed =findViewById(R.id.ed);
pump1 =findViewById(R.id.btn_pump1);
res1 =findViewById(R.id.th_result);

pump1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(sd.getText().toString()) ||
            (TextUtils.isEmpty(op.getText().toString()) ||
            (TextUtils.isEmpty(fit_l.getText().toString()) ||
            (TextUtils.isEmpty(fil_l.getText().toString()) ||
            (TextUtils.isEmpty(vl.getText().toString()) ||
            (TextUtils.isEmpty(ed.getText().toString())
            )
        ) {
            Toast.makeText(Selection_of_pump.this, "Fields can't be Empty",
                Toast.LENGTH_LONG)
                .show();
        }
        else {
            sd_ = Double.parseDouble(sd.getText().toString());
            op_ = Double.parseDouble(op.getText().toString());
            fit_l_ = Double.parseDouble(fit_l.getText().toString());
            fil_l_ = Double.parseDouble(fil_l.getText().toString());
            vl_ = Double.parseDouble(vl.getText().toString());
            ed_ = Double.parseDouble(ed.getText().toString());
            // for Total headloss
            thl = sd_ + op_ + fit_l_ + fil_l_ + vl_ + ed_ + data_hl_sm + data_hl_m;
            res1.setText(String.valueOf(thl));
        }
    }
});

Button next_pump = findViewById(R.id.next_pump);
next_pump.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(sd.getText().toString()) ||
            (TextUtils.isEmpty(op.getText().toString()) ||
            (TextUtils.isEmpty(fit_l.getText().toString()) ||

```

```

        (TextUtils.isEmpty(fil_l.getText().toString())) ||
        (TextUtils.isEmpty(vl.getText().toString())) ||
        (TextUtils.isEmpty(ed.getText().toString()))
    )
    {
        Toast.makeText(Selection_of_pump.this, "Fields can't be Empty",
            Toast.LENGTH_LONG)
            .show();
    }
    else {
        double data_thl = Double.parseDouble(res1.getText().toString());
        Intent pump = new Intent(Selection_of_pump.this, Selection_of_filter.class);
        pump.putExtra("data_user", data_user);
        pump.putExtra("data_inst", data_inst);
        pump.putExtra("data_city", data_city);
        pump.putExtra("data_ks_ra", data_ks_ra);
        pump.putExtra("data_mob", data_mob);
        pump.putExtra("data_length", data_length);
        pump.putExtra("data_Width", data_Width);
        pump.putExtra("total_area", total_area);
        pump.putExtra("data_crop", data_crop);
        pump.putExtra("data_sl", data_sl);
        pump.putExtra("data_sd", data_sd);
        pump.putExtra("data_water", data_water);
        pump.putExtra("data_dpp", data_dpp);
        pump.putExtra("data_lph", data_lph);
        pump.putExtra("data_tnop", data_tnop);
        pump.putExtra("data_tnod", data_tnod);
        pump.putExtra("data_wa", data_wa);
        pump.putExtra("data_it", data_it);
        pump.putExtra("data_dia_lat", data_dia_lat);
        pump.putExtra("data_one_side_sm", data_one_side_sm);
        pump.putExtra("data_nod_lat", data_nod_lat);
        pump.putExtra("data_dis_rate", data_dis_rate);
        pump.putExtra("data_dis_hl_lat", data_dis_hl_lat);
        pump.putExtra("data_dia_sm", data_dia_sm);
        pump.putExtra("data_len_sm", data_len_sm);
        pump.putExtra("data_nol_sm", data_nol_sm);
        pump.putExtra("data_dis_rate_sm", data_dis_rate_sm);
        pump.putExtra("data_hl_sm", data_hl_sm);
        pump.putExtra("data_dia_m", data_dia_m);
        pump.putExtra("data_len_m", data_len_m);
        pump.putExtra("data_hl_m", data_hl_m);
        pump.putExtra("data_thl", data_thl);
        startActivity(pump);
    }
    });
}
}
}

```

#### Selection of filter\_activity.xmt

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```

```

android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16sp"
tools:context=".Selection_of_filter">

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Selection of Pump"
    android:textAppearance="@style/TextAppearance.AppCompat.Body2"
    android:textSize="18sp" />

