IRRIGATION SCHEDULING SOFTWARES AS A TOOL FOR WATER-USE OPTIMISATION IN AGRICULTURE

BY

ANIBILOWO, ABDULJABBAR OLAOLUWA

2003/14756EA

BEING A FINAL YEAR PROJECT SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENT FOR THE AWARD
OF BACHELOR OF ENGINEERING (B.ENG)
DEGREE IN AGRICULTURAL AND BIORESOURCE
ENGINEERING,
FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA

NOVEMBER 2008

DEDICATION

To my Dad, the most important engineer in my life; my Mum for all her support and faith; my brothers and sister I hold so dearly, thank you all for your support.

CERTIFICATION

This Project Entitled IRRIGATION SCHEDULING SOFTWARE AS A TOOL FOR WATER-USE OPTIMIZATION IN AGRICULTURE by Anibilowo Abdul-jabbar. meets the regulations governing the award of the degree of Bachelor of Engineering (B.ENG.) of the Federal University of Technology, Minna, and it is approved for the contribution to scientific knowledge and literary presentation.

……………………………………… 28th November, 2008

Dr. M. S. Usman — Date

Supervisor

……………………………………… 3/12/2008

Dr. (Mrs) Z. D. Osunde — Date

Head of Department.

……………………………………… 19-11-08

External Examiner — Date

ACKNOWLEDGEMENTS

Praise be to Almighty ALLAH, the most beneficent the most merciful, for making it possible to reach this Stage in my academic pursuits.

I want to immensely give gratitude to the efforts and contributions of my supervisor Dr. M.S. Usman, for assisting me indeed, in the completion of this study.

I also acknowledge my able H.O.D. Dr. Z. D. Osunde and all the lecturers and staff of the Department of Agricultural and Bioresources Engineering for their understanding and support, both morally and otherwise.

I acknowledge the support of my parents and siblings, for without them my dreams and aspiration, including the completion of my degree program, wouldn't have been possible.

I equally want to thank all my closest friends, especially Aliyu M. Aliyu and Abdul-Jalil F. Yusuf, Ebenezer Iledun, Stanley Kalu, Kayode, Musa Danjuma, Ugocukwu Nwanchukwu, Ojegbase Ozoluwa, Yakubu Yakubu, and Yakubu Ibrahim. for their immense contributions and support to the completion of the project.

Thank You all.

ABSTRACT

Irrigation projects worldwide are formulated to accomplish their intended purpose with full consideration of physical, economic, social, and environmental factors. This project aims to satisfy these criteria, and showing how computers and its software can be effectively used in solving water-use problems and needs. The problem being the conservation of our major natural resource (water), and the need being provision of a well planned Irrigation scheme, for the overall increase in the agricultural production. Irrigation software for micro-computers are not widely used in most irrigation projects that are carried out in this country, so this project aims to address how useful they are in achieving the best possible irrigation schedule taking into account all the needed parameters to increase the efficiency of an irrigation scheme, like the each plants crop water requirement and calculating the evapotranspiration rates at any particular period of time This is carried out designing a mathematical model based on the Penman-monteith method of calculating evapotranspiration rates. The results gotten is are irrigation intervals needed to achieve the optimum crop water requirements needed to achieve an overall greater agricultural yield.

TABLE OF CONTENTS

CHAPTER ONE

1. INTRODUCTION

CHAPTER TWO

2. LITERATURE REVIEW

CHAPTER THREE

3. MATERIAL AND METHODS

CHAPTER FOUR

CHAPTER FIVE:

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

1. INTRODUCTION

Irrigation can be explained as the use of the available input water in the most economical and efficient way in the agricultural system (Egharevba, 2002). Due to possible errors in the intensive calculation of Irrigation Efficiencies, Crop Water Requirement and estimation of Evapotranspiration (ET), the use of computers and its software as a reliable tool in Irrigation Scheduling is important.

The role of microcomputers has been on an increase during the past 10 years. Before then computers were used scarcely, they were mainly used in universities and research institutes in the developed countries.

1.1 General Trends

Computer use has increased in the world of Irrigation and Drainage and thus also within the International Institute for Land Reclamation and Improvement (ILRI). The main task of the Institute is collecting and Disseminating knowledge in the two areas of Irrigation and Drainage. Computers have been used at ILRI for various purposes, such as developing simulation calculation models, besides the usual administration and word processing usage. They have also been used as a training tool in irrigation and drainage courses, particularly in the annual International Course on Land Drainage (K.J. Lenselink,1993).

Although hardware developments in the personal computer area have been fast and relatively cheap, this has led to a wide spread in many modern institutions concerned with irrigation and drainage, but the software development in this area is still lagging behind.

In a number of disciplines surrounding irrigation, computer modeling efforts have been made from time to time.

Although efforts exist in certain sub-sectors of irrigation, I have nevertheless thought that a more general knowledge of microcomputer models is needed to be made available to the irrigation practitioner or agricultural engineer.

1.1.2 Computer Use In Irrigation

Irrigation entails much more than computers, and many engineers advocate the central position of people rather than technology. Even so, this does not mean that the technical and engineering aspects should be neglected or could not benefit from the application of computers. Even social services could benefit from an interface with some computer-Based tools and techniques. Let us consider in a broad overview, areas in which computers, and more specifically microcomputers and their software are or can be advantageously applied.

Computers are used in Irrigation in the following broad areas and phases of project development:

- in research and education (for comparable uses in hydraulics and hydrology);
- in the planning and design of irrigation systems, either by sophisticated individual farmers, by western extension agencies or by professional designers of consulting engineers;
- during the implementation and design of irrigation systems, either by sophisticated individual farmers, by western agencies or by professional designers of consulting engineers;
- during the implementation of major projects, e.g. leveling work and in supervision and management;
- in the manufacture of irrigation pumps like pumps, sprinklers, drippers, valves, plastic piped, etc. ;
- in the operation of fields system, mainly for pressurized irrigation systems like sprinkler and drip irrigation; such computerized operation may include reservoir operation, barrage operation, controlled pumping and fertigation in greenhouses. Systems can be fully automated, or may be based on simpler technology, with only computerized data processing;

- in monitoring of irrigation systems; this may be automated; a computerized data collection and analysis system may be set up for immediate or later adjustment; reporting could easily be fitted into the system.

The need for Irrigation in Semi-arid regions is heightened by marginal rainfall and high evaporation demands. Therefore the need of irrigation scheduling on a daily basis is very significant to achieve the desired soil water content needed to yield a good harvest, or turnover of money invested.

As already indicated this project aims to call the attention of agricultural engineers and irrigation practitioners to the need of irrigation programs and how it can be used to achieve the major goal of agricultural engineering to alleviate the problems of a farmer.

This project, and the software discussed in it, is thus meant for agricultural engineers working in the field of irrigation, for e.g.

(i) dealing with applied research,

(ii) general design engineers, and

(iii) irrigation consultants.

I suppose that such practitioners have a keen interest in the use of microcomputers as a tool in their profession, not only to facilitate such common jobs as writing scientific reports, but also for analyzing research data or historical records, planning irrigation schemes, solving design and operational problems.

1.2 OBJECTIVES

- This study aims at achieving and demonstrating the necessity of Irrigation scheduling Software as an effective tool to improve irrigation efficiency and thereby the overall output of agriculture in the region, and prevent leaching and erosion.

- This study also aims to help out farmers do not know how much water their crops need and how to schedule their irrigation demands, which results to wasteful use of the resource and the impacts on the economy and environment.showing that decision making on water management is optimised with the help of a dedicated computer irrigation software.

1.3 SCOPE

This report aims to demonstrate how a yet again, herculean tasks which need precision calculations which can be easily achieved with the aid of computers and their software. It is reported from an Agricultural engineer or Irrigation practitioner's point of view and how it is relevant in the aim of improved production in agriculture. It contains various reports and journals on how irrigation should be carried out and subsequently how this can be achieved perfectly with the use of Irrigation programs. As stated earlier this project is with the engineer point of view and aims to call the attention of agricultural engineers and irrigation practitioners to the need of irrigation programs and how it can be used to achieve the major goal of agricultural engineering;"the application of engineering principles to alleviate the problems of the farmer".(Chukwu,2006)

A practical demonstration will be reported showing its ease of use and practicability in day to day irrigation scheduling and design for the main purpose of Water-Use Optimisation in agriculture, using data gotten from ET stations located in the semi-arid regions of Nigeria where irrigation in farming is particularly important.

