**MONTHLY EXPENSE COMPARATOR**

**A PROJECT REPORT**

**CSA0937-PROGRAMMING IN JAVA**

***Submitted by***

**Surisetty Gayathri**

**(192110183)**

***In partial fulfilment for the award of the degree of***

**BACHELOR OF ENGINEERING IN**

**COMPUTER SCIENCE**



**SAVEETHA SCHOOL OF ENGINEERING SAVEETHA NAGAR, THANDALAM, SIMATS, CHENNAI – 602105.**

**MARCH 2024**

**BONAFIDE CERTIFICATE**

This is to certify that the project report entitled “Monthly Expense Tracker” submitted by “**Surisetty Gayathri**(192110183)”, to Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, is a record of Bonafide work carried out by him/her under my guidance. The project fulfill the requirements as per the regulations of this institution and in my appraisal meets the required standards for submission

Dr.M.Buvanesvari

Professor

Saveetha School of Engineering SIMATS, Chennai – 602 105

Internal examiner External Examiner

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENTS** | **PAGE NO** |
| 1 | ABSTRACT | 1 |
| 2 | INTRODUCTION | 2 |
| 3 | DESCRIPTION | 3 |
| 4 | SYSTEM REQUIREMENTS | 4 |
| 5 | EXISTING WORK | 5 |
| 6 | PROPOSED WORK | 6 |
| 7 | TECHNOLOGY USED | 7 |
| 8 | USE CASE DIAGRAM | 8 |
| 9 | SOURCE CODE | 9-14 |
| 10 | SCREENSHOTS(OUTPUTS) | 15-17 |
| 11 | CONCLUSION & FUTURE ENHANCEMENTS | 18 |
| 12 | REFERENCES | 19 |

# ABSTRACT

The realm of personal finance management witnesses a paradigm shift with the introduction of an innovative solution: the Monthly Expense Calculator (MEC). Embracing cutting-edge technologies, including data analytics and cloud computing, this system redefines the traditional approach to tracking and managing monthly expenditures.

The Monthly Expense Calculator aims to simplify and optimize the often task of monitoring personal finances. By leveraging intuitive interfaces and seamless integration with banking and financial platforms, users can effortlessly input and categorize their expenditures in realtime. Gone are the days of cumbersome manual tracking; the MEC automates the process, offering users a comprehensive overview of their spending habits.

Key features of the system include dynamic budget allocation, allowing users to set personalized spending limits for different expense categories. Moreover, advanced analytics tools empower individuals with insightful reports and visualizations, facilitating informed decision-making and fostering financial literacy.

Cloud computing infrastructure ensures secure data storage and accessibility across devices, enabling users to manage their finances anytime, anywhere. With its user-friendly interface and personalized insights, the Monthly Expense Calculator revolutionizes how individuals approach budgeting and financial planning, ultimately promoting financial stability and wellbeing.

# INTRODUCTION

The Monthly Expense Calculator (MEC) concept is driven by a comprehensive set of aims and objectives, tailored to revolutionize the landscape of personal finance management. Rooted in addressing the challenges individuals face in tracking and optimizing their monthly expenditures, the MEC embodies a commitment to efficiency, accuracy, and user convenience.

* Real-Time Monitoring of Expenses
* Timely Insights for Decision-Making
* Cloud-Based Storage for Remote Accessibility
* Flexibility and Accessibility
* Advanced Analytics Tools
* User-Friendly Interfaces
* Simplified Attendance Management Processes
* Integration with Organizational Platforms

# DESCRIPTION

The background of the Monthly Expense Calculator (MEC) concept arises from the challenges associated with manual expense tracking methods. Traditional approaches, such as recording expenses on paper or using cumbersome spreadsheets, are time-consuming and prone to errors. These methods often result in inaccuracies in financial records, hindering individuals from maintaining an accurate overview of their spending habits. Moreover, the lack of realtime monitoring capabilities makes it challenging for individuals to stay updated on their financial status and adjust their budgets accordingly.

