

# **ASTROSIGHT - HOROSCOPE PREDICTOR USING JAVA**

**A PROJECT REPORT FOR MINI PROJECT**

*Submitted by*

**ARYAN KHATRI (22BCA10138)**

**RUHI SHARMA (22BCA10159)**

in

**BACHELOR OF COMPUTER APPLICATIONS**

*under the supervision of*

**MR. RISHABH**



**University Institute of Computing,  
Chandigarh University, India.**

**April, 2025**

## **CERTIFICATE**

Certified that this project report “**ASTROSIGHT – HOROSCOPE PREDICTOR**” is the work of “**ARYAN KHATRI & RUHI SHARMA**” who carried out the project work under supervision of **Mr. RISHABH**.

Date:

**Mr. Rishabh**  
Assistant Professor  
University Institute of Computing,  
Chandigarh University, India.

# **ASTROSIGHT - HOROSCOPE PREDICTOR USING JAVA**

## **ABSTRACT**

AstroSight is a horoscope prediction tool developed using JAVA, designed to provide users with personalized astrological insights based on their zodiac signs. The system employs a robust algorithm to generate predictions.

AstroSight is currently build on command line , allowing users to input their names to receive accurate astrological forecasts.

By leveraging object-oriented programming in JAVA, AstroSight incorporates structured data handling, efficient algorithms, and modular code to ensure scalability and flexibility in future updates. The use of JAVA ensures the program runs with high performance and allows for seamless integration.

AstroSight aims to make astrology accessible and engaging for a wide audience, combining the mystical world of horoscopes with the precision of technology, offering users a personalized glimpse into their cosmic journey.

## **TABLE OF CONTENTS**

	<b>Page No.</b>
Abstract	<b>2</b>
List of Figures	<b>5-6</b>
1. Identification of Problem	
2. Identification of task	
3. Review Summary	
4. Existing Solutions	
5. Goals/Objectives	
6. Implementation of Solution	
7. Project Management	
8. Testing/Characterization/Interpretation/Data Validation	
9. Conclusion	
10.Code Snippets	

# 1. Identification of Problem :

Horoscopes have been a popular way to offer insights into a person's future based on their astrological sign. However, traditional methods of horoscope prediction often rely on manual calculations and interpretations, which can be time-consuming, subjective, and prone to errors. This inconsistency can lead to a lack of trust in the predictions provided.

In today's digital age, there is an increasing demand for automated and personalized horoscope predictions. Users want quick access to information that is tailored to their specific astrological profile, including factors such as their date of birth and astrological sign.

The core problem this project addresses is the inefficiency of traditional horoscope prediction methods. By utilizing JAVA, we aim to create an application that automates the horoscope prediction process. This application will leverage algorithms to compute astrological data and generate personalized predictions in real time. The goal is to provide users with instant, accurate, and engaging horoscope insights, removing the need for manual calculations and interpretations.

The ancient practice of astrology relies on the positions of celestial bodies to make predictions about individuals' futures based on their birth data. However, traditional horoscope prediction methods face several challenges:

- **Manual Interpretations:** Astrologers often rely on manual calculations and subjective interpretations, which can lead to inconsistencies and errors.
- **Time-Consuming:** Generating personalized horoscope predictions can be a lengthy process, causing delays for users seeking immediate insights.
- **Lack of Personalization:** Many existing solutions offer generalized predictions that fail to consider individual nuances, reducing user engagement and trust.
- **Accessibility Issues:** Individuals may not have access to qualified astrologers or the resources to interpret their charts accurately. These issues highlight the need for a reliable, automated solution that provides accurate and timely horoscope predictions, catering to users' growing demand for personalized content.

## 2. Identification of Task

The primary task of this project is to develop an automated horoscope prediction system using JAVA. This involves several key components:

- **Algorithm Development:** Create algorithms that accurately calculate astrological data based on user input, including birth date, time, and location.
- **User Interface Design:** Develop a user-friendly interface that allows users to easily input their data and receive predictions without requiring technical knowledge.
- **Output Generation:** Ensure that the application can generate detailed and personalized horoscope predictions that users can understand and relate to.
- **Testing and Validation:** Conduct thorough testing to validate the accuracy of predictions and the overall user experience.