1.4 LIMITATIONS

The Limitations to the projects implementation would be the inability of most farmers to operate computers and the attitude and idea that computers and agriculture cannot be combined to increase yields. Another Limitation would be the unavailability of agro-meterological station websites where all data needed to compute the necessary calculations, can be gotten or downloaded directly into the software.

CHAPTER TWO

2. LITERATURE REVIEW

2.1 IRRIGATION

Irrigation schemes or processes can be explained as the;

- Distribution of water in the field after reserving or storing them in a storage tank, and then transporting them to a field or farmland, where it is needed;
- Use of input water in the most economical and efficient way in an agricultural system.

The use of the input water in the most economical and efficient way is a function of highest rate available or distributed, between the plants need and the practical portion of water given through the irrigation system by the operator or agricultural engineer. (Egharevba, 2002)

Irrigation systems are widely divided into:

a. Surface Irrigation systems (Open systems)
b. Sprinkler and Drip Irrigation Systems (Closed systems)

2.1.1 Surface Irrigation

Surface irrigation systems convey water from the source, streams, rivers or ponds, that have undergone earlier test for quality and availability to fields in lined or unlined canals.

Because the three controlled surface irrigation methods:

Basin Irrigation,

Border Irrigation, and

Furrows Irrigation methods.

2.1.2 Basin Irrigation

Basin irrigation distinguishes itself from the other methods by a zero gradient and an often longer ponding phase in relation to the water advance phase. It also shows an almost instantaneous recession: the entire field falls dry at about the same moment, due to the level topography. The basin irrigation also includes low longitudinal gradients and one speaks of level basins to make the distinction. In hydraulic terms, one tries to obtain a high water application efficiency (average depth required over average depth applied) and a uniform distribution over the field (comparing bottom end infiltrated depth to top end infiltrated depth). Small basins, with a short water advance phase and a long ponding phase, can have very high application efficiencies and uniformities, but may otherwise be uneconomical (required labour for levelling, obstructions by bunds, required field canal density, etc.).

2.1.3 Border Irrigation

Border-strip irrigation is characterized by the sloping land surface in the longitudinal direction and the zero gradient across the field. For the purpose of sideways spreading, one often sees a head ditch or a small section with a zero slope at the head of the field. Borders also have a relative great length/width ratio as compared to the traditional basins. Water control is more critical than with level basins, since infiltration takes place between water advance and recession, in the absence of a ponding phase. Thus, very careful levelling and inflow handling is required to avoid poor water distribution uniformity. Flow rate selection and l cut-off timing are more critical than in level basin irrigation. Design criteria include the width and the length of the border strip, the longitudinal slope, the stream size and the application time, although width and slope are to a certain extent dictated by existing conditions. Other factors like infiltration rate, application depth and surface roughness cannot normally be modified and are soil and crop dependent. In practice, border length, flow rate and application time are the variable design parameters available to ensure uniform application.

2.1.4 Furrow Irrigation Method

The method is especially suitable for row crops as it requires mechanized land preparation (levelling and ridging) anyhow. Crops are planted on the side(s) of the ridge to avoid salinity effects. Design criteria include furrow width, depth and shape and furrow length, slope of the furrow, inflow rate and application time.

Mechanization, crop density and soil texture (wetting) determine furrow spacing and furrow shape, so that, hydraulically, furrow length, inflow rate and application time remain the basic engineering design variables. Field application efficiency and distribution uniformity are important parameters for the hydraulic performance of a furrow irrigation system, and efforts to improve on these in-field parameters include the use of cut-back flows and the application of surge flow. Re-use of tailwater has also been introduced where water is scarce to improve water use efficiency on a farm, scheme or project basis.

This short description of the major controlled surface irrigation methods may suffice to demonstrate that surface irrigation has been and still is very much concerned with efficient and uniform water distribution over the field. There has thus been a strong emphasis on the two most important parameters in surface irrigation, i.e. infiltration of water into the soil and the advance of water flowing over the soil surface. Infiltration received substantial attention over a long period leading to may infiltration formulas (Green & Ampt, Kostiakov,1983).

2.1.5 Surface Irrigation Programs

Microcomputer programs dealing with surface irrigation date back less than 10 years, although work on mainframe computers started another decade or so earlier. They are based on a set of two governing equations:

- the continuity equation and
- the momentum equation (i.e. the Saint Venant equations).

According to the way in which these basic equations are applied, three types of models are usually distinguished (Bassett et al., 1981).

- without major simplifications: "full hydrodynamic models";

- disregarding the acceleration term in the equation of motion: "zero-inertia models";

- replacing the momentum equation by simpler assumptions “kinematic models”.

Full hydrodynamic models are accurate, but delicate and require considerable computer time; they can be standards against which simpler models can be tested.

The zero-inertia models, however, have received much more attention, especially since the publication by Strelkoff & Katopodes (1977). A direct result of this work was the border flow program BRDRFLW (Strelkoff, 1985). There are quite a number of publications dealing with the zero-inertia models, but e.g. Maheshwari et al. (1989) found in Australia that the zero-inertia form of BRDRFLW worked very well for analyzing field collected data.

Later programs like BASCAD for basin irrigation (Ebonstra & Jurriëns, 1988) applied the same zero-inertia approach but used other algorithms and numerical solution techniques (like the Newton-Raphson iteration). An update of BRDRFLW called SRFR was published recently (Strelkoff, 1990). It covers furrows, basins and borders and can be run in full hydrodynamic, in zero-inertia or in kinematic-wave (or normal depth or uniform flow) modes.

Two types of kinematic models have been distinguished (Walker & Skogerboe, 1987).The first kinematic approach assumes a unique relation between flow rate and flow depth. This leads to so-called kinematic wave models, or also uniform-depth models, because the relation between flow rate and flow depth is often uniform flow equation like Manning, Chezy or Darcy-Weisbach.

A second approach to replace the momentum equation assumes a constant average cross-sectional surface flow area over the length of the field, and thus has, in fact, no real relation "kinematics". The latter is also called "volume balance model" (see e.g. Walker and Skogerboe, 1987). An example of a small "volume-balance" based Fortran program for computing uniformity, efficiency and losses in surface irrigation (basin, border or furrow) is mentioned in standard irrigation textbooks and a similar one in Basic for furrow irrigation design in another textbook (Cuenca, 1989). The FAO program SURFACE (Walker, 1989) is another example of this approach. Finally, the observation by Walker (1989) is supported, stressing that mathematical treatment of surface irrigation is only one tool in arriving at a good lay-out; other factors like size and shape of individual land holdings, land consolidation programs, farmer preferences, and equipment limitation may have a greater weight. It is good to realize that the mathematical models are only applicable in part of the design process and that uniform water distribution (the core of all discussed models) is only one aspect.

2.2 WATER REQUIREMENTS AND SCHEDULING

2.2.1 Crop Water Requirement

Calculating irrigation requirements is a basic step in many technical irrigation activities, such as designing canal systems and structures, estimating pumping requirements, preparing irrigation distribution schedules, operating existing irrigation systems and evaluating water use efficiencies. As the major objective of irrigation activities is the optimum supply of water to agricultural crops, knowledge of the crop water requirements is essential.

Crop water requirements are difficult to measure directly and accurately, and hence estimation methods have been in use for a long time. Relationships between actual crop water use and easily measurable meteorological parameters have proved useful over time, such as pan evaporation, air temperature and, especially, sunshine and radiation. Many correlations were developed, the best including a radiation term (which provides the vaporization energy) and a humidity & wind term (which provides the vapour gradient and transport). Penman (1948) combined these two approaches in a - formula for the evaporation from an open water surface **Eo.** His formula has been extensively tested and modified. The modified Penman equation (Doorenbos & Pruitt, 1977) is widely accepted, although the latest CROPWAT version employs the Penman-Monteith approach, recommended by a 1990 FAO Expert Consultation in Rome. There may be other formulae and models in use in academic environments (e.g. among crop physiologists). However, for normal engineering practice, the most common way to calculate crop water requirements in irrigation is the procedure described by replacing Penman's open water by a specified grass cover, from standard agro are minor controversies over "constants" to be used in some relations, but reasonable estimates of the reference evapotranspiration are produced for normal conditions (Jensen et al., 1990).

They calculate a reference evapotranspiration ETo meteorological data, mainly: sunshine, temperature, humidity and wind speed. The link between crop water requirements ETc and this reference ETo is made through formulating crop coefficients;

$$kc = \frac{ETc}{ETo} \tag{2.1}$$

which vary mainly per crop and per crop development stage. Such crop coefficients and a possible division into practical crop development stages for many common crops have also been provided (Doorenbos & Pruitt 1977).