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/eom"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Efficiency of motor(%)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/eop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Efficiency of pump(%)"
        android:inputType="numberDecimal"
        android:textSize="10dp" />
</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/fill1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Calculate"
    android:textSize="10sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Pump" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/hp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

        android:text="Result" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=" HP" />
</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/sel_of_fill" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/filter"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" mÅ²/hr" />
</LinearLay
<Button
    android:id="@+id/pdf"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="120dp"
    android:layout_marginTop="34dp"
    android:text="Create Pdf"
    android:textSize="10dp"

/>
</LinearLayout>

```

### Selection of filter\_activity.java

```

package com.bawv.dirp irrigation;

import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.pdf.PdfDocument;
import android.os.Bundle;
import android.os.Environment;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

```

```

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;

public class Selection_of_filter extends AppCompatActivity {
    EditText em,ep;
    TextView res1, res2;
    double soft,em_,ep_,pr;
    Button pdf;
    Bitmap bmp,bmps;
    int pwd = 1220;
    String name,inst,city,crop;
    int mob,ksra;
    float area,length,width;
    float sl,sd,water_req;
    float no_of_dip, tno_dip, no_of_plant, dis_dip,wa,it;
    float dia_lat, one_side_sm, nof_lat,dis_lat,hl_lat;
    float dia_sm,len_sm,no_lat_sm,dis_sm,hl_sm;
    float dia_m,len_m,hl_m,thl;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_selection_of_filter);
        Bundle fill = getIntent().getExtras();
        final String data_user = fill.getString("data_user");
        final String data_inst= fill.getString("data_inst");
        final String data_city = fill.getString("data_city");
        final int data_mob = fill.getInt("data_mob");
        final int data_ks_ra = fill.getInt("data_ks_ra");
        final double total_area = fill.getDouble("total_area");
        final double data_length = fill.getDouble("data_length");
        final double data_Width = fill.getDouble("data_Width");
        final String data_crop = fill.getString("data_crop");
        final double data_sl = fill.getDouble("data_sl");
        final double data_sd = fill.getDouble("data_sd");
        final double data_water = fill.getDouble("data_water");
        final double data_dpp = fill.getDouble("data_dpp");
        final double data_lph = fill.getDouble("data_lph");
        final double data_tnop = fill.getDouble("data_tnop");
        final double data_tnod= fill.getDouble("data_tnod");
        final double data_wa = fill.getDouble("data_wa");
        final double data_it = fill.getDouble("data_it");
        final double data_dia_lat = fill.getDouble("data_dia_lat");
        final double data_one_side_sm = fill.getDouble("data_one_side_sm");
        final double data_nod_lat = fill.getDouble("data_nod_lat");
        final double data_dis_rate = fill.getDouble("data_dis_rate");
        final double data_dis_hl_lat = fill.getDouble("data_dis_hl_lat");
        final double data_dia_sm = fill.getDouble("data_dia_sm");
        final double data_len_sm = fill.getDouble("data_len_sm");
        final double data_nol_sm = fill.getDouble("data_nol_sm");
        final double data_dis_rate_sm = fill.getDouble("data_dis_rate_sm");
        final double data_hl_sm = fill.getDouble("data_hl_sm");
        final double data_dia_m = fill.getDouble("data_dia_m");

```

```

final double data_len_m = fill.getDouble("data_len_m");
final double data_hl_m = fill.getDouble("data_hl_m");
final double data_thl = fill.getDouble("data_thl");

name = data_user;
inst = data_inst;
city = data_city;
mob = data_mob;
ksra = data_ks_ra;

area = (float) total_area;
length = (float) data_length;
width = (float) data_Width;
crop = data_crop;

sl = (float) data_sl;
sd = (float) data_sd;
water_req = (float) data_water;

no_of_dip = (float) data_dpp;
tno_dip = (float) data_tnod;
no_of_plant = (float) data_tnop;
dis_dip = (float) data_lph;
wa = (float) data_wa;
it = (float) data_it;

dia_lat = (float) data_dia_lat;
one_side_sm = (float) data_one_side_sm;
nof_lat = (float) data_nod_lat;
dis_lat = (float) data_dis_rate;
hl_lat = (float) data_dis_hl_lat;

dia_sm = (float) data_dia_sm;
len_sm = (float) data_len_sm ;
no_lat_sm = (float) data_nol_sm;
dis_sm = (float) data_dis_rate_sm;
hl_sm = (float) data_hl_sm;

dia_m = (float) data_dia_m;
len_m = (float) data_len_m;
hl_m = (float) data_hl_m;
thl = (float) data_thl;

em =findViewById(R.id.eom);
ep =findViewById(R.id.eop);
Button sof1 =findViewById(R.id.fill1);
res1 =findViewById(R.id.hp);
res2 =findViewById(R.id.filter);
pdf =findViewById(R.id.pdf);
bmp = BitmapFactory.decodeResource(getResources(),R.drawable.igkv);
bmps = Bitmap.createScaledBitmap(bmp,1220,200,false);

createPDF();

sof1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if ( (TextUtils.isEmpty(em.getText().toString())) ||
            (TextUtils.isEmpty(ep.getText().toString()))

```