## 3. Review Summary

A comprehensive review of the literature and existing solutions reveals several trends in the field of astrology:

- **Digitization of Astrology:** There has been a notable shift towards digital platforms that automate horoscope predictions, but many lack the depth and accuracy users seek.
- **User Engagement:** Successful horoscope applications focus on user experience, often incorporating elements of gamification or community features to engage users.
- **Technological Advances:** Recent advancements in data processing and algorithm development present opportunities to create more accurate and responsive astrological applications. This review highlights the importance of blending traditional astrological knowledge with modern technology to enhance user trust and satisfaction.

## 4. Existing Solutions

Currently available horoscope prediction solutions can be broadly categorized into:

- **Web-Based Applications:** These platforms offer basic horoscope predictions based on zodiac signs, but they often provide generic insights without personalization.
- **Mobile Applications:** Many apps have emerged in recent years, incorporating features such as daily horoscopes, compatibility analysis, and more. However, they may rely on limited algorithms that do not consider specific user data.
- **Personalized Services:** Some professional astrologers offer personalized services through websites or social media, but these services can be costly and are not always scalable. The existing solutions often fail to deliver the personalized experience users desire, which presents a significant opportunity for improvement with a dedicated JAVA application.

## 5. Goals/Objectives

The objectives of this project are as follows:

- **Automation of Predictions:** Develop a robust JAVA application that automates the horoscope prediction process, eliminating the need for manual calculations.
- **Accurate Algorithm Implementation:** Create algorithms that incorporate astrological principles to ensure reliable and personalized predictions.
- **User-Friendly Design:** Design an intuitive interface that simplifies user interaction and provides clear output without overwhelming users with technical jargon.
- **Thorough Testing and Validation:** Conduct extensive testing to validate the accuracy of predictions and user satisfaction, ensuring the application meets high standards of quality.
- **Continuous Improvement:** Establish a feedback loop with users to refine the application further and introduce new features based on user needs and preferences.

## 6. Implementation of Solution

In the analysis phase, we will:

- **Research Astrological Principles:** Study the fundamental aspects of astrology, including zodiac signs, planetary influences, and the calculation of birth charts.
- **Identify Key Algorithms:** Determine which algorithms will be used for calculating planetary positions and generating predictions, considering factors such as the current positions of celestial bodies and their aspects.
- **Define User Inputs:** Outline the necessary inputs required from users to generate accurate predictions, such as date, time, and place of birth.

## 7. Project Management and Communication

To ensure the project runs smoothly, we will implement a project management strategy that includes:

- **Task Breakdown:** Divide the project into manageable tasks with clear timelines and responsibilities assigned to team members.
- **Regular Meetings:** Hold weekly meetings to discuss progress, address challenges, and ensure alignment with project objectives.
- **Collaboration Tools:** Utilize collaboration tools such as GitHub for version control and project tracking, facilitating efficient communication and workflow management.

## 8. Testing/Characterization/Interpretation/Data Validation

The testing phase will involve several key activities:

- **Output Characterization:** Analyze the predictions generated by the application to assess their relevance and accuracy against established astrological principles.
- **User Testing:** Engage a group of users to interact with the application, providing feedback on usability, accuracy, and overall satisfaction. This feedback will be crucial for refining the application.



