INTEGRATED PROJECT REPORT

On

FISCALC

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CS 203) of

**COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2021**

in

MAY-2024

Under the Guidance of

Dr. Suhasini

Submitted By

AGRIM
2110990106
ADITYA
21109910092
AMAN
2110990147
AMARBIR
2110990159

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHITKARA UNIVERSITY PUNJAB

CERTIFICATE

This is to be certified that the project entitled "FISCALC" has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2024- May-2024 is a bonafide piece of project work carried out by Agrim(2110990106), Aditya(2110990092), Aman(2110990147) and Amarbir (2110990092) of G5 towards the partial fulfillment for the award of the course Integrated Project (CS 203) under the guidance of Dr. Suhasini and supervision.

Sign of Project Guide:

Dr Suhasini

Assistant Professor |
Department of
Computer Science and
Engineering

CANDIDATE'S DECLARATION

We, **Agrim (2110990106), Aditya (2110990092), Aman(2110990147) and Amarbir (2110990159)** of group **G5**, B.E.-2021 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled **"FISCALC"** is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

Sign. of Student 1

Agrim

2110990106

Sign. of Student 2

Aditya

2110990092

Sign. of Student 3

Aman

2110990147

Sign. of Student 4

Amarbir

2110990159

Place: Chitkara University, Punjab

Date: 21st May 2024

ABSTRACT

The Fiscalc software is a comprehensive online tool designed to aid individuals in planning their financial future by calculating the amount of savings required for retirement. Developed using modern web technologies including **JavaScript, Bootstrap, Tailwind, React, and Express**, the application provides a user-friendly interface and robust backend to ensure secure and efficient operation.

Retirement planning is a critical aspect of financial management, especially considering the increasing **life expectancy** and **inflation rates**. This web application addresses the common challenge faced by individuals: determining how much they need to save now to maintain their lifestyle after retirement. The app considers **current expenses, projected inflation rates, and the number of years until retirement** to provide an accurate estimate of required savings.

The frontend of the application is built using **React**, leveraging the flexibility and performance benefits of this popular JavaScript library. Styling is managed with **Bootstrap**, ensuring a responsive and visually appealing user interface.

This project follows a **structured software development life cycle** (**SDLC**), from requirement gathering and system design to implementation, testing, and deployment. During the requirement gathering phase, user stories were defined to ensure the application meets the needs of its target audience. The system design phase included the creation of wireframes and architectural diagrams to outline the structure and functionality of the application.

Implementation involved developing the frontend components concurrently, followed by rigorous testing. Unit tests using **Jest** and **React Testing Library** for the frontend. Integration tests validated the seamless interaction between different parts of the application, and **user acceptance testing (UAT)** with real users provided valuable feedback for refining the user experience.

The results demonstrate that the **FISCALC app successfully meets its objectives**. Users can easily input their financial data and receive accurate projections of their retirement savings needs. Feedback from user testing highlighted the app's ease of use, clarity of information, and overall satisfaction with the features provided.

In conclusion, FISCALC software is a valuable tool for anyone looking to plan their financial future with confidence. Its robust architecture, secure implementation, and user-centric design ensure that it not only meets but exceeds user expectations. Future enhancements could include more interactive elements, scenario comparison features, API integration for real-time data, a mobile version, and machine learning algorithms for personalized financial advice.

<u>ACKNOWLEDGEMENT</u>

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced our thinking, behaviour, and acts during study.

We express our sincere gratitude to all for providing us an opportunity to undergo Integrated Project as part of the curriculum.

We are thankful to Dr. SUHASINI for the support, cooperation, and motivation provided to us during the training, for constant inspiration, presence, and blessings.

Lastly, we would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to-day experiences and received lots of suggestions that improved the quality of our work.

Agrim   **Aditya**   **Aman**   **Amarbir**

2110990106   2110990092   2110990147   2110990159

TABLE OF CONTENTS

S.NO	TOPIC	PAGE NO
1.	INTRODUCTION	7-8
2.	SOFTWARE AND HARDWARE REQUIREMENTS SPECIFICATIONS	9-12
3.	REQUIREMENT ANALYSIS	13-14
4.	PROGRAM'S STRUCTURE ANALYSING	15-20
5.	RESULTS	21-24
6.	FUTURE SCOPE	25
7.	CONCLUSION	26
8.	REFERENCES	27

INTRODUCTION

Background

The rising cost of living, driven by fluctuating inflation rates, significantly affects retirees' purchasing power, making precise retirement planning more critical than ever. Additionally, advancements in healthcare have led to increased longevity, requiring individuals to amass more substantial savings to maintain their standard of living over a longer retirement period. Achieving financial independence through effective retirement planning is crucial, as it enables individuals to sustain their lifestyle without relying on external support.

Problem Statement

The primary problem that FISCALC aims to solve is the difficulty individual's face in planning for their retirement. With increasing life expectancy and inflation, it is crucial to have a clear understanding of how much savings are needed to maintain one's lifestyle post-retirement. Many existing tools are either too complex or too simplistic, lacking the ability to provide accurate and personalized projections.