Even if all agro-meteorological data for the Penman formula are available, the calculation of Eo or ETo is time-consuming and hence ways have been sought early on to facilitate the computation. Tables have been prepared (e.g. McCulloch, 1965) and nomographs were made (e.g. Koopmans, 19699) but the advent of the computer has really made an impact on the use of the (modified) Penman formula. Early attempts like an Algol program by Chidley & Pike, (1970) were followed by many others. Some were for private or incidental use, some for in-house application (e.g. Schellekens et al., 1992), some for local use (e.g. Kalders, 1988), while a number of them were published and thus available for general use. Most of these programs include the use the two most influential in the developmental process of the Washington Irrigation Scheduling Expert (WISE) software were SCHED developed by Buchleiter et al. (1988) and Washington Irrigation Forecaster (WIF) developed by Best et al. (1986).

SCHED calculates a daily water balance to estimate the present soil moisture depletion and this depletion in conjunction with a future estimate of crop evapotranspiration is used to predict the earliest and latest dates to irrigate a particular field.

WIF uses a soil-moisture measurement to determine the present soil moisture depletion and this sensor-derived depletion is used with a future estimate of crop-water use to predict the earliest irrigation that will refill the soil profile to a predetermined level. Both SCHED and WIF irrigation scheduling methods have been used successfully by agricultural consultants (Salazar et al., 1996; Dockter, 1996).

One of the design principles used to develop WISE was to create a tool that producers could use without the aid of professional consultants. As such, it was assumed that producers would not be willing to enter all the book-keeping required by SCHED or perform the soil-moisture sensor calibration required for WIF. Therefore, WISE employs a short-term water balance that can be adjusted according to measured soil-moisture trends. Also, the graphical user interface is intuitive and will help the user input their field specific parameters such as crop type/timing, soil moisture and irrigation system specifications.

2.2.2 Irrigation Scheduling

Irrigation scheduling can be understood as the determination of the right time and amount of irrigation application for optimal crop production. It addresses the basic questions of when the next irrigation is due and how much water to apply (assuming that the "how" is known). Since water is applied to the crop via the soil, the process is theoretically quite complicated and involves factors such as initial soil moisture. conditions, rates of change in soil moisture (evaporation, evapotranspiration), root extraction patterns, moisture transport in the root zone, limits of soil moisture suction in relation to plant growth, relationships between suction and moisture content, infiltration, re-wetting and percolation. Each of these sub-arms has been studied widely, leading to a large knowledge base. Modelling and simulation have been introduced in many of these aras over the past 20 years. Sophisticated computer simulation for irrigation scheduling now includes evapo-transpiration models, soil moisture movement models, root and crop growth models, although most models can as yet be used for analysis and not for real time scheduling. More general information on computer-based scheduling can be found in recent publications of Hoffman et al. (1990), Stewart & Nielsen (1990), and Hanks & Ritchie (1991).

In the scheduling programs discussed below the process is rather simplified, however. Most of the programs contain three elements:

- Potential evapotranspiration, as the "drawing force" depleting the soil water;
- The soil moisture storage, as a percentage of the volume between field capacity and wilting point, depending on the soil type and crop rooting depth;
- The relation between soil water content and crop yield. If the soil water falls below a certain value, yield reductions may occur, depending on the crop type, crop stage and evaporative demand.

The programs then calculate the optimum irrigation intervals under potential evapotranspiration, and water depths applied. Programs also have possibilities to simulate the effect of sub-optimum intervals, by calculating reduced ET values and relating these to yield reductions. The result is the change of soil moisture content with time. In all programs the theory on this aspect has been taken from Doorenbos & Kassam (1979), who summarized the then available knowledge on crop yield response to water.

Still, the scheduling programs are mostly a theoretical exercise. They can be used for design of surface irrigation or to assess what-if questions. Their practical operational value for smallholders in tertiary units is limited because the basic elements as application depth and interval are usually largely determined by external factors. They can be useful, however, for students, lecturers, engineers and planners to "play" with relatively simple data on water requirements, yield response to water and soil moisture retention, and see the consequences of different combination. The programs CROPWAT and IRSIS fall in this category.

Another category of programs are geared to assist the individual (large) farmer who wants to use his own personal computer for a tailor-made advice on when to irrigate his crops and how much to apply, not only on the basis of a day-to-day operation, but probably also in advance, so that he can weigh alternative cropping plans (Heermann et al., 1974). In this respect, large center-pivot sprinkler installations for instance would be well-advised to make use of a computerised scheduling service.

Irrigation programs which are designed to ease the planning and scheduling operations of the agricultural engineer must possess the appropriate planning and operational decisions characteristics needed to achieve the desired efficiency of the system.

Characteristics of Irrigation Planning and Operational Decision are:

a) the identification of the specific project;
b) the time at which the project should be built;
c) the size and magnitude of the irrigation project;
d) the target output which should be set;
e) specification off the real-time control decisions.

Calculation of evapotranspiration is also essential for the estimation of crop water use or for studying the effect of drought stress on crop performance with simulation models. Several methods are available for calculation of evapotranspiration. This report describes three different methods : the Penman method (1948) and the approaches of Makkink (1957) and Priestley-Taylor (1972).

The Penman method is important for the general understanding of evapotranspiration from surfaces both in more advanced models (such as greenhouse models) and in more simple approaches. When considered over longer periods of time (>10 days), the Penman method calculates crop water loss with a reasonable degree of accuracy. It can be considered the best among the simple approaches.

All three methods have in common that they estimate evapotranspiration of , well-watered crops, however, they differ in their data requirements. Makkink and Priestley-Taylor require fewer meteorological observations because they are based on the observation that in many climates, the radiation-driven part of evapotranspiration is much more important than the part driven by vapour pressure deficit and wind speed. In the Priestley -Taylor equation, evapotranspiration is proportional to net radiation, while Makkink evapotranspiration is proportional to short-wave radiation. The Penman method requires daily values of radiation, temperature, vapour pressure and wind speed. The Makkink and Priestley-Taylor equations require only radiation and temperature.

The Priestley-Taylor equation is used world-wide, e.g. in the IBSNAT network, but regional calibration can be necessary since it is based on the assumption that a constant relation exists between the evaporative demand by radiation and by wind. The same holds for the Makkink equation which is calibrated for use during the growing season in The Netherlands. In The Netherlands, Makkink and Priestley -Taylor should be used only during the growing season. An important finding is that the Makkink and Priestley -Tailor methods are valid for a larger part of the year in areas closer to the equator. This more or less justifies the use of these simple methods in agro-ecological zonation studies in these areas. The Penman formula calculates evapotranspiration by assuming that the surface temperature is not very different from the air temperature. Under normal circumstances this is indeed the case, but under extreme conditions surface temperatures can differ much from air temperatures, resulting in unwanted errors. To avoid this situation, the Penman module, as described here, can iteratively search for the equilibrium surface temperature and give an improved estimate of surface water loss.

CHAPTER THREE

3. MATERIALS AND METHOD

The methods used in developing an Irrigation scheduling software comprises Java programming coding and debugging, using a Java compiler on a java development platform which makes it function on any operating system. The model used is the Cropwat mathematical model which is based on the calculation of the rate at which particular crops undergo Evapotranspiration, which is then compared to the cropwater requirements at each stage of it cultivation, dependent on the crop-coefficient.

3.1 Method

3.1.1 Mathematical Model Background

The combination equation of Penman can be derived from the energy balance equation for an extensive area of open water, wet soil or crop as given in equation (2.1).

$$Rn - G - \lambda E - H = 0 \qquad (3.1)$$

"*Rn*" equals the energy lost by heat storage in the crop, water or soil (*G*, heat storage in the crop is too small to be considered here), plus the energy lost through evaporation (λE = latent heat of evaporation of water, λ multiplied by rate of evaporation *E*), plus the energy lost or gained through convection of sensible heat by the air (*H*). The direction of the sensible heat flux (*H*) is dependent on the sign of the temperature difference between the air and the surface under study. If the surface temperature (*Ts*) is lower than the air temperature (T_2), additional energy is transferred to the surface (as sensible heat). If the surface is warmer than the surrounding air the direction of the energy flux is the other way. When we consider that over longer periods of time the net energy flux into the ground (*G*) is zero, equation (2.1) simplifies to:

$$R_n - \lambda E - H = 0 \qquad (3.2)$$

All terms of equation (2.2) are in some way dependent on the surface temperature (*Ts*) Given the environmental conditions such as radiation, air temperature, wind speed and vapour pressure, there is only one surface temperature for which equation (2.2) holds . The rate of evaporation then simply equals: ***Rn* - *H*.**

The latent heat flux basically is driven by the difference in vapour pressure between the surface and the environment. If the surface is considered wet, this is the difference between the saturated vapour pressure at the surface temperature minus the vapour pressure of the environment. The latent heat flux is equal to this vapour pressure difference multiplied by the conductance to transfer derived from wind speed and surface characteristics (Dalton, 1802), the so-called wind function **(f(*u*2))** and multiplied by the latent heat of vaporization of water, **λ:**

$$\lambda E = \lambda \mathrm{f}(U_2)[e_s(T_s) - e_2] \qquad (3.3)$$

The symbol $e_s(T_s)$ indicates the saturated vapour pressure at the temperature of the surface, e_2 is the vapour pressure measured at screen height (usually two meters above the surface). The relationship between temperature and saturated vapour pressure is not linear and can be approximated by several empirical formulas

Similar to equation (2.3) the flux of sensible heat is:

$$H = \gamma\lambda\ \mathrm{f}(U_2)(T_s - T_2) \qquad (3.4)$$

where **(T_s - T_2)** is the temperature difference between surface and air, and "**γ**" is the psychrometer constant.