The administrative burden of managing expenses manually further compounds the issue, particularly for larger households or individuals with busy schedules. This manual process consumes valuable time and resources, leading to delays in decision-making and adjustments to financial plans. Additionally, concerns about privacy and security arise when handling sensitive financial data manually. Individuals may worry about the confidentiality of their information and the risk of data breaches, prompting the need for a more secure and controlled approach to personal finance management.

Despite these challenges, technological advancements in biometrics and cloud computing offer promising solutions to streamline expense tracking. These technologies provide more accurate, efficient, and secure alternatives to traditional methods. As the complexities of modern financial management continue to grow, an automated system like the MEC becomes increasingly essential. Furthermore, recent global health crises have highlighted the importance of contactless solutions in daily activities, including expense tracking, emphasizing the necessity of technology-driven solutions to promote safety and hygiene in managing personal finances.

# SOFTWARE REQUIREMENTS

## User Authentication:

The MEC shall implement user authentication for accessing the system.

Distinct user roles, including regular users and administrators, shall have different privileges within the system.

## Expense Tracking:

Users shall have the capability to record expenses for various categories such as groceries, utilities, and entertainment.

The system shall allow users to input expenses for specific dates and allocate them to relevant categories.

## Viewing Expense Records:

Users shall be able to view their expense records for a specific date or category.

The MEC shall provide summary reports of spending trends over time, including graphical representations for better visualization.

## Performance:

The MEC shall support multiple users accessing the system simultaneously without experiencing significant performance degradation.

Database queries and processing tasks shall be optimized to ensure efficient system performance.

## Usability:

The user interface of the MEC shall be developed using intuitive design principles.

The system shall include tooltips and error messages to guide users and provide assistance when needed.

User feedback mechanisms shall be incorporated to gather suggestions for improving system usability.

# EXISTING WORK

* A personal finance app tailored for individual users, offering expense tracking features along with budget management tools.
* An expense management software designed for small businesses, providing capabilities for tracking expenses, generating reports, and managing budgets.
* A web-based expense tracker focused on freelance professionals and self-employed individuals, offering invoicing and expense categorization functionalities.
* A mobile expense management app integrated with banking services, enabling users to track expenses, set budget goals, and receive personalized financial insights.
* An open-source expense tracking platform suitable for nonprofit organizations, featuring customizable expense categories, donation tracking, and budget planning tools.
* An expense management tool designed for travel and hospitality businesses, offering features for tracking travel expenses, managing reimbursements, and generating expense reports.
* A cloud-based expense management solution targeted at enterprise clients, providing expense tracking, approval workflows, and integration with accounting software.
* A personal finance software with expense tracking capabilities, offering budgeting, investment tracking, and financial goal setting functionalities.
* An expense tracking app tailored for freelancers and gig workers, featuring integration with gig economy platforms, real-time expense monitoring, and tax deduction tracking.
* A web-based expense management system designed for educational institutions, offering expense tracking, budget allocation, and financial reporting functionalities.
* Develop a user-friendly interface with intuitive navigation for ease of use.
* Implement expense categorization functionality for organizing spending into different categories.
* Enhance expense tracking capabilities by integrating with bank accounts and credit cards.
* Develop customizable reporting tools for generating detailed expense reports.
* Implement mobile application version of the MEC for on-the-go expense tracking.

