Development of MATLAB App for pre-installation of Solar PV Plant

Report Submitted to the Central University of Karnataka, Kalaburagi
in partial fulfillment of the award of the Degree of

Bachelor of Technology

In
Electrical Engineering

By

Madhumita Sanjay Patil

(2019BEE17)

Under the Guidance of

Dr. Arunkumar Patil

Assistant Professor

Department of Electrical Engineering

School of Engineering

Central University of Karnataka

Kadaganchi, Kalaburagi, India 585 367.

2022-2023

Central University of Karnataka
Kadaganchi, Kalaburagi, India 585 367.
School of Engineering
Department of Electrical Engineering

Declaration By The Student

I MADHUMITA SANJAY PATIL(Registration no: 2019BEE17) hereby declare that the work reported in the B. Tech. thesis entitled **"Development of MATLAB App for pre-installation of Solar PV Plant "** submitted to the Department of Electrical Engineering, School of Engineering Central University of Karnataka, Kalaburagi, India. I also declare that this work has not been submitted for the award of any other degree/diploma of this University or any other institute. I further attest that this work is original and that I am fully responsible for the content of our B.Tech thesis.

Madhumita Sanjay Patil

Place:

Date:

Central University of Karnataka

Kadaganchi, Kalaburagi, India 585 367.

School of Engineering

Department of Electrical Engineering

CENTRAL UNIVERSITY OF KARNATAKA

CERTIFICATE

Certified that the project work entitled "**Development of MATLAB App for pre-installation of Solar PV Plant**" is a bonafide work carried out by **Madhumita Sanjay Patil (2019BEE17)** in partial fulfillmentof the requirement for the award of the degree of Bachelor of Technology in Electrical Engineering, School of Engineering, Central University of Karnataka, during the year 2022-2023. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Technology.

Signature of the Guide	Signature of the HOD	Signature of the Dean
Name of the Supervisor/Guide	Department of Electrical Engineering	School of Engineering

Madhumita Sanjay Patil 2019BEE17

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and appreciation to Dr. Arunkumar Patil, my project guide, for his invaluable guidance, expertise, and support throughout the project on "Development of MATLAB App for pre-installation of Solar PV Plant" at the Central University of Karnataka. His extensive knowledge, constant encouragement, and insightful suggestions have been instrumental in shaping this project.

Additionally, I am grateful to the Head of the Department Dr. Amarendra .M and Dean of School of Engineering Dr. Veeresh G. Kasabegoudar for their continuous support, encouragement, and helpful discussions. Their presence has been a source of inspiration and positivity throughout this project. I would like to acknowledge the support and cooperation of the faculty members at the Central University of Karnataka, whose expertise and guidance have been invaluable in enriching my understanding and enhancing the quality of this project.

I would also like to extend my heartfelt thanks to my family for their unwavering support and understanding. Their love, encouragement, and belief in my abilities have been a constant source of motivation and strength.

MADHUMITA SANJAY PATIL
(Reg. No: 2019BEE17)

ABSTRACT

The objective of this project is to develop a MATLAB-based program that can help with pre-installation solar calculations for any area. The software will estimate the earnings for various metering schemes. The economics of several solar panel types will also be compared, that includes monocrystalline, polycrystalline, and thin film. Users of this application will have access to a user-friendly platform where they may enter pertinent information about their location, rooftop size, and electricity usage. The optimal solar panel and metering system will then be selected for the user's requirements when the program completes the necessary calculations. This project aims to simplify the process of selecting the most profitable solar panels and metering systems and to promote the use of solar energy as a renewable energy source. The application is user-friendly and accessible, enabling homeowners to contribute to the transition to a more sustainable future.

TABLE OF CONTENT

LIST OF FIGURES

NOMENCLATURE

Term	Abbreviation
Irr :	Solar irradiance
Pf :	Performance Ratio
Eff :	The efficiency of solar panel
ADS :	Average daily direct Sunlight
Gen :	Generated Energy
Con :	Consumed Energy
Exp rate :	The export rate of electricity
Imp rate :	Import rate of electricity

Chapter-I

INTRODUCTION

1.1 Introduction

The transition to renewable energy sources has been gaining momentum in recent years, with solar energy being a popular choice for residential and commercial use. However, selecting the most profitable solar panel and metering system for a specific location can be a complicated and time-consuming process. The purpose of this project is to develop a MATLAB-based program that simplifies pre-installation solar calculations for any residential area.

The proposed program will estimate earnings for various metering schemes and compare the economics of different solar panel types, including monocrystalline, polycrystalline, and thin film. This will enable homeowners to make an informed decision about the best solar panel and metering system for their specific requirements. The user-friendly interface will allow users to input relevant information about their location, rooftop size, and electricity usage. Based on the provided data, the program will select the optimal solar panel and metering system for the user's requirements. This project aims to promote the use of solar energy as a sustainable energy source and to help homeowners contribute to a more eco-friendly future.

By providing an easy-to-use tool that simplifies pre-installation solar calculations, this project aims to encourage the adoption of solar energy and reduce carbon emissions. In addition to environmental benefits, using solar energy can also provide significant cost savings on electricity bills over time.

Overall, this project aims to bridge the gap between consumers' interest in adopting solar energy and the complex and time-consuming process of selecting the most profitable solar panel and metering system. By simplifying pre-installation solar calculations and providing users with a user-friendly interface, the program will help make solar energy more accessible and practical for residential use.

1.2 **Types of Solar Panels**

1. Monocrystalline Solar Panels: Monocrystalline panels are made from a single crystal structure, typically silicon. They are known for their high efficiency and sleek black appearance. Monocrystalline panels tend to be more expensive compared to other types, but they offer better performance in limited space and under low-light conditions.

2. Polycrystalline Solar Panels: Polycrystalline panels are made from multiple silicon crystals. They have a distinctive blue color and a lower efficiency compared to monocrystalline panels. However, they are generally more affordable and have good performance in high-temperature conditions.

3. Thin-Film Solar Panels: Thin-film panels are made by depositing thin layers of semiconductor material onto a substrate, such as glass or metal. They are less efficient compared to crystalline panels but have certain advantages. Thin-film panels are flexible, lightweight, and can be used in various applications. They also perform better in low-light conditions and high temperatures. However, they require more installation space and may degrade faster over time.

1.2 MNRE

Ministry of New and Renewable Energy (MNRE): MNRE is a government ministry in India that focuses on the development and deployment of renewable energy technologies and initiatives. It is responsible for formulating policies, programs, and regulations to promote renewable energy sources such as solar energy, wind energy, hydroelectric power, bioenergy, and more. MNRE plays a crucial role in facilitating the growth of renewable energy projects, providing financial assistance and incentives, and promoting research and development in the renewable energy sector.

Chapter II

LITERATURE SURVEY

2.1 Review of Literature Survey

The transition to renewable energy sources has gained significant momentum in recent years, with solar energy being a popular choice for residential and commercial use. In this context, several researchers have contributed to the development and implementation of solar photovoltaic systems.