- **Data Validation:** Implement validation techniques to ensure that the algorithms consistently produce reliable results. This may include cross-referencing outputs with traditional astrological interpretations and conducting statistical analyses to measure accuracy.
- **Some Data :**

```
static {
    // Aries (Mesh)
    AstrologyInfo aries = new AstrologyInfo(
        "Aries (Mesh)",
        "Coral (Moonga)",
        "Tuesday (Mangalwar)",
        "Ashit Dhatu",
        "6-8 ct"
    );

    // Taurus (Vrish)
    AstrologyInfo taurus = new AstrologyInfo(
        "Taurus (vrish)",
        "Opel (Opel)",
        "Friday (Shukrawar)",
        "Asht Dhatu and silver",
        "8-11 ct"
    );

    // Gemini (Mithun)
    AstrologyInfo gemini = new AstrologyInfo(
        "Gemini (Mithun)",
        "Emerald (Panna)",
        "Wednesday (Bhudhwar)",
        "Asht Dhatu",
        "8-11 ct"
    );

    // Cancer (Kark)
    AstrologyInfo cancer = new AstrologyInfo(
        "Cancer (Kark)",
        "Pearl (Moti)",
        "Monday (Somwar)",
        "Asht Dhatu and Silver",
        "8-11 ct"
    );

    // Leo (Singh)
    AstrologyInfo leo = new AstrologyInfo(
        "Leo (Singh)",
        "Ruby (Manik)",
        "Sunday (Raviwar)",
        "Asht Dhatu",
        "8-11 ct"
    );

    // Virgo (Kanya)
    AstrologyInfo virgo = new AstrologyInfo(
        "Virgo (Kanya)",
        "Emerald (Panna)",
        "Wednesday (Bhudhwar)",
        "Asht Dhatu",
        "8-11 ct"
    );
};
```

```
// Libra (Tula)
AstrologyInfo libra = new AstrologyInfo(
    "Libra (Tula)",
    "Opel (Opel)",
    "Friday (Shukrawar)",
    "Asht Dhatu and Silver",
    "8-11 ct"
);

// Scorpio (Vrishchik)
AstrologyInfo scorpio = new AstrologyInfo(
    "Scorpio (Vrishchik)",
    "Coral (Moonga)",
    "Tuesday (Mangalwar)",
    "Asht Dhatu",
    "6-8 ct"
);

// Sagittarius (Dhanu)
AstrologyInfo sagittarius = new
AstrologyInfo(
    "Sagittarius (Dhanu)",
    "Yellow Sapphire (Pukhraj)",
    "Thursday (Guruwar)",
    "Asht Dhatu and Gold",
    "6-8 ct"
);

// Capricorn (Makkar)
AstrologyInfo capricorn = new
AstrologyInfo(
    "Capricorn (Makkar)",
    "Blue Sapphire (Neelam)",
    "Saturday (Shaniwar)",
    "Panch Dhatu and Silver",
    "6-8 ct"
);

// Aquarius (Kumbh)
AstrologyInfo aquarius = new AstrologyInfo(
    "Aquarius (Kumbh)",
    "Blue Sapphire (Neelam)",
    "Saturday (Shaniwar)",
    "Panch Dhatu and Silver",
    "6-8 ct"
);

// Pisces (Miet)
AstrologyInfo pisces = new AstrologyInfo(
    "Pisces (Miet)",
    "Yellow Sapphire (Pukhraj)",
    "Thursday (Guruwar)",
    "Asht Dhatu and Gold",
    "6-8 ct"
);
};
```

## 9. Conclusion

The "Future Horoscope Prediction" project aims to modernize the practice of astrology by providing an automated, reliable, and user-friendly application for generating personalized horoscope predictions. By harnessing the power of JAVA for efficient data processing and accurate calculations, the project addresses the limitations of traditional methods and existing digital solutions. The expected outcomes include enhanced user satisfaction, improved trust in predictions, and a greater understanding of astrology among users. Future developments may focus on expanding the application's features, incorporating user feedback, and exploring integrations with other technologies, such as machine learning, to further refine predictions.