# PROPOSED WORK

* Implement expense categorization functionality for organizing spending into different categories.
* Develop a user-friendly interface with intuitive navigation for ease of use.
* Introduce budgeting features to allow users to set and track financial goals.
* Enhance expense tracking capabilities by integrating with bank accounts and credit cards.
* Implement customizable reporting tools for generating detailed expense reports.
* Introduce reminders and alerts for upcoming bills and financial commitments.
* Develop a mobile application version of the MEC for on-the-go expense tracking.
* Explore options for integration with digital payment platforms to streamline expense management.
* Enhance security measures to protect user data and financial information.
* Incorporate machine learning algorithms for predictive expense analysis and budget optimization.
* Introduce gamification elements to encourage users to stick to their budgeting goals.
* Develop collaborative features for sharing expense data with family members or financial advisors.
* Implement data backup and synchronization functionalities to prevent loss of financial records.
* Introduce a subscription-based model with premium features for advanced users.
* Develop tutorials and educational resources to help users maximize the potential of the MEC.
* Integrate with tax preparation software to simplify the process of filing taxes.
* Explore partnerships with financial institutions to offer exclusive benefits to MEC users.
* Implement customizable expense templates for recurring expenses such as rent and utilities..

# TECHNOLOGY USED

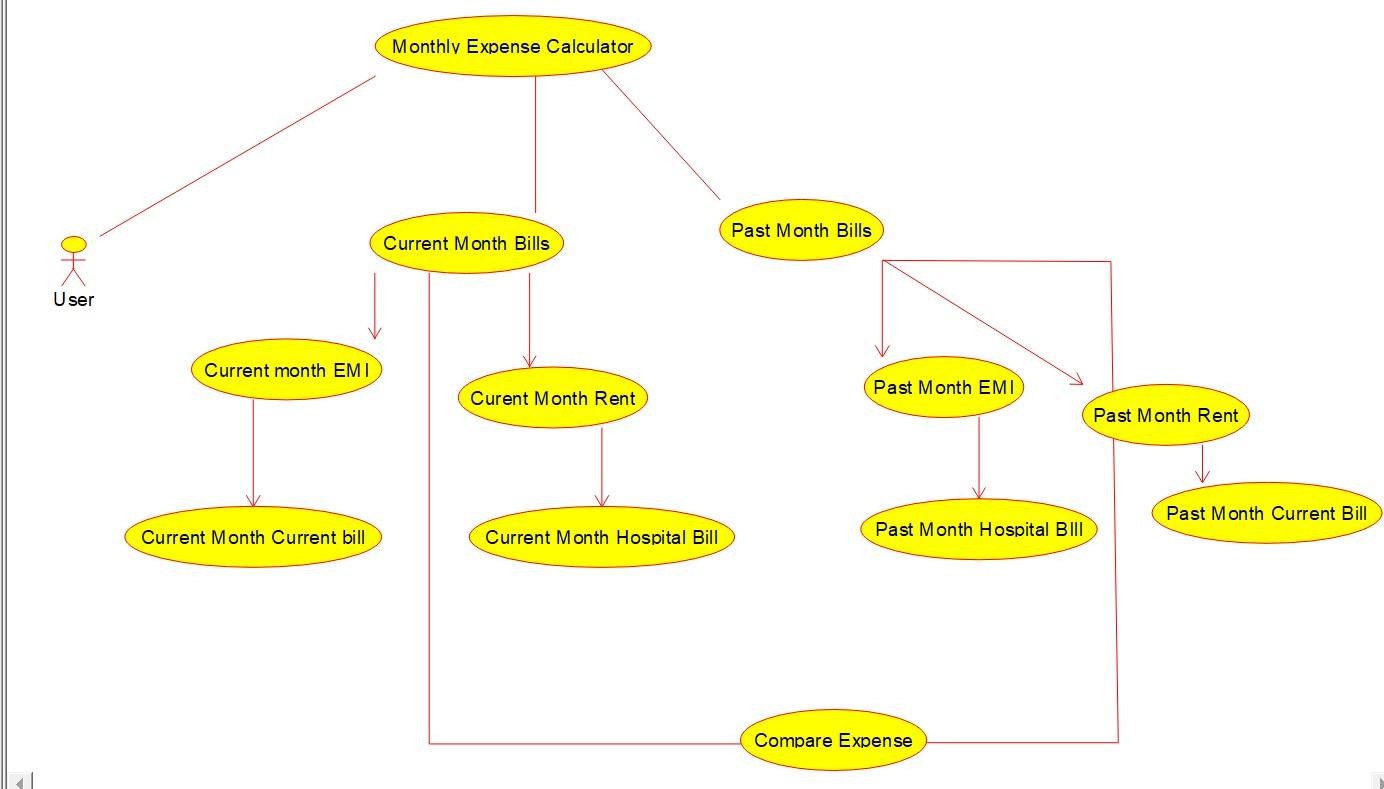
The Monthly Expense Tracker, developed using Java Swing technology, offers a robust and user-friendly solution for efficiently managing personal finances. Leveraging Java Swing's platform-independent graphical user interface (GUI) toolkit, the system ensures accessibility and consistency across different operating systems.