Saurav Sokanlki and Saurabh Kumar, in their book Solar Photovoltaic Technology and Systems: A Manual for Technicians, Trainers, and Engineers [1], covers the technical aspects of solar photovoltaic technology and systems. The book includes comprehensive coverage of the fundamental concepts and principles of solar PV technology, including system components, PV module types, battery storage, charge controllers, inverters, and other power electronics. The book also provides practical guidance on system design, installation, and maintenance, as well as troubleshooting and repair techniques.

The government of India has issued tariff systems for gross metering and net metering of rooftop solar systems, which are regulated by the Ministry of New and Renewable Energy [2]. These systems determine the amount of money that a rooftop solar system owner is paid for the excess electricity they generate and feed back into the grid.

Prakasit Sritakaew, Anawach Sangswang, and Krissanapong Kirtikara [3] studied the reliability improvement of distribution systems using PV grid-connected systems. They investigated the effects of PV system installation on the distribution system's reliability by measuring the system's output characteristics. The authors found that the installation of PV grid-connected systems improved the system's reliability and reduced the system losses and loading level of the tie switch.

Souvik Ganguli & Sunanda Sinha [4] presented a paper about the Estimation of Grid Quality Solar Photovoltaic Power Generation Potential and its Cost Analysis in Some Districts of West Bengal. The objective of their work was to estimate the potential of grid-quality solar photovoltaic power in some districts of West Bengal, study the solar radiation level and potential of the above-mentioned districts, and finally, develop a system corresponding to the potential. Equipment specifications were provided based on the system developed and finally, cost analysis was also carried out.

A. Beccali, G. Ciulla, and M. Galatioto [5] presented a paper on the selection of the most appropriate photovoltaic system for a building. The authors proposed a methodology that considers the building's energy demand, the available space, and the economic viability of different photovoltaic system configurations.

S. Mekhilef, R. Saidur, and A. Safari [6] presented a review paper on the design and installation of photovoltaic systems. The authors discussed the different types of photovoltaic systems, the factors that affect the performance of a photovoltaic system, and the design and installation considerations for different types of buildings.

Moreover, National Portal for Rooftop Solar was the basis for the development of the MATLAB-based program. The National Portal for Rooftop Solar is a government initiative aimed at promoting the use of solar energy in India by providing a platform for consumers to connect with solar installation service providers. The portal also provides information on government policies and incentives for solar installations, as well as a solar calculator to estimate the potential savings and earnings from installing solar panels on rooftops.

2.2 Problem Statement

To develop a web-based tool that can provide an accurate estimate of the potential energy generation and cost savings of a rooftop solar photovoltaic (PV) system for residential, and commercial buildings, and land in India. The tool should consider various factors such as location, orientation, shading, and roof size to recommend the most appropriate PV system for the specific building. It should also provide information on the estimated system cost, financial incentives, and payback period to help landowners make informed decisions about investing in solar PV systems. The project aims to promote the adoption of rooftop solar PV systems in India, which can contribute to the country's renewable energy targets and reduce greenhouse gas emissions.

2.3 **Objective**

1. Develop a web-based tool that accurately estimates the potential energy generation and cost savings of rooftop solar photovoltaic (PV) systems for residential, and commercial buildings, and land in India.
2. Consider various factors such as location, orientation, shading, and roof size to recommend the most suitable PV system for specific buildings.

3. Provide comprehensive information on the estimated system cost, financial incentives, and payback period to assist landowners in making informed decisions about investing in solar PV systems.
4. Promote the adoption of rooftop solar PV systems in India to contribute to the country's renewable energy targets and reduce greenhouse gas emissions.

Chapter III

METHODOLOGY

3.1 Introduction

The proposed system appears to be a solar panel sizing and cost estimation tool. The system takes user inputs such as the city, rooftop area available(Square meter as well as acres), and monthly average unit consumption(KWh), and calculates various outputs related to the potential solar energy generation, cost, payback period, and return on investment.

3.2 Proposed Methodology

The methodology of the proposed system can be outlined as follows:

3.2.1 Understanding of Solar Photovoltaic Systems: The first step in developing a MATLAB solar app is to have a thorough understanding of solar photovoltaic systems. It is important to understand the underlying principles of solar energy, including the photoelectric effect, and how solar cells convert sunlight into electricity. A good understanding of solar cell technology and the different types of solar cells available is also necessary. In addition, knowledge of electrical engineering and circuit design is important for designing and implementing solar power systems. This knowledge will be crucial in developing a user-friendly app that accurately simulates the performance of solar photovoltaic systems.

Equations	
Solar Size = (Area*irr*pf*eff)/1000	3.1
Estimated cost = Solar size* Rate of pannel	3.2
Total energy generated = Solar Size* ADS*365	3.3
Netmetering Installation = Estimated cost + Netmetering cost	3.4
Annual Savings from net metering = (Gen- Con)*Exp rate + Con*Imp rate	3.5
Grossmetering Installation = Estimated cost + Grossmetering cost	3.6

Annual Savings from net metering = (Gen*Export rate) - (Con* Imp rate)	3.7
Return period = Installation cost/Annual saving	3.8

TABLE 3.1: Equations used in APP

3.2.2 Developing a User Interface: The user interface is an important component of the MATLAB solar app. It should be designed to be user-friendly and intuitive, allowing users to easily input data and obtain meaningful results. A well-designed interface will enhance the usability of the app and make it more accessible to a wider audience. The interface should be designed with consideration for different user levels, allowing both novice and experienced users to utilize the app to its full potential.

3.2.3 Implementing Solar Cell Models: Accurately modeling the behavior of solar cells is an important aspect of the MATLAB solar app. The app should be able to simulate the performance of different types of solar cells under different conditions, such as varying levels of sunlight and temperature. Models for solar cells should take into account factors such as the efficiency of the cell, the spectral response, and the effect of shading on the cell's performance.