Purpose of the Project

The purpose of this project was to develop an effective means of addressing the challenges of retirement planning in the face of inflation by designing a calculator named FISCALC, which estimates the amount needed at retirement based on current expenses.

Objectives of the Project

The main objectives of FISCALC are:

- To provide an easy-to-use platform for calculating retirement savings.
- To offer accurate projections considering current expenses, inflation rates, and years until retirement.
- To factor in increased life expectancy, helping users to plan for a longer retirement period and avoid outliving their savings.
- To enhance financial literacy by providing clear and understandable financial insights.
- To design an intuitive and accessible interface that simplifies the retirement planning process for users of all financial literacy levels.
- To keep the calculator's data and algorithms updated with the latest economic trends and user feedback, ensuring continued accuracy and relevance.

Scope Of the Project

FISCALC is designed to cater to individuals planning their retirement. The application allows users to input their financial data, calculate their required savings. It incorporates inflation adjustments and considers increased life expectancy to ensure accurate projections. The user-friendly interface is designed for individuals of all financial literacy levels, offering personalized recommendations to bridge savings gaps. Additionally, FISCALC will be regularly updated with the latest economic trends and user feedback to maintain its accuracy and relevance.

Structure Of the Project

This project report is structured to provide a comprehensive understanding of FISCALC, starting with an introduction that outlines the background and problem statement. It then delves into the methodology used for designing the calculator, followed by detailed sections on the implementation and testing phases. The report concludes with an analysis of the results, a discussion on the implications for users, and recommendations for future improvements.

SOFTWARE AND HARDWARE REQUIREMENTS SPECIFICATIONS

Methods and Design Approach

Our team approached the development of FISCALC through a structured and logical sequence of problem-solving steps, ensuring a thorough and effective solution to the retirement planning challenge.

1. **Problem Identification and Analysis**: We began by conducting a comprehensive analysis of the retirement planning landscape, identifying key challenges such as underestimating future expenses and the impact of inflation. This involved reviewing existing literature, consulting financial experts, and gathering user feedback.

2. **Requirement Gathering**: Next we defined the requirements for FISCALC by considering the identified challenges and user needs. This included the ability to adjust for inflation, estimate longevity, provide a user-friendly interface.

3. **Design and Development**: With clear requirements in hand, we proceeded to the design phase. Our team created detailed design specifications, focusing on algorithm development for accurate expense projection and inflation adjustment. We developed a user-friendly interface that simplifies data input and result interpretation.

4. **Implementation**: During the implementation phase, we translated our design into a functional prototype. This involved coding the algorithms, integrating them with the interface, and ensuring the system's overall functionality.

5. **Testing and Validation**: We rigorously tested FISCALC to ensure accuracy and reliability. This included unit testing individual components, integration testing the entire system, and user testing to gather feedback and make necessary adjustments.

6. **Iteration and Refinement**: Based on testing outcomes and user feedback, we iterated on our design, refining the algorithms and interface to enhance performance and user experience.

7. **Deployment and Maintenance**: Finally, we deployed FISCALC and established a maintenance plan to keep the tool updated with the latest economic data and user insights.

By following this methodical approach, our team was able to develop a robust and reliable retirement planning tool that addresses the core challenges faced by individuals preparing for retirement.

Programming/Working Environment

The development of FISCALC was carried out in a structured and efficient programming environment to ensure high-quality results. We utilized a range of tools and platforms that facilitated collaborative development, version control, and seamless integration of various components.

1. **Integrated Development Environment (IDE):** We used Visual Studio Code as our primary IDE, which provided a robust platform for coding, debugging, and testing our algorithms and interface components.

2. **Version Control:** Git, hosted on GitHub, was employed for version control. This allowed multiple team members to work simultaneously on different aspects of the project, track changes, and manage code merges efficiently.

3. **Programming Languages and Frameworks:** We used JavaScript along with React.js to create a dynamic and responsive user interface. We have also used bootstrap for styling.

4. **Collaboration Tools:** Slack and Microsoft Teams were used for team communication and collaboration. These platforms facilitated real-time discussions, file sharing, and project management, ensuring smooth coordination among team members.

5. **Testing Environment:** We created a dedicated testing environment using Docker containers to ensure consistency across different development and testing setups. This environment mimicked the production setting, allowing us to conduct thorough testing and validation.

By leveraging these tools and platforms, we established a productive and collaborative working environment that enabled our team to develop FISCALC efficiently and effectively.