Java Swing components are employed to create an intuitive and interactive interface, enabling users to effortlessly navigate and perform expense-related tasks. The use of JFrame, JPanel, JButton, and other Swing components ensures a visually appealing and responsive user experience.

Through the Monthly Expense Tracker's Java Swing interface, users can easily input expense data, view expense reports, and manage financial records with simplicity. The system leverages the event-driven programming model of Swing, facilitating seamless interaction with buttons, text fields, and other GUI elements.

In Java Swing, JLabel and JButton are fundamental components contributing to the creation of interactive graphical user interfaces (GUIs). The JLabel class represents non-editable text or image labels, commonly used to display information, provide descriptions, or act as identifiers within the GUI. Developers can customize the appearance of JLabel instances by adjusting attributes such as font, text colour, and alignment. JLabel serves as a versatile tool for presenting static content in various Swing applications.

# USE CASE DIAGRAM



## Fig 1 Use case diagram of Monthly Expense Calculator

**FIG 1** Shows that Monthly Expense Calculator which contains monthly expenses and compare with past month and display the solution as it is Increased or Decreased. This Monthly Expense Calculator display Comparative Solution.

# SOURCE CODE

import javax.swing.\*; import java.awt.\*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener;

public class MonthlyExpenseTrackerApp extends JFrame { private JTextField pastMonthRentField, pastMonthHospitalBillField, pastMonthCurrentBillField, pastMonthEMIField, pastMonthTaxField; private JTextField currentMonthRentField, currentMonthHospitalBillField,

currentMonthCurrentBillField, currentMonthEMIField, currentMonthTaxField; private JButton compareButton;

private JLabel pastMonthTotalLabel, currentMonthTotalLabel, pastMonthTotalValueLabel, currentMonthTotalValueLabel; private JTextField usernameField; private JPasswordField passwordField; private JButton loginButton;

public MonthlyExpenseTrackerApp() { setTitle("Monthly Expense Tracker"); setSize(400, 500);

setDefaultCloseOperation(EXIT\_ON\_CLOSE); setLayout(new GridLayout(6, 1));

// Initialize components

JPanel loginPanel = new JPanel(new GridLayout(3, 2, 5, 5));

JPanel pastMonthPanel = new JPanel(new GridLayout(3, 2, 5, 5));

JPanel currentMonthPanel = new JPanel(new GridLayout(3, 2, 5, 5)); JPanel buttonPanel = new JPanel(new FlowLayout()); pastMonthRentField = new JTextField(8);

pastMonthHospitalBillField = new JTextField(8); pastMonthCurrentBillField = new JTextField(8); pastMonthEMIField

= new JTextField(8);

pastMonthTaxField = new JTextField(8); currentMonthRentField = new JTextField(8); currentMonthHospitalBillField = new JTextField(8); currentMonthCurrentBillField = new JTextField(8); currentMonthEMIField = new JTextField(8); currentMonthTaxField = new JTextField(8); compareButton =

new JButton("Compare Expenses"); pastMonthTotalLabel = new JLabel("Past Month Total: "); currentMonthTotalLabel = new JLabel("Current Month Total: "); pastMonthTotalValueLabel = new JLabel("0.0"); currentMonthTotalValueLabel = new JLabel("0.0"); usernameField = new JTextField(10); passwordField = new JPasswordField(10); loginButton = new JButton("Login");