The key question of this system of equations is to find the surface temperature at which equations (3.2), (3.3) and (3.4) are satisfied (the dependence of **R_n** on **T_s** is only slight and is ignored here).

Although methods are available that iteratively determine the surface temperature, it was Penman (1948) who was able to eliminate the surface temperature by approximating the equation for exchange of latent heat (3.3) by using a linear relationship between temperature and saturated vapour pressure.

In this way a straightforward solution for $\boldsymbol{\lambda E}$ can be obtained for (3.2). The linearization stems from the notion that under practical circumstances, the surface temperature is often close to the environment temperature so that the saturated vapour pressure at the surface temperature can be approximated by

:

$$e_s\,(T_s\,) \cong e_s\,(\mathrm{T}_2\,) + s(\mathrm{T}_s - \mathrm{T}_2\,) \qquad (3.5)$$

The quantity s is the slope of the saturated vapour pressure curve around T_2, the temperature at 2m above the surface. The formula obtained by Penman is known as the combination equation:

$$\mathrm{E}_{\mathrm{PM}} = \frac{1}{\lambda}\,\frac{sR_n+\gamma\lambda f(U_2)(e_s(T_2)-e_2)}{s+\gamma} \qquad (3.6)$$

A great advantage of this formula is that weather data have to be measured only at one height above the surface contrary to earlier methods that required additional measurements of the surface temperature.

3.1.2 Basic Theoretical Concepts

The program developed was done entirely using the FAO CropWat 4.3 as a guide using the Java Language and the FAO 56 manual (1998) - Crop Evapotranspiration Guidelines for computing crop water requirements.

The program as well as the version uses the FAO (1992) Penman-Monteith methods for estimating reference crop Evapotranspiration. These estimates are used in crop water requirements and irrigation scheduling calculations.

These calculations are done with the help of soil, climate and rainfall data obtained for a region. The program uses a flexible menu system and file handling, and extensive use of graphics. Graphs of the input data (climate, cropping pattern) and results (crop water requirements, soil moisture deficit) can be drawn and printed with ease. Complex cropping patterns can be designed with several and crops with staggered planting dates.

The program uses a Windows style “pull-down” menu system, and a “toolbar” at the top of the main screen which leads you to many of the most frequently accessed data entry and results screens. It is important to note that JCropWat uses the tool bar as the main path to access the program operations. In addition, the program logic does not have to follow a traditional rigid flow chart. You can choose any tool on the tool bar or click on any menu on the menu bar.

The formula below is the governing Penman-Monteith equation for the calculation of reference evapotranspiration (mm/day).

3.1.2 Basic Theoretical Concepts

The program developed was done entirely using the FAO CropWat 4.3 as a guide using the Java Language and the FAO 56 manual (1998) - Crop Evapotranspiration Guidelines for computing crop water requirements.

The program as well as the version uses the FAO (1992) Penman-Monteith methods for estimating reference crop Evapotranspiration. These estimates are used in crop water requirements and irrigation scheduling calculations.

These calculations are done with the help of soil, climate and rainfall data obtained for a region. The program uses a flexible menu system and file handling, and extensive use of graphics. Graphs of the input data (climate, cropping pattern) and results (crop water requirements, soil moisture deficit) can be drawn and printed with ease. Complex cropping patterns can be designed with several and crops with staggered planting dates.

The program uses a Windows style "pull-down" menu system, and a "toolbar" at the top of the main screen which leads you to many of the most frequently accessed data entry and results screens. It is important to note that JCropWat uses the tool bar as the main path to access the program operations. In addition, the program logic does not have to follow a traditional rigid flow chart. You can choose any tool on the tool bar or click on any menu on the menu bar.

The formula below is the governing Penman-Monteith equation for the calculation of reference evapotranspiration (mm/day).

$$ET_o = \frac{0.408\Delta(R_n - G) + \gamma \frac{900}{T+273} u_2 (e_s - e_a)}{\Delta + \gamma(1 + 0.34u_2)} \qquad 3.7$$

R_n- net radiation at the crop surface [MJ m-2 day-1],

G- Soil heat flux density [MJ m-2 day-1],

T- Mean daily air temperature at 2 m height [°C],

U_2- wind speed at 2 m height [m s-1],

e_s- Saturation vapour pressure [kPa],

e_a- actual vapour pressure [kPa],

e_s – ea- saturation vapour pressure deficit [kPa],

Δ - Slope vapour pressure curve [kPa °C-1],

γ- Psychometric constant [kPa °C-1], and 900 is the conversion factor for daily-basis calculation.

A logical sequence by which the individual components (themselves consisting of complex formulae) of the formula relate to each other is indicated below:

- e_s is obtained from e_o which is also calculated at maximum and minimum temperatures;
- e_a can be obtained using three methods; from dew temperature directly; from RHmax or from RHmax and RHmin combined;

i) Vapour pressure deficit is then obtained from e_s and e_a.

ii) Ra is obtained from the latitude of the location and the day of the year

iii) Rn is obtained from Rns and Rnl

iv) Psychometric constant is obtained from the atmospheric pressure

v) The average of the maximum and minimum temperature gives Tmean which is used to calculate the slope of vapour pressure (Δ).

On the other hand, at many places reliable wind speed data are rarely available. For such cases, an expression, which does not require wind speed data, is proposed. The simplified formula for the Penman equation proposed to estimate potential evaporation from open-water without recourse to wind data is

$$ET_o = 0.047R_nT + 9.5 - 2.4\left(\frac{R_n}{R_a}\right)^2 + 0.09(T + 20)(1 - \frac{RH}{100}) \quad 3.8$$

where Ra ($MJ/m^2/d$) is the extraterrestrial radiation

Once these values are obtained, ETo can then be calculated. For details on individual calculation formulae please refer to handbook.

3.2 MATERIALS

It will be of use to give a brief comment on the language used for developing the jabbarCropWat model. Java technology is both a programming language and a platform. It is a high-level language that can be characterized by all of the following buzzwords: simple, object oriented, distributable, multi-threaded, dynamic, portable, robust, secured, high performance and architectural neutral.

In the Java programming language, all source code is first written in plain text files ending with the java extension. Those source files are then compiled into **.class** files by the Java compiler (java). A class file does not contain code that is native to your processor; it instead contains byte codes-the machine language of the Java Virtual Machine. The Java launcher tool (java) then runs your application with an instance of the Java Virtual Machine.

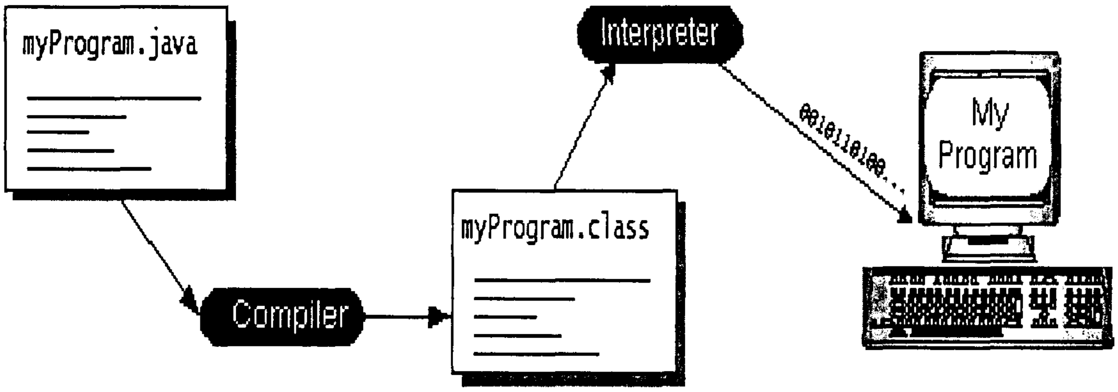


Figure 3.1: Computer Reading Java Program (Sun Developer Network, Products & Technology)

Java Virtual Machine is available on many different operating systems, the same class files are capable of running on Microsoft Windows, the Solaris Operating System (Solaris OS), Linux, or MacOS.