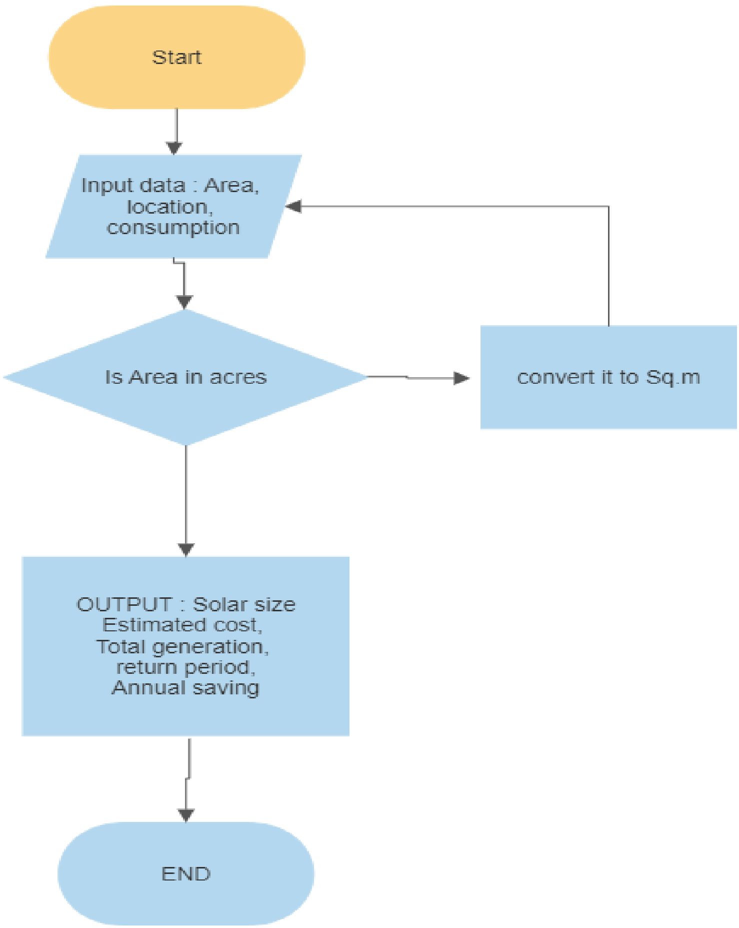

FIG 3.1: Flowchart of App

3.2.4 Developing a Simulation Engine: The MATLAB solar app should be able to simulate the performance of solar photovoltaic systems under different conditions. A simulation engine will be required to run simulations and generate accurate results. The simulation engine should take into account factors such as the efficiency of the solar cells, the angle of the sun, the weather conditions, and the configuration of the solar panel array. The engine should be designed to provide real-time feedback and generate graphical representations of the results.

3.2.5 Implementing Data Analytics: The MATLAB solar app should be able to analyze the performance of solar photovoltaic systems and provide insights into system efficiency and effectiveness. The app should be able to track key performance indicators such as energy output, system efficiency, and cost savings. It should also be able to generate graphical representations of the data, allowing users to easily visualize the performance of their system.

3.2.6 Integration with External Data Sources: The MATLAB solar app should be able to integrate with external data sources to provide users with real-time data on weather conditions, solar irradiance, and other relevant factors. This integration will allow the app to provide more accurate results and better insights into system performance.

3.2.7 Testing and Validation: The MATLAB solar app should be thoroughly tested and validated to ensure that it accurately simulates the performance of solar photovoltaic systems. The app should be tested under different conditions and validated against real-world data. This testing will ensure that the app is accurate, reliable, and user-friendly.

3.2.8 Documentation and User Guide: Proper documentation is an important aspect of any software project. The MATLAB solar app should be well-documented, with a user guide that provides detailed instructions on how to use the app. The documentation should be designed to be accessible to users with different levels of experience, from novice to expert.

3.2.9 Continuous Improvement: The development of the MATLAB solar app should not end with its release. Continuous improvement is important to ensure that the app remains up-to-date and relevant. User feedback should be taken into account and used to improve the app's functionality and usability over time. Regular updates should be released to address bugs, add new features, and improve the overall performance ofthe app.

Overall, the proposed system aims to provide an easy-to-use tool for homeowners to estimate the potential benefits and costs of installing a solar panel system on their rooftops or land.

Chapter IV

SOFTWARE DESIGN

The software design requirements for the proposed Matlab solar app are more complex and critical to the success of the project. The app will be developed using the Matlab software package, which is a high-level programming language that allows for data analysis, visualization, and simulation. The following are the key software design considerations for the project:

4.1 Data collection

Define the problem and project goals: The first step in this project is to define the problem you are trying to solve and set clear project goals. The problem here is to create a solar app that can accurately predict the amount of solar energy generated by a solar panel system. The goals of the project should include creating an accurate solar energy prediction algorithm and designing a user-friendly app interface.

Research and data collection: The next step is to research solar energy systems and collect data that will be used in the development of the solar energy prediction algorithm. This includes gathering data on the solar panel system's location, orientation, and specifications, as well as historical weather data for the area

4.2 Designing

User Interface Design: The user interface design is a crucial aspect of the project, as it determines the ease of use and effectiveness of the app. The user interface should be intuitive, user-friendly, and visually appealing, with easy access to all the features and functions of the app. The user interface should also allow for customization of the inputs, outputs, and display options, depending on the user's requirements.