// Add components to the login panel loginPanel.add(new JLabel("Username:")); loginPanel.add(usernameField); loginPanel.add(new JLabel("Password:"));

loginPanel.add(passwordField); loginPanel.add(new JLabel()); loginPanel.add(loginButton);

// Add components to the past month panel pastMonthPanel.add(new JLabel("Past Month Rent:")); pastMonthPanel.add(pastMonthRentField);

pastMonthPanel.add(new JLabel("Past Month Hospital Bill:")); pastMonthPanel.add(pastMonthHospitalBillField); pastMonthPanel.add(new JLabel("Past Month Current Bill:")); pastMonthPanel.add(pastMonthCurrentBillField);

pastMonthPanel.add(new JLabel("Past Month EMI:")); pastMonthPanel.add(pastMonthEMIField); pastMonthPanel.add(new JLabel("Past Month Tax:")); pastMonthPanel.add(pastMonthTaxField); pastMonthPanel.add(pastMonthTotalLabel); pastMonthPanel.add(pastMonthTotalValueLabel);

// Add components to the current month panel currentMonthPanel.add(new JLabel("Current Month Rent:")); currentMonthPanel.add(currentMonthRentField);

currentMonthPanel.add(new JLabel("Current Month Hospital Bill:")); currentMonthPanel.add(currentMonthHospitalBillField); currentMonthPanel.add(new JLabel("Current Month Current Bill:")); currentMonthPanel.add(currentMonthCurrentBillField); currentMonthPanel.add(new JLabel("Current Month EMI:")); currentMonthPanel.add(currentMonthEMIField); currentMonthPanel.add(new JLabel("Current Month Tax:"));

currentMonthPanel.add(currentMonthTaxField); currentMonthPanel.add(currentMonthTotalLabel); currentMonthPanel.add(currentMonthTotalValueLabel);

// Add button to the button panel buttonPanel.add(compareButton);

// Add panels to the frame add(loginPanel); add(pastMonthPanel); add(currentMonthPanel);

add(buttonPanel);

// Add action listener to the login button loginButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// Check login credentials

String username = usernameField.getText();

String password = new String(passwordField.getPassword()); if (username.equals("narendra") && password.equals("12345")) { JOptionPane.showMessageDialog(null, "Login Successful!");

// Enable input fields and buttons after successful login enableInputFields(true);

} else {

JOptionPane.showMessageDialog(null, "Invalid username or password.");

// Clear password field passwordField.setText("");

}

}

});

// Add action listener to the "Compare Expenses" button compareButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// Calculate total expenses for past month and current month

double pastMonthTotal = getTotal(pastMonthRentField, pastMonthHospitalBillField, pastMonthCurrentBillField, pastMonthEMIField, pastMonthTaxField);

double currentMonthTotal = getTotal(currentMonthRentField, currentMonthHospitalBillField, currentMonthCurrentBillField,

currentMonthEMIField, currentMonthTaxField);

// Update total labels pastMonthTotalValueLabel.setText(String.valueOf(pastMonthTotal));

currentMonthTotalValueLabel.setText(String.valueOf(currentMonthTotal));

// Compare total expenses

if (currentMonthTotal > pastMonthTotal) {

JOptionPane.showMessageDialog(null, "Total expenses for the current month increased compared to the past month.");

} else if (currentMonthTotal < pastMonthTotal) {

JOptionPane.showMessageDialog(null, "Total expenses for the current month decreased compared to the past month.");

} else {

JOptionPane.showMessageDialog(null, "Total expenses for the current month remained the same compared to the past month.");

}

}

});

// Disable input fields and buttons until successful login enableInputFields(false);