Some virtual machines, such as the java hot spot virtual machine perform additional steps at runtime to give your application a performance boost. This includes various tasks such as finding performance bottlenecks and recompiling (to native code) frequently-used sections of your code.

A platform is the hardware or software environment in which a program runs. Mention is already made of some of the most popular platforms like Microsoft Windows, Linux, Solaris OS, and MacOS. Most platforms can be described as a combination of the operating system and underlying hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms. The Java platform has two components:

- The Java Virtual Maschine
- The Java Application Programming Interface (API)

The Virtual Machine is the base for the Java platform and is ported onto various hardware-based platforms. The API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. It is grouped into libraries of related classes and interfaces; these libraries are known as packages. The following figure depicts how the API and the Java Virtual Machine insulate the program from the hardware.

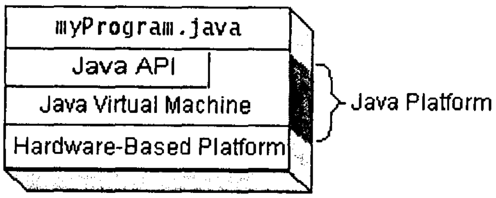
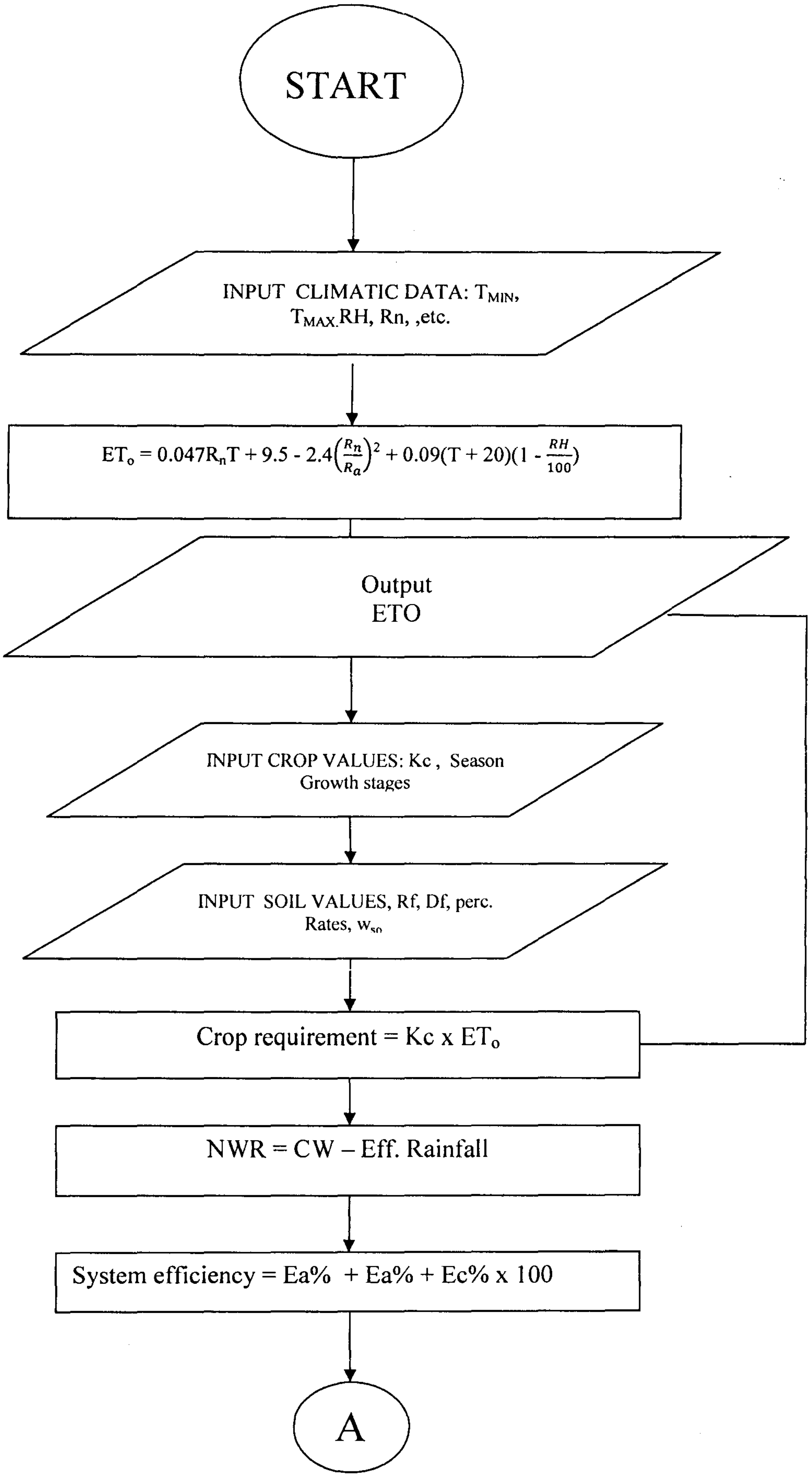


Figure 1: API, Java Virtual Machine And The Hardware Platform


START
INPUT CLIMATIC DATA: T_{MIN}, T_{MAX}.RH, Rn, ,etc.
$ET_o = 0.047R_nT + 9.5 - 2.4\left(\frac{R_n}{R_a}\right)^2 + 0.09(T + 20)(1 - \frac{RH}{100})$
Output ETO
INPUT CROP VALUES: Kc , Season Growth stages
INPUT SOIL VALUES, Rf, Df, perc. Rates, w_{so}
Crop requirement = Kc x ET_o
NWR = CW – Eff. Rainfall
System efficiency = Ea% + Ea% + Ec% x 100
A

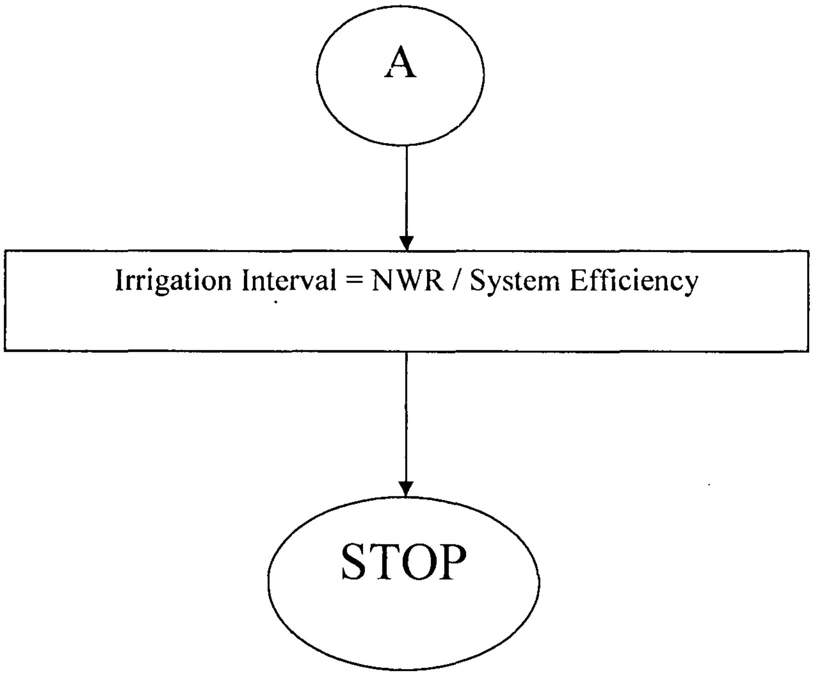


Figure 3.3: Software Workflow

CHAPTER FOUR

4. RESULTS AND CALCULATION

4.1 RESULTS

The results gotten show that the program runs effectively and the results gotten comply with the data already gotten from previous calculated irrigation schedules.