## 10. Code Snippets

```
public class AstroSightAWT extends Frame {
    private TextField nameField;
    private TextArea resultArea;

    public AstroSightAWT() {
        setTitle("AstroSight");
        setSize(600, 500);
        setLayout(new BorderLayout());

        // Create components
        Label titleLabel = new Label("AstroSight", Label.CENTER);
        titleLabel.setFont(new Font("Arial", Font.BOLD, 24));

        Panel inputPanel = new Panel(new FlowLayout());
        Label nameLabel = new Label("Enter your name:");
        nameField = new TextField(20);
        inputPanel.add(nameLabel);
        inputPanel.add(nameField);

        Panel buttonPanel = new Panel(new FlowLayout());
        Button calculateButton = new Button("Calculate");
        Button clearButton = new Button("Clear");
        Button saveButton = new Button("Save Results");
        buttonPanel.add(calculateButton);
        buttonPanel.add(clearButton);
        buttonPanel.add(saveButton);

        resultArea = new TextArea(10, 50);
        resultArea.setEditable(false);

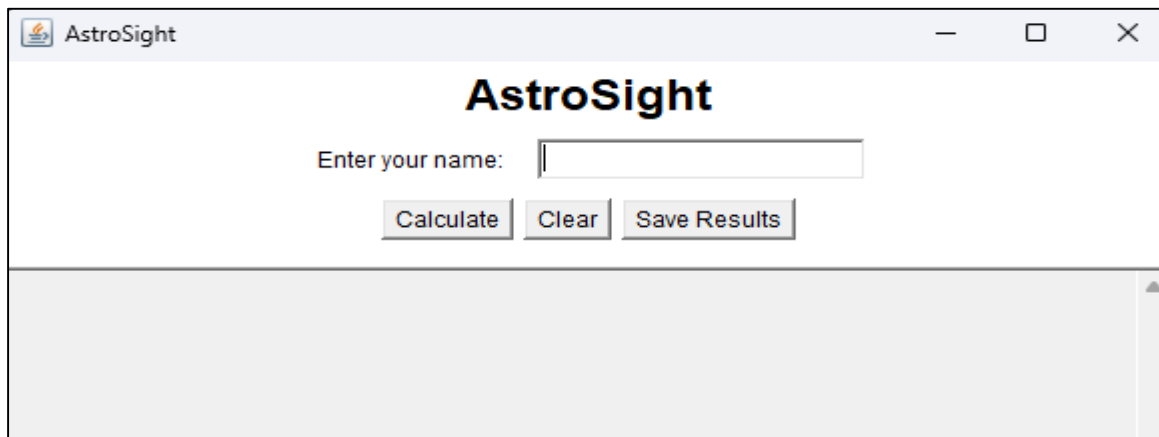
        // Add components to the frame
        add(titleLabel, BorderLayout.NORTH);
        add(inputPanel, BorderLayout.CENTER);
        add(buttonPanel, BorderLayout.SOUTH);
        add(resultArea, BorderLayout.EAST);

        // Add event handlers
        calculateButton.addActionListener(e -> calculateAstrology());
        clearButton.addActionListener(e -> {
            nameField.setText("");
            resultArea.setText("");
        });
        saveButton.addActionListener(e -> saveResults());

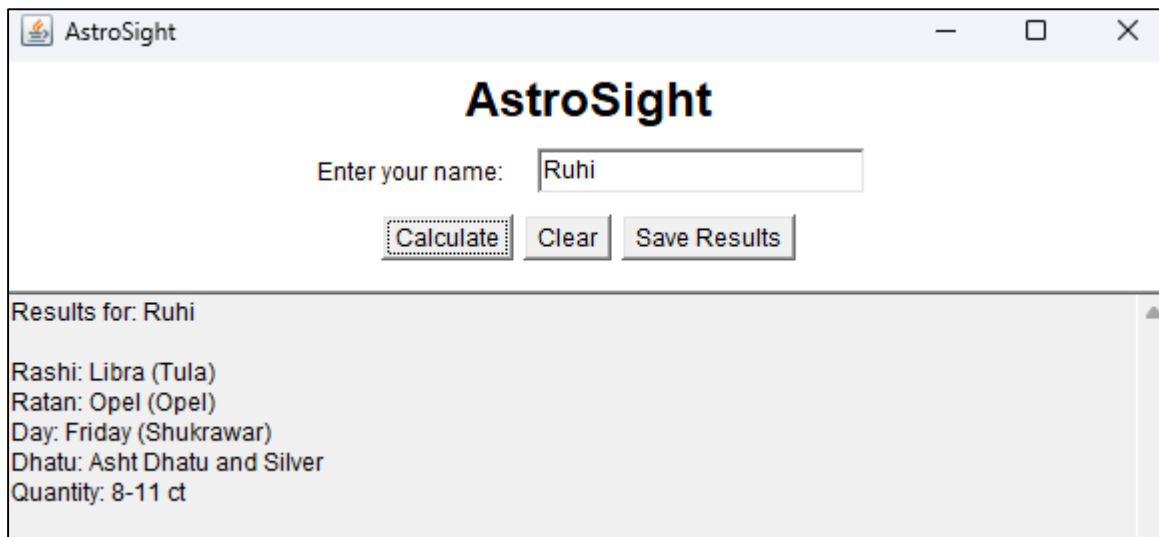
        // Add window closing event
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                dispose();
            }
        });

        setVisible(true);
    }
}
```