```

    )
    {
        Toast.makeText(Selection_of_filter.this,"Fields can't be Empty",
            Toast.LENGTH_LONG)
            .show();

    }

    else
    {
        em_ = Double.parseDouble(em.getText().toString());
        ep_ = Double.parseDouble(ep.getText().toString());
        double hp1= data_thl * data_dis_rate_sm;
        double hp2 = 75 * em_ * ep_;

        // for Pump HP
        pr = hp1/ hp2;
        res1.setText(String.valueOf(pr));

        soft = data_dis_rate_sm * 3.6;
        res2.setText(String.valueOf(soft));

        if(soft < 5)
        {
            Toast.makeText(Selection_of_filter.this,"Use 1 inch (25mm) filter",
                Toast.LENGTH_LONG)
                .show();
        }
        else if(soft < 9)
        {
            Toast.makeText(Selection_of_filter.this,"Use 1.5 inch (40mm) filter",
                Toast.LENGTH_LONG)
                .show();
        }
        else if(soft < 25)
        {
            Toast.makeText(Selection_of_filter.this,"Use 2.0 inch (63mm) filter",
                Toast.LENGTH_LONG)
                .show();
        }
        else
        {
            Toast.makeText(Selection_of_filter.this,"Use 1.5 inch (40mm) filter",
                Toast.LENGTH_LONG)
                .show();
        }
    }
    }
    }
    }
    }

private void createPDF() {
    pdf.setOnClickListeners(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Toast.makeText(Selection_of_filter.this,"Saved to Internal Storage/DISD Report.pdf",
                Toast.LENGTH_LONG)
                .show();

            PdfDocument myPdfDocument = new PdfDocument();

```

```

Paint myPaint = new Paint();
Paint titlePaint = new Paint();

PdfDocument.PageInfo myPageInfo1 = new PdfDocument.PageInfo.Builder(1220,1650,1).create();
PdfDocument.Page myPage1 =myPdfDocument.startPage(myPageInfo1);

Canvas canvas = myPage1.getCanvas();

canvas.drawBitmap(bmps,0,0,myPaint);

titlePaint.setTextAlign(Paint.Align.CENTER);
titlePaint.setTextSize(50);
canvas.drawText("Drip Irrigation Designing System Report", pwd/2,270,titlePaint);

myPaint.setTextAlign(Paint.Align.LEFT);
myPaint.setTextSize(35);
myPaint.setColor(Color.BLACK);
canvas.drawText("Name of farmer/ scholar : "+name,20,350,myPaint);
canvas.drawText("Institute : "+inst,20,400,myPaint);
canvas.drawText("Village/City : "+city,20,450,myPaint);
canvas.drawText("Mobile No. : "+mob,20,500,myPaint);
canvas.drawText("Khasra No : "+ksra,20,550,myPaint);

canvas.drawLine(1,605,1220,605,myPaint);

myPaint.setTextAlign(Paint.Align.LEFT);
myPaint.setTextSize(30);
myPaint.setColor(Color.BLACK);
canvas.drawText("Name of Crop : "+crop,20,700,myPaint);