The screenshots showing the process from inputing these values to the result page is displayed below:

4.1.1 Climatic Data

MONTHLY ETO PENMAN-MONTEITH DATA

Altitude: 17 m. Latitude: 41.90 °N Longitude: 12.48 °E

Table 4.1: CLIMWAT climatic data

Month	Min Temp °C	Max Temp °C	Humidity %	Wind km/day	Sunshine hrs	Radiation MJ/m²/day	ETo m/day
January	4.5	11.1	77	181	3.4	6.0	0.88
February	5.4	12.6	75	164	3.4	7.9	1.17
March	7.2	15.2	70	181	4.7	11.9	1.90
April	9.8	18.8	69	156	5.9	16.2	2.66
May	13.3	23.4	66	147	7.2	19.8	3.63
June	17.2	27.6	61	130	7.8	21.4	4.41
July	19.6	30.4	56	130	9.7	23.6	5.16
August	19.4	29.8	58	130	8.9	20.9	4.63
September	16.9	26.3	67	121	7.2	15.9	3.25
October	12.8	21.5	73	147	5.3	10.6	2.03
November	9.3	16.1	77	156	3.2	6.3	1.18
December	6.4	12.6	78	164	2.6	4.9	0.86
Average	11.8	20.4	69	150	5.8	13.8	2.65

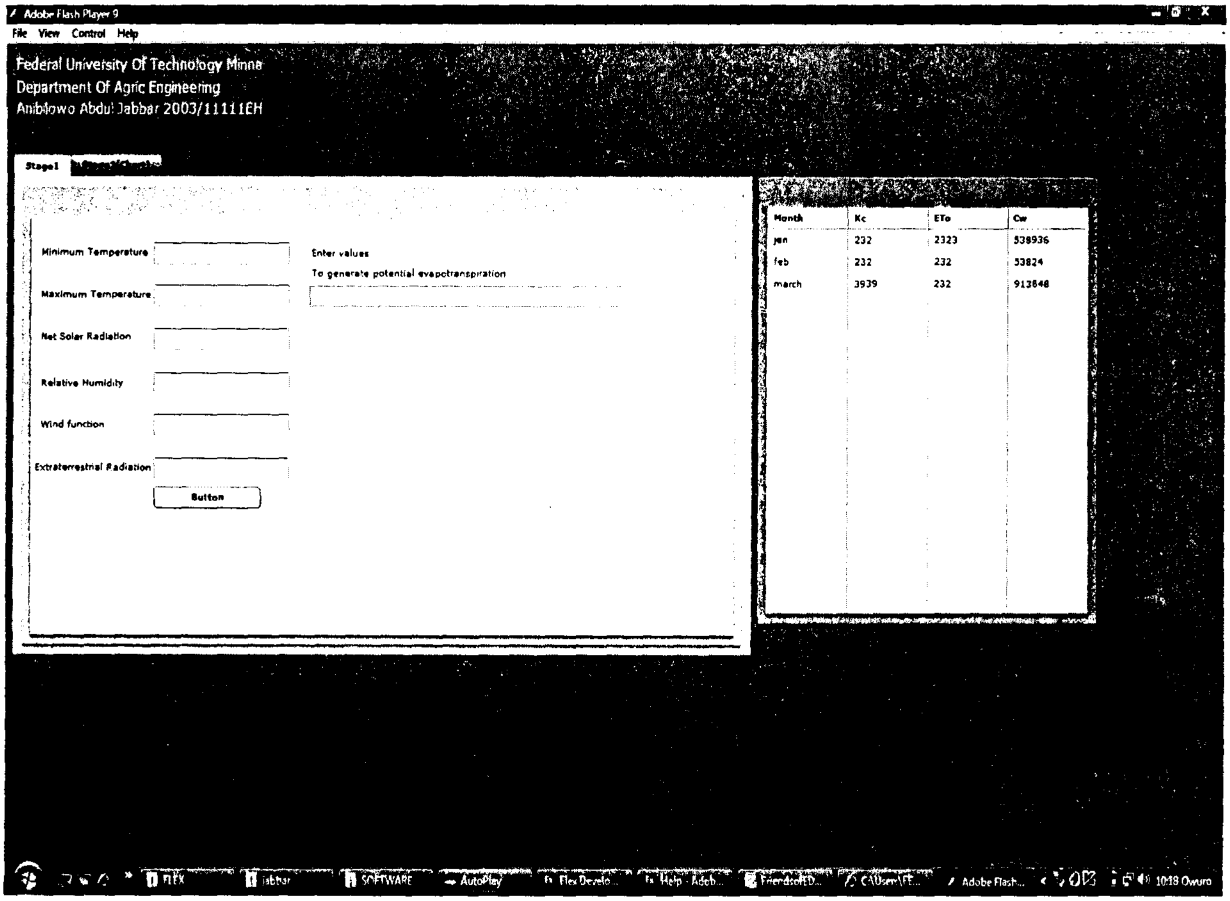
The data can be extracted for a single or multiple stations in the format suitable for their use in CROPWAT. Two tables are created for each selected station. As an example, these files are presented below in the case of Rome in Italy as displayed by CROPWAT. The table below contains long-term monthly rainfall data [mm/month]. Additionally, effective rainfall is also included calculated through the USDA Soil Conservation Service formula.

```
MONTHLY RAIN DATA

Eff. rain method: USDA Soil Conservation Service formula:

Peff = (Pdec * (125 - 0.6 * Pdec)) / 125 for Pmon <= 250 mm

Peff = 125 / 3 + 0.1 * Pdec for Pmon > 250 mm
```

Table 4.2: CLIMWAT Rainfall Data

Month	Rain mm	Eff. rain mm
January	81.0	70.5
February	68.0	60.6
March	71.0	62.9
April	64.0	57.4
May	56.0	51.0
June	38.0	35.7
July	16.0	15.6
August	25.0	24.0
September	65.0	58.2
October	124.0	99.4
November	112.0	91.9
December	98.0	82.6
Total	818.0	709.9

4.1.2 Program Screenshots

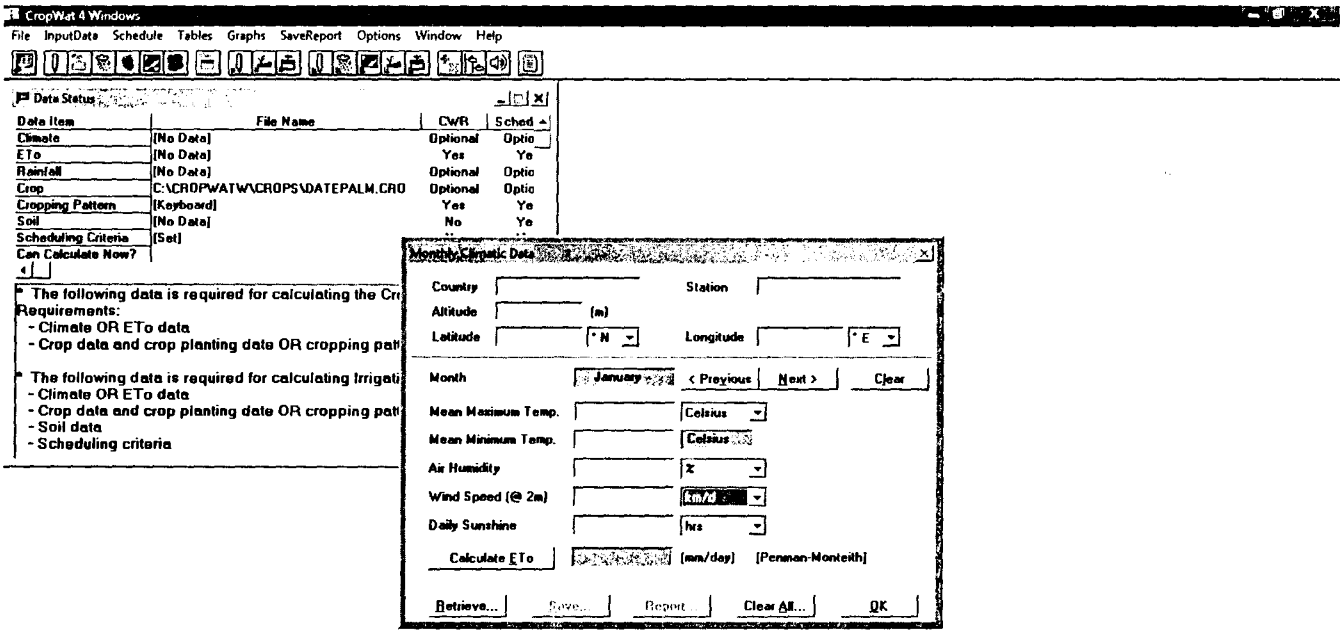


Federal University Of Technology Minna
Department Of Agric Engineering
Anibilowo Abdul Jabbar 2003/11111EH