}

private void enableInputFields(boolean enable) { pastMonthRentField.setEnabled(enable); pastMonthHospitalBillField.setEnabled(enable); pastMonthCurrentBillField.setEnabled(enable); pastMonthEMIField.setEnabled(enable); pastMonthTaxField.setEnabled(enable); currentMonthRentField.setEnabled(enable); currentMonthHospitalBillField.setEnabled(enable); currentMonthCurrentBillField.setEnabled(enable); currentMonthEMIField.setEnabled(enable); currentMonthTaxField.setEnabled(enable); compareButton.setEnabled(enable);

}

private double getTotal(JTextField... fields) { double total = 0.0; for (JTextField field :

fields) { try {

total += Double.parseDouble(field.getText().trim());

} catch (NumberFormatException e) {

// Ignore non-numeric inputs

} }

return total; }

public static void main(String[] args) { SwingUtilities.invokeLater(new Runnable() {

@Override public void run() {

MonthlyExpenseTrackerApp app = new MonthlyExpenseTrackerApp(); app.setVisible(true);

}

});

}

}

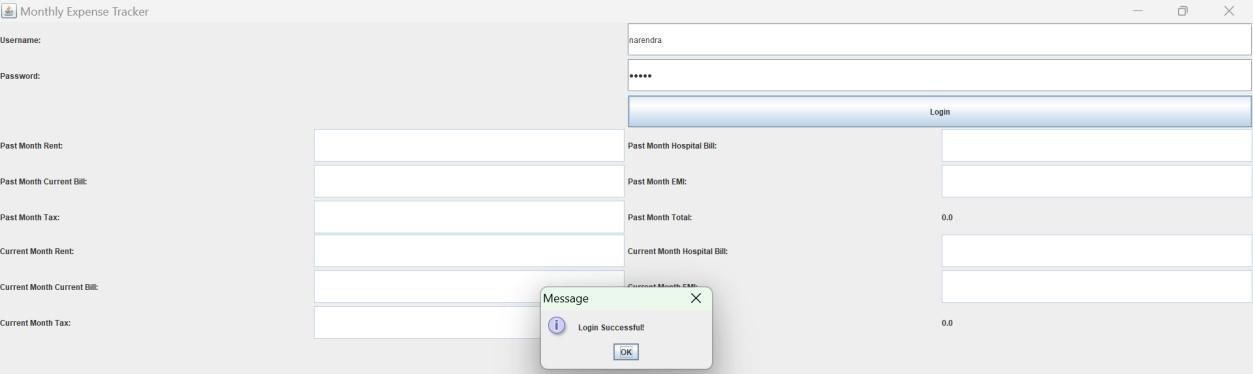
# OUTPUT

**Fig 2** The interface of an attendance app typically features a login screen with fields for entering the username and password, ensuring secure and personalized access. The username field allows users to input their unique identification, often assigned by the system administrator.



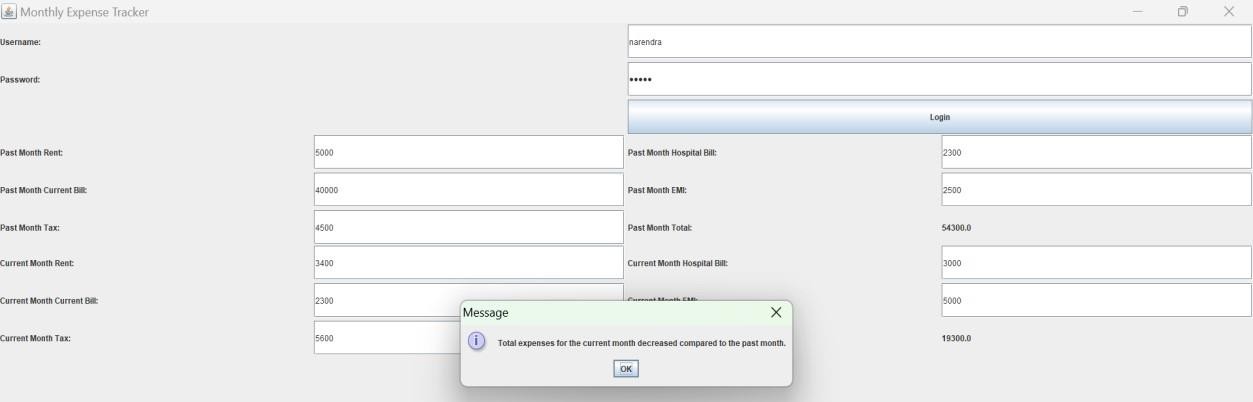
## Fig 2 Login Page

**Fig 3** After a successful login into the Monthly Expense Calculator app, users are seamlessly transitioned into the main interface.