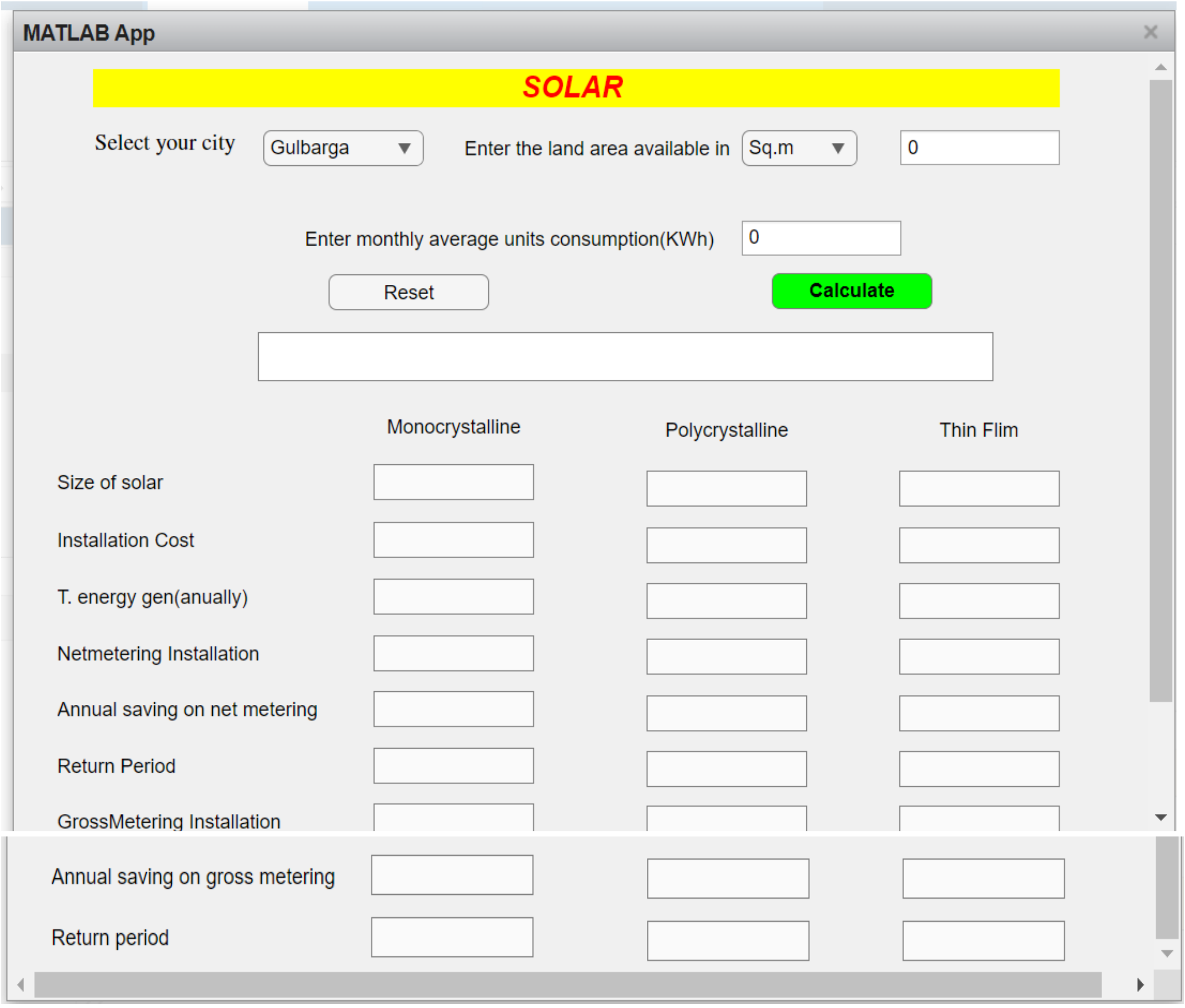

FIG 4.1: App Interface

Algorithm Design: The algorithm design is the backbone of the app, as it determines the accuracy and reliability of the simulations and predictions. The algorithm should be designed to handle various inputs, including weather data, location data, and panel specifications, and produce outputs such as power generation, efficiency, and cost-effectiveness. The algorithm should also be optimized for speed and accuracy, with the ability to handle large datasets and complex simulations.

4.3 Data and Error Management

Data Management: The data management system is critical to the success of the app, as it determines the ease of data input, storage, and retrieval. The data management system should allow for easy input of weather data, location data, and panel specifications, with the ability to store and retrieve this data efficiently. The system should also allow for data visualization, analysis, and export options, depending on the user's requirements.

Error Handling: The error handling system is critical to the success of the app, as it determines the reliability and robustness of the simulations and predictions. The error-handling system should be designed to handle various types of errors, including input errors, algorithm errors, and output errors. The system should also provide appropriate error messages and suggestions for corrective action.

4.4 Testing

The testing and validation system is critical to the success of the app, as it determines the accuracy and reliability of the simulations and predictions. The testing and validation system should be designed to test various scenarios, inputs, and outputs, with the ability to compare the results with actual data. The system should also allow for feedback and suggestions from users, with the ability to incorporate these suggestions into the app.

Chapter V

RESULT

The results of the Matlab solar app are estimates and calculations based on the user inputs and the predetermined formulas and data. These results provide valuable information related to the solar potential of a rooftop area and the potential benefits of installing a solar panel system. Here are some of the key results that the app provides:

- Solar System Size: The app calculates and displays the recommended solar system size for three different panel types: monocrystalline, polycrystalline, and thin film. These sizes are determined based on the entered rooftop area and other parameters.
- Installation Cost: The app estimates the installation cost for each panel type based on the calculated solar system size. It provides the cost in the local currency.
- Total Energy Generation: The app calculates the total energy generated annually by the solar panel system based on the solar system size and average solar irradiation. It displays this value in kilowatt-hours (KWh).
- Net Metering and Gross Metering: For users who input their monthly average unit consumption, the app calculates the potential savings and return on investment for net metering and gross metering scenarios. It provides information on the estimated savings and payback period.
- Return Period: The app calculates the return period for both net metering and gross metering scenarios. It indicates the number of years required to recover the installation cost based on the estimated savings.

These results provide users with valuable insights into the solar potential of their rooftop area and the economic viability of installing a solar panel system. Users can make informed decisions based on the estimated savings, return on investment, and payback period.

1) INPUTS:

LOCATION: Gulbarga

AREA: 3 Acres

Monthly Consumption: 700 KWh

OUTPUT:

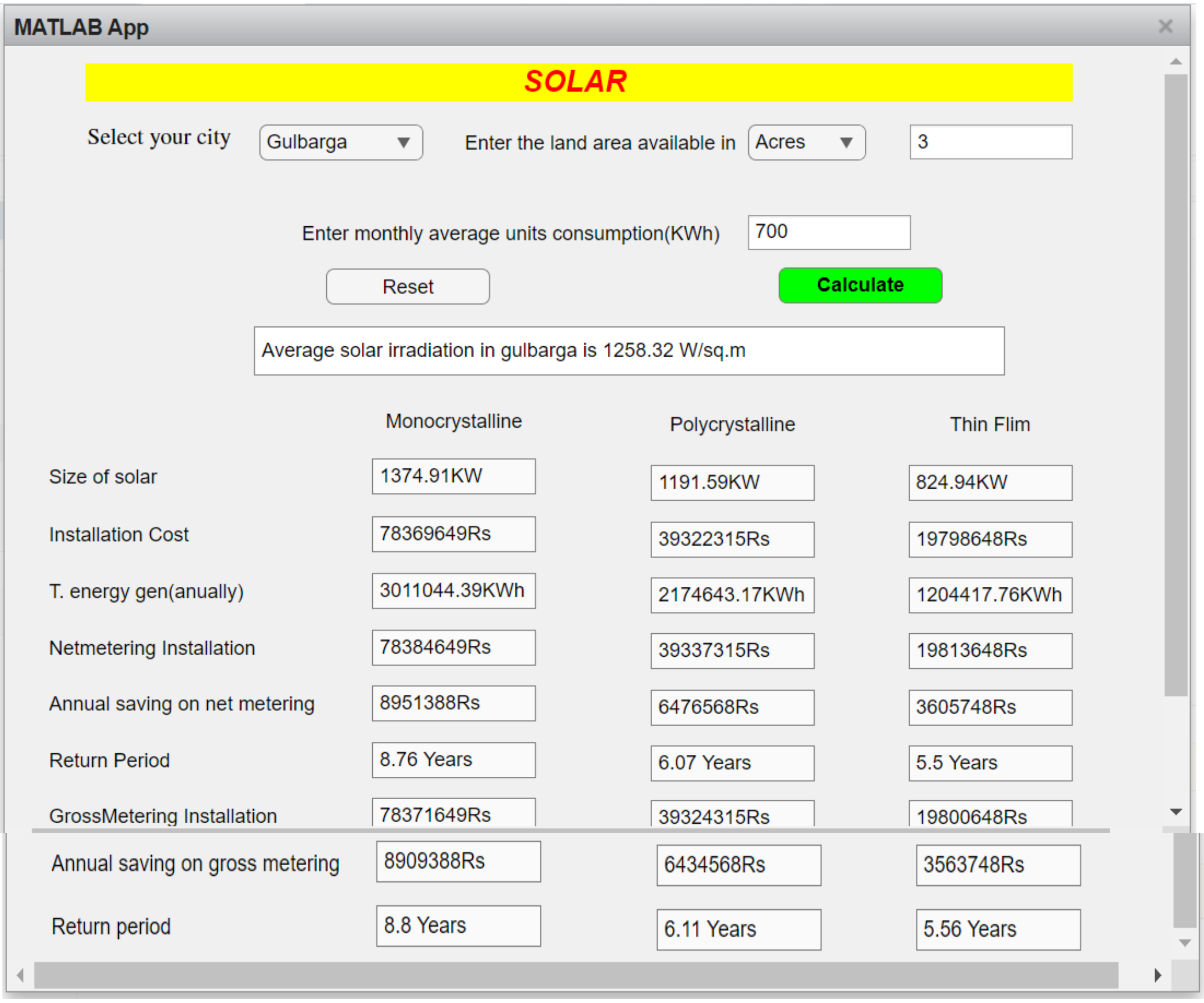

FIG 5.1:Result 1

2) INPUTS:

LOCATION: Mangalore

AREA: 100 Sq.m

Monthly Consumption:800 KWh

OUTPUT:

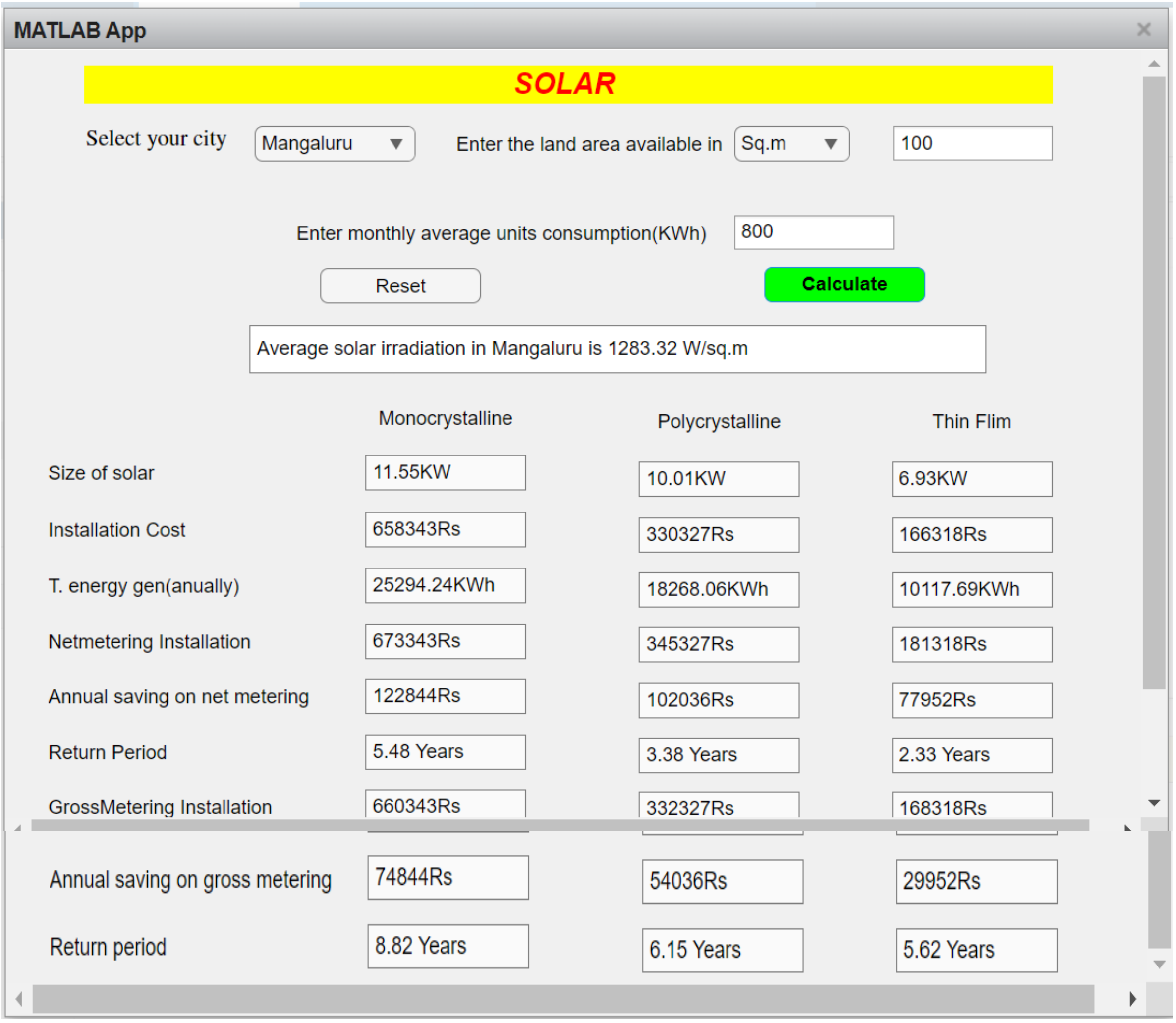

FIG 5.2:Result 2

3) INPUTS:

LOCATION: Bangalore

AREA: 40Sq.m

Monthly Consumption: 500 KWh

OUTPUT:

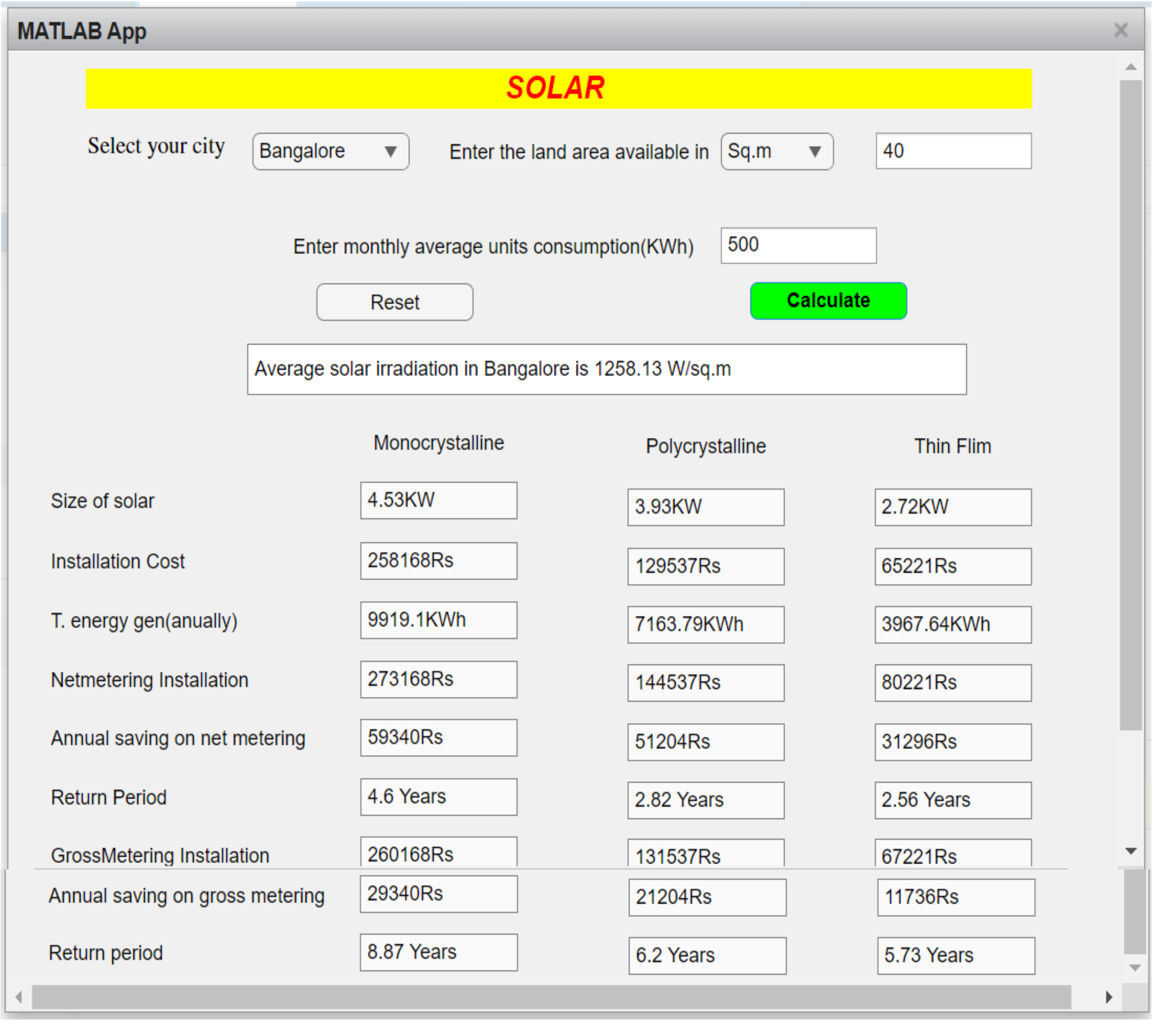

FIG 5.3:Result 3

4) INPUTS:

LOCATION: Belagavi

AREA: 5Acres

Monthly Consumption:300 KWh

OUTPUT:

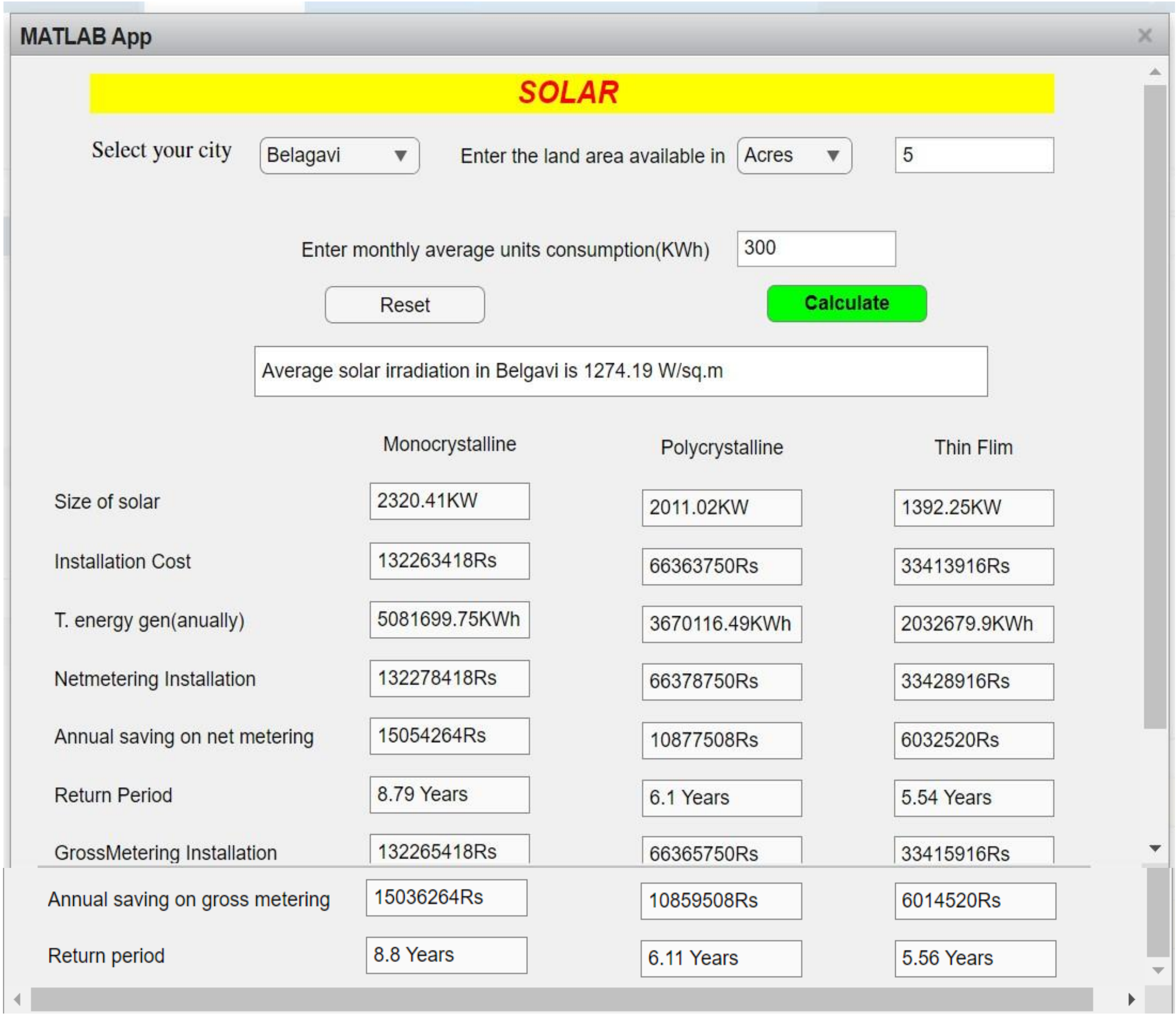

FIG 5.4:Result 4

5) INPUTS:

LOCATION: Belagavi

AREA: 5Acres

Monthly Consumption:300 KWh

OUTPUT:

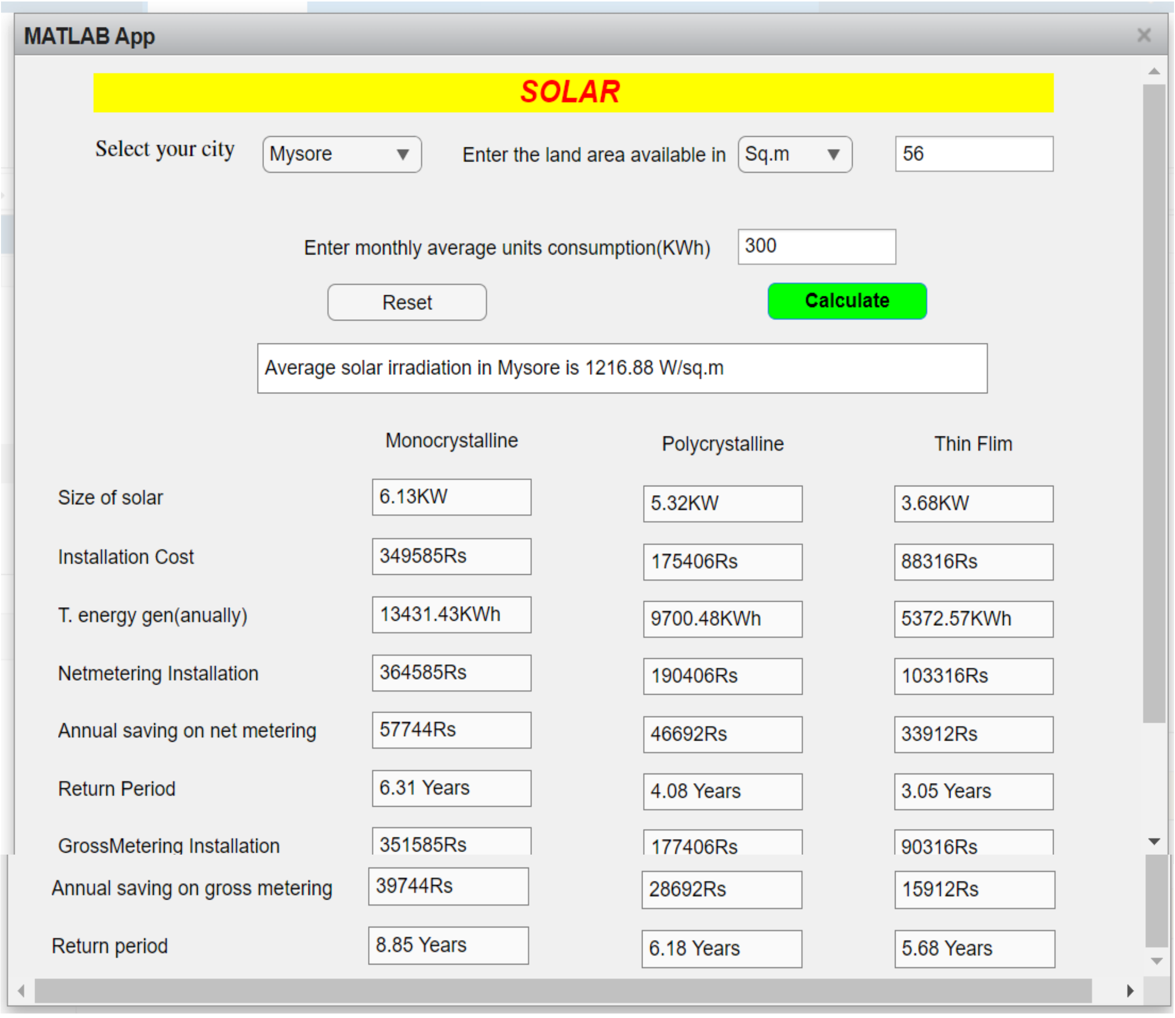

FIG 5.5:Result 5

Chapter VI

CONCLUSION

6.1 Conclusion

The Matlab solar app project was successfully developed to estimate the solar potential of a rooftop area using different parameters such as location, panel type, and efficiency. The app provides a user-friendly interface that allows users to input their rooftop area, location, and other relevant data to estimate the solar potential of their rooftop area. The app also provides valuable insights into the savings that could be achieved by installing a solar panel system, which can be useful for homeowners, businesses, and organizations.

6.2 Future Scope

There is a wide scope for future enhancements and improvements to the Matlab solar app project. Some potential areas of improvement are :

- Integration with real-time weather data to provide more accurate solar potential estimates.
- Incorporation of additional panel types and their corresponding efficiency values to provide more options to users.
- Integration with a cost calculator to estimate the return on investment for a solar panel installation.
- Integration with social media platforms to increase the reach of the app and enable users to share their solar potential estimates.
- Development of a mobile app version of the solar app to enable users to access the solar potential estimates on the go.

Overall, the future enhancements to the Matlab solar app project would make it more comprehensive and user-friendly, which would increase its adoption and usage.

REFERENCES

1. Sokanlki, Saurav, and Kumar, Saurabh. "Solar Photovoltaic Technology and Systems: A Manual for Technicians, Trainers, and Engineers." Publisher: TERI Press, 2016.
2. Ministry of New and Renewable Energy, Government of India. "Tariff Systems for Gross Metering and Net Metering of Rooftop Solar Systems." Available at: [2]
3. Sritakaew, Prakasit, Sangswang, Anawach, and Kirtikara, Krissanapong. "Reliability Improvement of Distribution Systems Using PV Grid-Connected Systems." IEEE Transactions on Energy Conversion, vol. 24, no. 2, June 2009.
4. Ganguli, Souvik, and Sinha, Sunanda. "Estimation of Grid Quality Solar Photovoltaic Power Generation Potential and its Cost Analysis in Some Districts of West Bengal." International Journal of Renewable Energy Research, vol. 3, no. 1,2013.
5. Beccali, A., Ciulla, G., and Galatioto, M. "Selection of the Most Appropriate Photovoltaic System for a Building: A Comprehensive Approach." Energy and Buildings, vol. 39, no. 8, August 2007.
6. Mekhilef, S., Saidur, R., and Safari, A. "A Review on Solar Energy Use in Industries." Renewable and Sustainable Energy Reviews, vol. 15, no. 4, May 2011.