Enter values
To generate potential evapotranspiration

Month	Kc	ETo	Cw
jan	232	2323	538936
feb	232	232	53824
march	3939	232	913848

Figure 4.1: Input Page



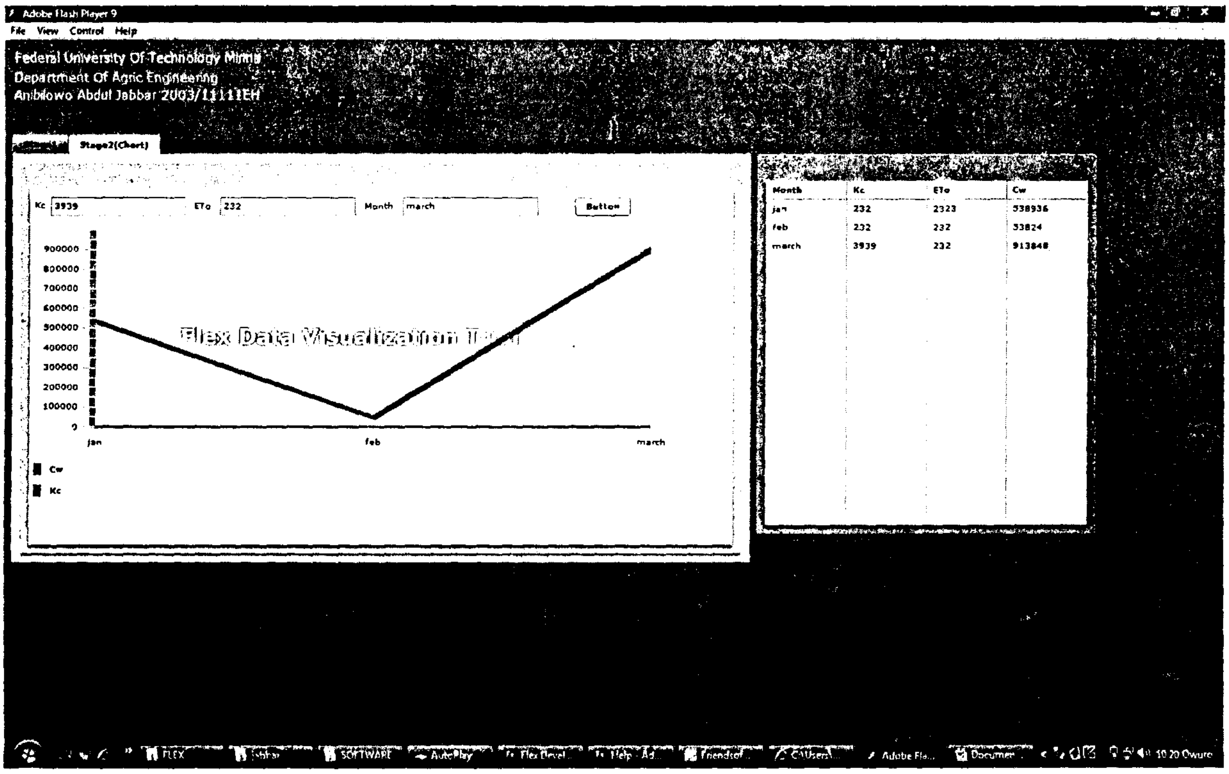
Data Item	File Name	CWR	Sched
Climate	[No Data]	Optional	Optio
ETo	[No Data]	Yes	Ye
Rainfall	[No Data]	Optional	Optio
Crop	C:\CROPWATW\CROPS\DATEPALM.CRO	Optional	Optio
Cropping Pattern	[Keyboard]	Yes	Ye
Soil	[No Data]	No	Ye
Scheduling Criteria	[Set]		
Can Calculate Now?			

- The following data is required for calculating the Cr
Requirements:
  - Climate OR ETo data
  - Crop data and crop planting date OR cropping patt

- The following data is required for calculating Irrigati
  - Climate OR ETo data
  - Crop data and crop planting date OR cropping patt
  - Soil data
  - Scheduling criteria

Country
Station
Altitude (m)
Latitude °N
Longitude °E
Month January
Mean Maximum Temp. Celsius
Mean Minimum Temp. Celsius
Air Humidity %
Wind Speed (@ 2m) km/d
Daily Sunshine hrs
(mm/day) [Penman-Monteith]

Figure 4.2: ET_o Calculation Screenshot

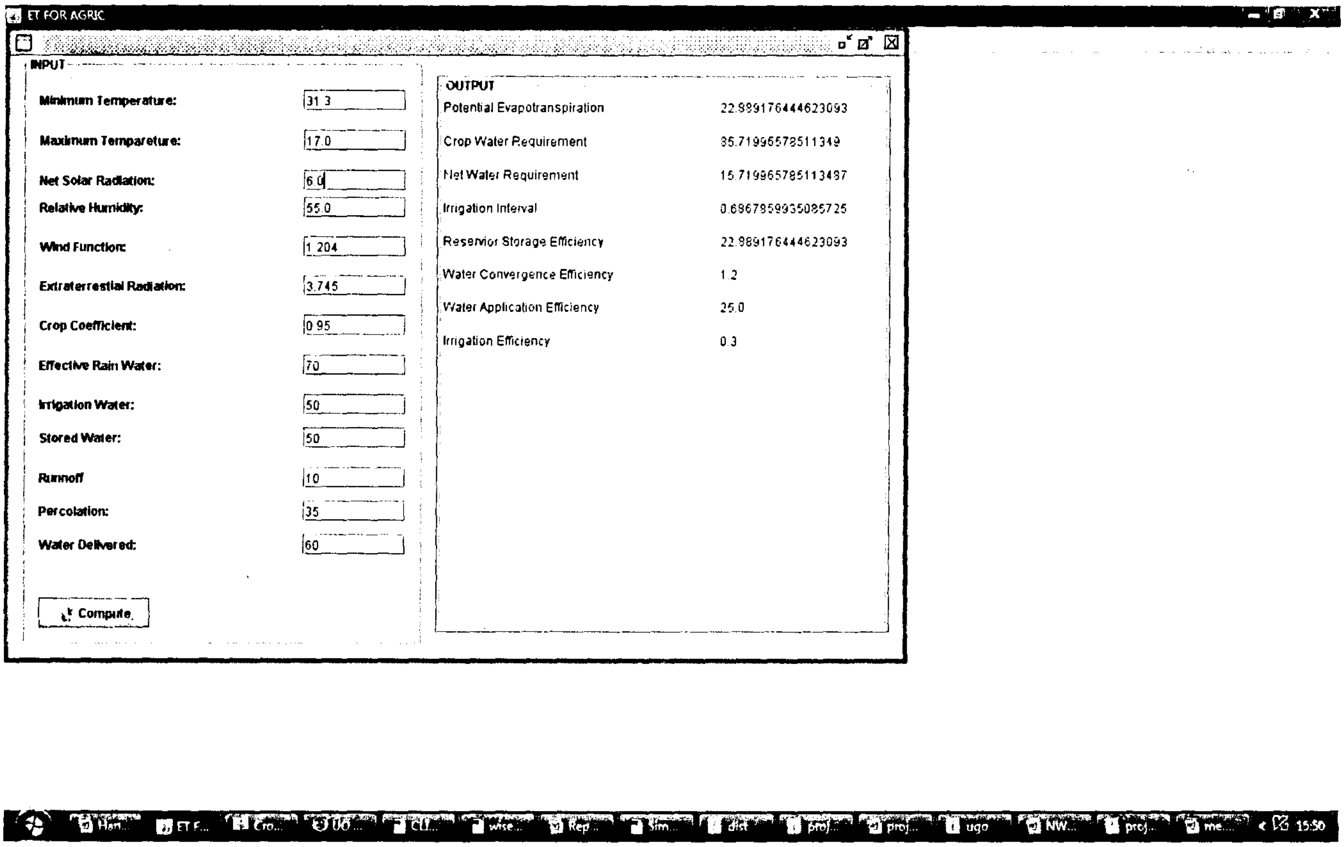


Federal University Of Technology Minna
Department Of Agric Engineering
Anibilowo Abdul Jabbar 2003/11111EH