## Fig 3 Login Successful or Invalid username

**Fig 4** To Compare Past Month Expense and present Month Expense. It shows Comparison Like Increased or Decreased.



**Fig 4 Showing Comparative Analysis as Increased or Decreased**

# CONCLUSION

The implementation of the proposed Monthly Expense Calculator represents a significant advancement in personal finance management, surpassing the limitations of manual tracking methods. Scalability of the system ensures its adaptability to evolving financial needs, making it a sustainable solution for individuals and businesses alike. Furthermore, its seamless integration with existing systems enables smooth data flow across different financial platforms, fostering consistency and reducing manual efforts in managing personal finances. Overall, the Monthly Expense Calculator revolutionizes the way individuals approach budgeting and financial planning, promoting financial stability and well-being.

Moreover, exploring opportunities for integration with emerging payment systems, such as digital wallets and cryptocurrency platforms, can provide users with more diverse and flexible options for managing their finances in an increasingly digital world. By staying at the forefront of technological advancements and listening closely to user feedback, the Monthly Expense Calculator remains dedicated to empowering individuals with the tools and insights they need to achieve financial wellness and prosperity.

# FUTURE ENHANCEMENT

* Integration with Financial Institutions.
* Machine Learning and Predictive Analytics.
* Expense Categorization and Tagging.
* Expense Comparison and Benchmarking.
* Mobile Application Development.
* Financial Goal Setting and Tracking.
* Expense Collaboration and Sharing.
* Security Enhancements.
* Integration with IoT Devices.
* Customization and Personalization.

The Monthly Expense Calculator holds immense potential for future enhancements and innovations. Integration with financial institutions' APIs would streamline expense tracking, while incorporating machine learning and predictive analytics could provide users with insightful predictions and proactive budgeting assistance. Further advancements in expense categorization, comparison with benchmarks, and the development of a mobile application would enhance accessibility and usability for users across various platforms

Additionally, integration with IoT devices could introduce automated expense tracking based on user behaviour. Customization options would allow users to tailor the system to their individual needs and preferences, ensuring a personalized and efficient financial management experience. Overall, these future developments promise to elevate the Monthly Expense Calculator to new heights, empowering users to take control of their finances with greater ease and precision.

# REFERENCES

<https://scholar.google.com/citations?user=sXhBIVgAAAAJ&hl=en&oi=sra> <https://www.theseus.fi/bitstream/handle/10024/801991/THESIS.pdf?sequence=2> <https://ieeexplore.ieee.org/abstract/document/10170735/> <https://ieeexplore.ieee.org/abstract/document/9917982/> <https://scholar.google.com/citations?user=f2TZYp4AAAAJ&hl=en&oi=sra> <https://ijresonline.com/assets/year/volume-10-issue-3/IJRES-V10I3P108.pdf> <https://www.theseus.fi/handle/10024/785937> <https://www.academia.edu/download/53337467/Manchanda_Angad.pdf> [https://www.irjmets.com/uploadedfiles/paper/issue\_4\_april\_2022/21604/final/fin\_irjmets165](https://www.irjmets.com/uploadedfiles/paper/issue_4_april_2022/21604/final/fin_irjmets1651132467.pdf) [1132467.pdf](https://www.irjmets.com/uploadedfiles/paper/issue_4_april_2022/21604/final/fin_irjmets1651132467.pdf) <https://dl.ucsc.cmb.ac.lk/jspui/handle/123456789/4424>