APPENDIX - 1

CODE:

```
class def solar < Matlab.apps.AppBase
   % Properties that correspond to app components
   properties (Access = public)
      % Callbacks that handle component events
   methods (Access = private)
       % Button pushed function: ResetButton
       function ResetButtonPushed(app, event)
           app.EntermonthlyaverageunitsconsumptionKWhEditField.Value=0;
           app.EntertherooftopareaavailableinsqmEditField.Value=0;
       end
       % Button pushed function: CalculateButton
       function CalculateButtonPushed(app, event)
          option=app.SelectyourcityDropDown.Value;
          if strcmpi(option,'bangalore')
              roof=app.EnterthelandareaavailableinDropDown.Value;
              if strcmpi(roof,"acres")

app.EntertherooftopareaavailableinsqmEditField.Value=app.Entertherooftopareaa
vailableinsqmEditField.Value*4046.86;

              end
              irr=1258.13;

SSP=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.13*0.6)/1000;


SSM=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.15*0.6)/1000
;


SST=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.09*0.6)/1000;




               app.irrEditField.Value = "Average solar irradiation in
Bangalore is " + num2str(irr) + " W/sq.m";
               app.SSPEditField.Value= num2str(round(SSP,2))+ "KW";
               app.SSMEditField.Value= num2str(round(SSM,2))+ "KW";
               app.SSTEditField.Value= num2str(round(SST,2))+ "KW";
               app.ICPEditField.Value=num2str(round(33*SSP*1000))+ "Rs";
               app.ICMEditField.Value=num2str(round(57*SSM*1000))+ "Rs";
               app.ICTEditField.Value=num2str(round(24*SST*1000))+ "Rs";
               app.TEPEditField.Value=num2str(round(5*SSP*365,2))+ "KWh";
               app.TEMEditField.Value=num2str(round(5*SSM*365,2))+ "KWh";
               app.TETEditField.Value=num2str(round(5*SST*365,2))+ "KWh";

               app.NIPEditField.Value=num2str(round(33*SSP*1000+15000)) +
"Rs";
```

```
                app.NIMEditField.Value=num2str(round(57*SSM*1000+15000)) +
"Rs";
                app.NITEditField.Value=num2str(round(24*SST*1000+15000)) +
"Rs";

                a=app.EntermonthlyaverageunitsconsumptionKWhEditField.Value;
                B= a*8;
                genP=round(5*SSP*30);
                genM=round(6*SSM*30);
                genT=round(4*SST*30);
                if(genP>a)
                    cp=genP-a;
                    Dp=(cp*3);
                    NPP=(B+Dp)*12;
                    app.NPPEditField.Value=num2str((B+Dp)*12)+"Rs";
                else
                    cp=a-genP;
                    Dp=(cp*8);
                    NPP=(B-Dp)*12;
                    app.NPPEditField.Value=num2str((B-Dp)*12)+"Rs";
                end
                if(genM>a)
                    cm=genM-a;
                    Dm=(cm*3);
                    NPM=(B+Dm)*12;

                    app.NPMEditField.Value=num2str((B+Dm)*12)+"Rs";
                else
                    cm=a-genM;
                    Dm=(cm*8);
                    NPM=(B-Dm)*12;
                    app.NPMEditField.Value=num2str((B-Dm)*12)+"Rs";
                 end
                 if(genT>a)
                    ct=genT-a;
                    Dt=(ct*3);
                    NPT=(B+Dt)*12;
                    app.NPTEditField.Value=num2str((B+Dt)*12)+"Rs";
                else
                    ct=a-genT;
                    Dt=(ct*8);
                    NPT=(B-Dt)*12;
                    app.NPTEditField.Value=num2str((B-Dt)*12)+"Rs";
                 end
                app.GIPEditField.Value=num2str(round(33*SSP*1000+2000)) +
"Rs";
                app.GIMEditField.Value=num2str(round(57*SSM*1000+2000)) +
"Rs";
                app.GITEditField.Value=num2str(round(24*SST*1000+2000)) +
"Rs";
                app.GPPEditField.Value=num2str((genP*3)*12)+
                "Rs"; app.GPMEditField.Value=num2str((genM*3)*12)+
                "Rs"; app.GPTEditField.Value=num2str((genT*3)*12)+
                "Rs";
```

```
                app.RPNMEditField.Value=
num2str(round(((57*SSM*1000+15000)/NPM),2)) + " Years";

                app.RPNPEditField.Value=
num2str(round(((33*SSP*1000+15000)/NPP),2)) +" Years";

                app.RPNTEditField.Value=
num2str(round(((24*SST*1000+15000)/NPT),2)) +" Years";
                app.RPGMEditField.Value=
num2str(round((57*SSM*1000+2000)/(genM*36),2)) +" Years";

                app.RPGPEditField.Value=
num2str(round((33*SSP*1000+2000)/(genP*36),2)) +" Years";

                app.RPGTEditField.Value=
num2str(round((24*SST*1000+2000)/(genT*36),2)) +" Years";

          elseif strcmpi(option,'gulbarga')
               roof=app.EnterthelandareaavailableinDropDown.Value;
               if strcmpi(roof,"acres")

app.EntertherooftopareaavailableinsqmEditField.Value=app.Entertherooftopareaa
vailableinsqmEditField.Value*4046.86;

               end
                irr=1258.32;

SSP=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.13*0.6)/1000;


SSM=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.15*0.6)/1000
;


SST=(app.EntertherooftopareaavailableinsqmEditField.Value*irr*0.09*0.6)/1000;




                app.irrEditField.Value = "Average solar irradiation in Mysore
is " + num2str(irr) + " W/sq.m";
                app.SSPEditField.Value= num2str(round(SSP,2))+ "KW";
                app.SSMEditField.Value= num2str(round(SSM,2))+ "KW";
                app.SSTEditField.Value= num2str(round(SST,2))+ "KW";
                app.ICPEditField.Value=num2str(round(33*SSP*1000))+ "Rs";
                app.ICMEditField.Value=num2str(round(57*SSM*1000))+ "Rs";
                app.ICTEditField.Value=num2str(round(24*SST*1000))+ "Rs";
                app.TEPEditField.Value=num2str(round(5*SSP*365,2))+ "KWh";
                app.TEMEditField.Value=num2str(round(6*SSM*365,2))+ "KWh";
                app.TETEditField.Value=num2str(round(4*SST*365,2))+ "KWh";
```

```
            app.NIPEditField.Value=num2str(round(33*SSP*1000+15000)) +
"Rs";
            app.NIMEditField.Value=num2str(round(57*SSM*1000+15000)) +
"Rs";
            app.NITEditField.Value=num2str(round(24*SST*1000+15000)) +
"Rs";

            a=app.EntermonthlyaverageunitsconsumptionKWhEditField.Value;
            B= a*8;
            genP=round(5*SSP*30);
            genM=round(6*SSM*30);
            genT=round(4*SST*30);
            if(genP>a)
                cp=genP-a;
                Dp=(cp*3);
                NPP=(B+Dp)*12;
                app.NPPEditField.Value=num2str((B+Dp)*12)+"Rs";
            else
                cp=a-genP;
                Dp=(cp*8);
                NPP=(B-Dp)*12;
                app.NPPEditField.Value=num2str((B-Dp)*12)+"Rs";
            end
            if(genM>a)
                cm=genM-a;
                Dm=(cm*3);
                NPM=(B+Dm)*12;

                app.NPMEditField.Value=num2str((B+Dm)*12)+"Rs";
            else
                cm=a-genM;
                Dm=(cm*8);
                NPM=(B-Dm)*12;
                app.NPMEditField.Value=num2str((B-Dm)*12)+"Rs";
             end
             if(genT>a)
                ct=genT-a;
                Dt=(ct*3);
                NPT=(B+Dt)*12;
                app.NPTEditField.Value=num2str((B+Dt)*12)+"Rs";
            else
                ct=a-genT;
                Dt=(ct*8);
                NPT=(B-Dt)*12;
                app.NPTEditField.Value=num2str((B-Dt)*12)+"Rs";
             end
            app.GIPEditField.Value=num2str(round(33*SSP*1000+2000)) +
"Rs";
            app.GIMEditField.Value=num2str(round(57*SSM*1000+2000)) +
"Rs";
            app.GITEditField.Value=num2str(round(24*SST*1000+2000)) +
"Rs";
            app.GPPEditField.Value=num2str((genP*3)*12)+ "Rs";
```

```
                app.GPMEditField.Value=num2str((genM*3)*12)+ "Rs";
                app.GPTEditField.Value=num2str((genT*3)*12)+ "Rs";


                app.RPNMEditField.Value=
num2str(round(((57*SSM*1000+15000)/NPM),2)) + " Years";

                app.RPNPEditField.Value=
num2str(round(((33*SSP*1000+15000)/NPP),2)) +" Years";

                app.RPNTEditField.Value=
num2str(round(((24*SST*1000+15000)/NPT),2)) +" Years";
                app.RPGMEditField.Value=
num2str(round((57*SSM*1000+2000)/(genM*36),2)) +" Years";

                app.RPGPEditField.Value=
num2str(round((33*SSP*1000+2000)/(genP*36),2)) +" Years";

                app.RPGTEditField.Value=
num2str(round((24*SST*1000+2000)/(genT*36),2)) +" Years";
```