Kc 3939
ETo 232
Month march

Month	Kc	ETo	Cw
jan	232	2323	538936
feb	232	232	53824
march	3939	232	913848

Figure 4.3: Graphical Representation Screenshot



INPUT

Minimum Temperature: 31.3

Maximum Tempareture: 17.0

Net Solar Radiation: 6.0

Relative Humidity: 55.0

Wind Function: 1.204

Extraterrestial Radiation: 3.745

Crop Coefficient: 0.95

Effective Rain Water: 70

Irrigation Water: 50

Stored Water: 50

Runnoff 10

Percolation: 35

Water Delivered: 60

OUTPUT

Potential Evapotranspiration	22.989176444623093
Crop Water Requirement	35.71996578511349
Net Water Requirement	15.719965785113487
Irrigation Interval	0.6867859935085725
Reservior Storage Efficiency	22.989176444623093
Water Convergence Efficiency	1.2
Water Application Efficiency	25.0
Irrigation Efficiency	0.3

Figure 4.4: Result Screen

CHAPTER FIVE

5. CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

The achievement of this project report requires quite a few resources (materials, money, etc) and time, including an electronic computer is needed for testing and running the programs, The final product of this semester project is an irrigation software written with java language. It estimates crop water requirement and schedules irrigation demand. Furthermore, it is a free access software which allows the development of recommendations for improved irrigation practices, the planning of irrigation schedules under varying water supply conditions, and the assessment of production under rain-fed conditions or deficit irrigation. With the planning and scheduling of irrigation achieved and carried in real-time, it thereby serving as a very viable tool for water-use optimisation. Another advantage is the opportunity for problems to be easily identified and solutions preferred before a particular design is implemented.

Irrigation scheduling softwares are mostly modelled using the Penman-monteith evapotranspiration equation, and scheduling based on the depletion of the soil moisture from the field capacity range. The software is not a true expert system in the sense that it doesn't decide explicitly when to start or stop irrigating, rather it allows the user be expert and decide when and how best to irrigate to achieve optimal harvest for a any particular crop in season. It uses a soil-moisture measurement to determine the present soil moisture depletion and this sensor-derived depletion is used with a future estimate of crop-water use to predict the earliest irrigation that will refill the soil profile to a predetermined level.

5.1.1 Constraints

The constraints faced in implementing or using the software as a tool, would be observing and acquiring accurate climatic data to run the software, used to predict or forecast the appropriate time to irrigate the field. Also the user has to be well versed with the process of irrigation, for he is the true expert not the software.

5.2 RECOMMENDATION

It is necessary to note that as it would be very helpful if agro-meteorological stations be connected to the world via the internet and share the monthly climatic values needed for the implementation of irrigation scheduling. Nonetheless, we feel that our work can be used as a basis for other students who are interested in software design for future works in the area of irrigation.

REFERENCE

Buchleiter, G. W., H.R. Duke, and D. F. Heermann. 1988. User's guide for USDA-ARS Irrigation scheduling program SCHED. USDA-Agricultural Research Service Agricultural Engineering Research Center, Colorado State University, Fort Collins, CO.

Dockter, D. T. 1996. AgriMet – The Pacific Northwest Cooperative Agricultural Weather Station Network. In C.R. Camp, E.J. Sadler and R.E. Yoder (Eds.) Evapotranspiration and Irrigation Scheduling. Proceedings of the International Conference. pp.75-80.

Doorenbos, J. and W. O. Pruitt. 1977. Guidelines for predicting crop water requirements., Food and Agriculture Organization ofthe United Nations. Irrigation and Drainage Paper 24

.

Egharevba N.A. 2002. Irrigation design and Drainage. Crop Water Requirement and Irrigation frequency. Nma Printers & Publiishing Co. Ltd. pp.1-17.

Lenselink, K.J. & M. Jurriëns, 1993. An inventory of irrigation software for microcomputers. ILFU Special report, Wageningen. p.172.

Lyndon K. 2000. Irrigation Scheduling Tools. Irrigation Fact Sheets #3, MSU Extension/Purdue Extension. 269/467-5511.

Rydzewski, J.R. & C.F. Ward, 1989. Irrigation and the microcomputer revolution Salazar, L,J., K.R. Thompson, and K. Crane 1996. Computerized irrigation scheduling in the San Luis Valley of Colorado. In C.R. Camp, E.J. Sadler, and R.E. Yoder (Eds.) Evapotranspiration and Irrigation Scheduling. Proceedings of the International Conference. pp.75-80

APPENDIX

Java Source Codes.

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package agric;

/**
 *
 * @author abduljabbar
 */
public class Jabba {

    //input
    private double t1;
    private double t2;
    private double Rs;
    private double rh;
    private double u;
    private double ra;
    private double er;
    private double ws;
    private double ws0;
    private double rf;
    private double df;
    private double wf;
    private double kc;

    /**
     * @return the t1
     */
    public double getT1() {
        return t1;
    }

    /**
     * @param t1 the t1 to set
     */
    public void setT1(double t1) {
        this.t1 = t1;
    }

    /**
     * @return the t2
     */
    public double getT2() {
        return t2;
    }

    /**
     * @param t2 the t2 to set
```

```
 */
public void setT2(double t2) {
    this.t2 = t2;
}

/**
 * @return the Rs
 */
public double getRs() {
    return Rs;
}

/**
 * @param Rs the Rs to set
 */
public void setRs(double Rs) {
    this.Rs = Rs;
}

/**
 * @return the rh
 */
public double getRh() {
    return rh;
}

/**
 * @param rh the rh to set
 */
public void setRh(double rh) {
    this.rh = rh;
}

/**
 * @return the u
 */
public double getU() {
    return u;
}

/**
 * @param u the u to set
 */
public void setU(double u) {
    this.u = u;
}

/**
 * @return the ra
 */
public double getRa() {
    return ra;
}

/**
 * @param ra the ra to set
 */
```

```
public void setRa(double ra) {
    this.ra = ra;
}

/**
 * @return the er
 */
public double getEr() {
    return er;
}

/**
 * @param er the er to set
 */
public void setEr(double er) {
    this.er = er;
}

/**
 * @return the ws
 */
public double getWs() {
    return ws;
}

/**
 * @param ws the ws to set
 */
public void setWs(double ws) {
    this.ws = ws;
}

/**
 * @return the ws0
 */
public double getWs0() {
    return ws0;
}

/**
 * @param ws0 the ws0 to set
 */
public void setWs0(double ws0) {
    this.ws0 = ws0;
}

/**
 * @return the rf
 */
public double getRf() {
    return rf;
}

/**
 * @param rf the rf to set
 */
public void setRf(double rf) {
```

```
        this.rf = rf;
    }

    /**
     * @return the df
     */
    public double getDf() {
        return df;
    }

    /**
     * @param df the df to set
     */
    public void setDf(double df) {
        this.df = df;
    }

    /**
     * @return the wf
     */
    public double getWf() {
        return wf;
    }

    /**
     * @param wf the wf to set
     */
    public void setWf(double wf) {
        this.wf = wf;
    }
    /**
     * @return the kc
     */
    public double getKc() {
        return kc;
    }

    /**
     * @param kc the kc to set
     */
    public void setKc(double kc) {
        this.kc = kc;
    }

    //output
    public double getETO(){
        double et0=(0.047*getRs()*((getT1()+getT2())/2))+((9.5-
2.4)*Math.pow(getRs()/getRa(),2))+((0.09*(1-
getRh())/100)*(((getT1()+getT2())/2)+20));
        return et0;
    }
    public double getCW(){
        return getETO()*getKc();
    }

    public double getNWR(){
        return getCW()-getEr();
```

```
    }
    public double getII(){
        return getNWR()/getETO();
    }
    public double getES(){
        return 100*getWs()/getWs0();
    }
    public double getEC(){
        return getWf()/getWs();
    }
    public double getEA(){
        return 100*(getWf()-(getDf()+getRf()))/getWf();
    }
    public double getIE(){
        return (getES()*getEC()*getEA()*100)/(100*100*100);
    }

    @Override
    public String toString() {
        StringBuffer s=new StringBuffer();
        s.append("Potential Evapotranspiration\t\t"+getETO()+"\n\n");
        s.append("Crop Water Requirement\t\t"+getCW()+"\n\n");
        s.append("Net Water Requirement\t\t"+getNWR()+"\n\n");
        s.append("Irrigation Interval\t\t"+getII()+"\n\n");
        s.append("Reservior Storage Efficiency\t\t"+getETO()+"\n\n");
        s.append("Water Convergence Efficiency\t\t"+getEC()+"\n\n");
        s.append("Water Application Efficiency\t\t"+getEA()+"\n\n");
        s.append("Irrigation Efficiency\t\t"+getIE()+"\n\n");
        return s.toString();
    }


}
```