myPaint.setTextAlign(Paint.Align.LEFT);
myPaint.setTextSize(25);
myPaint.setColor(Color.BLACK);
canvas.drawText("Length of plot : "+length+" m",20,750,myPaint);
canvas.drawText("Width of plot : "+width+" m",20,800,myPaint);
canvas.drawText("Total area : "+area+" m2",20,850,myPaint);
canvas.drawText("Lateral spacing/ row to row spacing : "+sl+" m",20,900,myPaint);
canvas.drawText("Dripper spacing/plant to plant spacing : "+sd+" m",20,950,myPaint);
canvas.drawText("Water requirement : "+water_req+" (liter/day/plant)",20,1000,myPaint);
canvas.drawText("Dripper discharge : "+dis_dip+" lph",20,1050,myPaint);
canvas.drawText("Number of dripper per plant : "+no_of_dip,20,1100,myPaint);
canvas.drawText("Total number of dripper : "+tno_dip,20,1150,myPaint);
canvas.drawText("Total number of plant : "+no_of_plant,20,1200,myPaint);
canvas.drawText("Water application : "+wa+" lph",20,1250,myPaint);
canvas.drawText("Irrigation timing : "+it+" hr",20,1300,myPaint);
canvas.drawText("Diameter of lateral : "+dia_lat+" mm",20,1350,myPaint);
canvas.drawText("Length for one side of submain : "+one_side_sm+" m",20,1400,myPaint);
canvas.drawText("Number of Dipper on one lateral : "+nof_lat,20,1450,myPaint);

myPaint.setTextAlign(Paint.Align.LEFT);
myPaint.setTextSize(25);
myPaint.setColor(Color.BLACK);
canvas.drawText("Discharge rate of one latera : "+dis_lat+" lps",690,750,myPaint);
canvas.drawText("Head loss od lateral : "+hl_lat+" m",690,800,myPaint);
canvas.drawText("Diameter of submain : "+dia_sm+" mm",690,850,myPaint);
canvas.drawText("Length of submain : "+len_sm+" m",690,900,myPaint);
canvas.drawText("Number of lateral on submain : "+no_lat_sm,690,950,myPaint);

```

```

canvas.drawText("Discharge rate of submain : "+dis_sm+" lps",690,1000,myPaint);
canvas.drawText("Head loss of submain : "+hl_sm+" m",690,1050,myPaint);
canvas.drawText("Diameter of main line : "+dia_m+" mm",690,1100,myPaint);
canvas.drawText("Length of main : "+len_m+" m",690,1150,myPaint);
canvas.drawText("Head loss of main : "+hl_m+" m",690,1200,myPaint);
canvas.drawText("Total head loss : "+thl+" m",690,1250,myPaint);
canvas.drawText("Efficiency of motor : "+em_+" %",690,1300,myPaint);
canvas.drawText("Efficiency of pump : "+ep_+" %",690,1350,myPaint);
canvas.drawText("Pump : "+pr+" HP",690,1400,myPaint);
canvas.drawText("Filter capacity : "+soft+" m3/hr",690,1450,myPaint);

canvas.drawLine(1,1500,1220,1500,myPaint);

myPaint.setTextAlign(Paint.Align.RIGHT);
myPaint.setTextSize(15);
myPaint.setColor(Color.BLACK);
canvas.drawText("This report is generated by DIDS app developed by SVCAET." ,pwd-
20,1530,myPaint);

myPdfDocument.finishPage(myPage1);

File file = new File(Environment.getExternalStorageDirectory(),"DISD Report.pdf");

try {
    myPdfDocument.writeTo(new FileOutputStream(file));
} catch (IOException e) {
    e.printStackTrace();
}
myPdfDocument.close();
}
});
}
}

/**/

```

## Resume



**Name** : Priyanka Netam  
**Contact Information**  
**Contact No.** : 9039432285  
**Email I'd** : [priyankanetam46@gmail.com](mailto:priyankanetam46@gmail.com)  
**Present address** : Godavari Girls Hostel, Administration Block VC  
Campus, IGKV Raipur (C.G.) 492012  
**Permanent address** : H.N.239,Subhash Ward Bhati Para Sudhakar Nayak  
Gali, Kanker, Kanker (C.G.), 494334

**Personal Information**  
**Father's name** : Mr. Laxminath Netam  
**Mothers's name** : Mrs. Maaya Netam  
**Sex** : Female  
**Date of Birth** : 01.05.1996  
**Religion** : Hinduism  
**Cast** : ST

### Academic Qualification

| S. No. | Degree/Exam                    | Year | University/Institute |
|--------|--------------------------------|------|----------------------|
| 1.     | M .Tech (Soil and Water Engg.) | 2020 | IGKV, Raipur (C.G)   |
| 2.     | B .Tech (Agrial. Engg.)        | 2018 | IGKV, Raipur (C.G)   |
| 3.     | 12 <sup>th</sup>               | 2013 | CGBSE, Raipur (C.G.) |
| 4.     | 10 <sup>th</sup>               | 2011 | CGBSE, Raipur (C.G.) |

**Signature**