## 11. Outputs



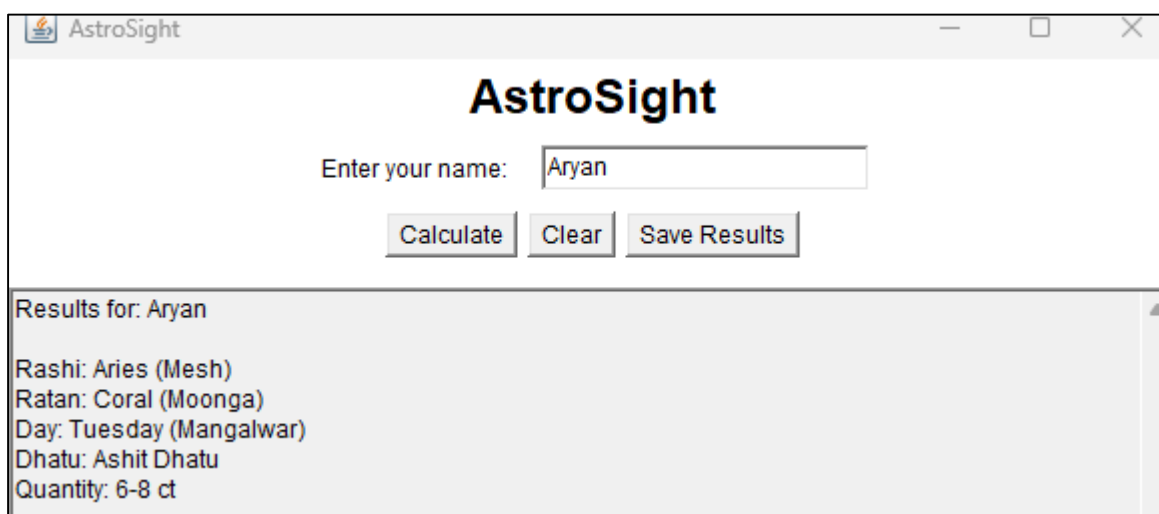
A screenshot of a web browser window titled "AstroSight". The page has a light blue header with the title "AstroSight" in bold black font. Below the header, there is a form with the label "Enter your name:" followed by a text input field. Below the input field are three buttons: "Calculate", "Clear", and "Save Results". The bottom half of the page is a large, empty light gray area, likely for displaying results.



A screenshot of the "AstroSight" application window. The name input field now contains the text "Ruhi". The "Calculate" button is highlighted with a dotted border. Below the input field, the results are displayed in a light gray box:

Results for: Ruhi

Rashi: Libra (Tula)  
Ratan: Opel (Opel)  
Day: Friday (Shukrawar)  
Dhatu: Asht Dhatu and Silver  
Quantity: 8-11 ct



A screenshot of the "AstroSight" application window. The name input field now contains the text "Aryan". The "Calculate" button is highlighted with a dotted border. Below the input field, the results are displayed in a light gray box:

Results for: Aryan

Rashi: Aries (Mesh)  
Ratan: Coral (Moonga)  
Day: Tuesday (Mangalwar)  
Dhatu: Ashit Dhatu  
Quantity: 6-8 ct