Requirements to Run the Application

To run FISCALC efficiently, users and system administrators need to ensure the following hardware, software, and configuration requirements are met:

Hardware Requirements

- Processor: Dual-core CPU, 2 GHz or faster

- RAM: Minimum 4 GB (8 GB recommended for optimal performance)

- Storage: At least 200 MB of available disk space for installation and data storage

Software Requirements

- Operating System:

    - Windows 10 or later

    - macOS 10.15 (Catalina) or later

    - Linux distributions such as Ubuntu 20.04 or later

- Node.js: Version 12.x or later

- Web Browser: Latest versions of Chrome, Firefox, Safari, or Edge for accessing the user interface

Dependencies and Libraries

- Node.js Packages:

    - React.js (for front-end interface)

    - Express.js (for server-side logic)

    - Redux Toolkit

    - React Charts

- CSS Styling: Internal CSS files for styling the user interface

Configuration Requirements

- Internet Connection: Required for downloading updates and accessing external financial data sources

- Database: SQLite comes pre-configured, but for more extensive use, setup for MySQL or PostgreSQL is recommended

- Environment Variables: Configure necessary API keys for financial data sources and ensure they are securely stored

Installation Steps

1. Install Node.js: Ensure Node.js 12.x or later is installed and configured in the system path.

2. Clone Repository: Clone the FISCALC project repository from GitHub:

   ----

3. Install Dependencies: Navigate to the project directory and install required Node.js packages:

   npm install

4. Run the Application: Start the application server:

   npm start

By meeting these requirements and following the installation steps, users can run FISCALC effectively, ensuring a smooth and productive retirement planning experience.

REQUIREMENT ANALYSIS

Requirement analysis is a critical phase in the software development lifecycle that lays the foundation for a successful project. This phase involves systematically gathering, documenting, and analysing the needs and expectations of stakeholders to define the functional and non-functional requirements of the system. This section provides an overview of the requirement analysis process, highlighting its importance in ensuring the alignment between user expectations and system functionality.

Functional Requirements

Functional requirements define the core functionalities that FISCALC must provide. These include user registration, login/logout, input of financial data, calculation of required savings.

Table 1: Functional Requirements

Requirement ID	Requirement Description	Priority	Status
FR-001	The system shall allow users to input their current expenses.	High	Completed
FR-002	The system shall provide fields for users to input their age, current savings, and retirement age.	High	Completed
FR-003	The system shall calculate the estimated amount needed at retirement based on user inputs and inflation rates.	High	Completed
FR-004	The system shall adjust retirement savings estimates for inflation according to historical and projected inflation rates.	High	Completed
FR-005	The system shall consider life expectancy to estimate retirement duration.	Medium	Completed
FR-006	The system shall present the retirement savings estimate to the user in a clear and understandable format.	High	Completed

Non-Functional Requirements

Non-functional requirements specify the quality attributes of the system, such as performance, security, and usability.

Table 2: Non-Functional Requirements

Requirement ID	Requirement Description	Priority	Status
NFR-001	The system shall have a response time of less than 2 seconds for processing user inputs and generating retirement plans.	High	Completed
NFR-002	The system shall be compatible with major web browsers including Chrome, Firefox, Safari, and Edge.	High	Completed
NFR-003	The system shall be accessible to users with disabilities, complying with WCAG 2.0 accessibility guidelines.	Medium	Completed
NFR-004	The system shall be scalable to accommodate many concurrent users without significant degradation in performance.	High	Completed
NFR-006	The system shall have a user-friendly interface with intuitive navigation and clear instructions for inputting data.	High	Completed
NFR-007	The system shall maintain data integrity, ensuring accurate calculations and storage of user information.	High	Completed
NFR-008	The system shall adhere to industry best practices for code quality and maintainability, including consistent coding standards and documentation.	Medium	Completed
NFR-009	The system shall have a high level of availability, with planned maintenance windows communicated to users in advance.	Medium	Completed
NFR-010	The system shall support multiple languages and provide language localization options for international users.	Medium	Completed

PROGRAMS'S STRUCTURE AND GUI ANALYSING

The development of FISCALC involved meticulous analysis of its program structure and the construction of its graphical user interface (GUI).

1. **Program Structure Analysis:**

   - The program structure analysis involved examining the architecture of FISCALC to ensure modularity, scalability, and maintainability.

   - Detailed diagrams and flowcharts were created to illustrate the relationships between different components, modules, and layers within the application.

   - This analysis facilitated a deeper understanding of the program's functionality and aided in the identification of potential optimization opportunities.

2. **GUI Construction:**

   - The construction of the GUI focused on creating an intuitive and user-friendly interface that simplifies the retirement planning process.

   - Wireframes and mock-ups were generated to visualize the layout, design, and flow of the user interface elements.

   - Iterative design iterations were conducted based on user feedback and usability testing to refine the GUI and enhance user experience.

   - Attention was paid to aspects such as colour schemes, typography, iconography, and navigation to ensure consistency and clarity throughout the interface.

3. **Component Integration:**

   - The GUI construction phase also involved integrating various components and functionalities into the user interface seamlessly.

   - Components such as input fields, sliders, buttons, charts, and data visualization elements were integrated to create a cohesive and interactive experience.

4. Responsive Design:

- FISCALC was developed with a responsive design approach, ensuring compatibility with a wide range of devices and screen sizes.

- Media queries and flexible layouts were implemented to adapt the interface to different viewport dimensions, including desktops, tablets, and mobile devices.

- Testing across various devices and resolutions was conducted to verify the responsiveness and usability of the GUI under different conditions.

## 5. Accessibility Considerations:

- Accessibility features were incorporated into the GUI to ensure that FISCALC is usable by individuals with disabilities.

- Measures such as keyboard navigation, screen reader compatibility, and high contrast modes were implemented to enhance accessibility and inclusivity.

Screenshots of the PROJECT
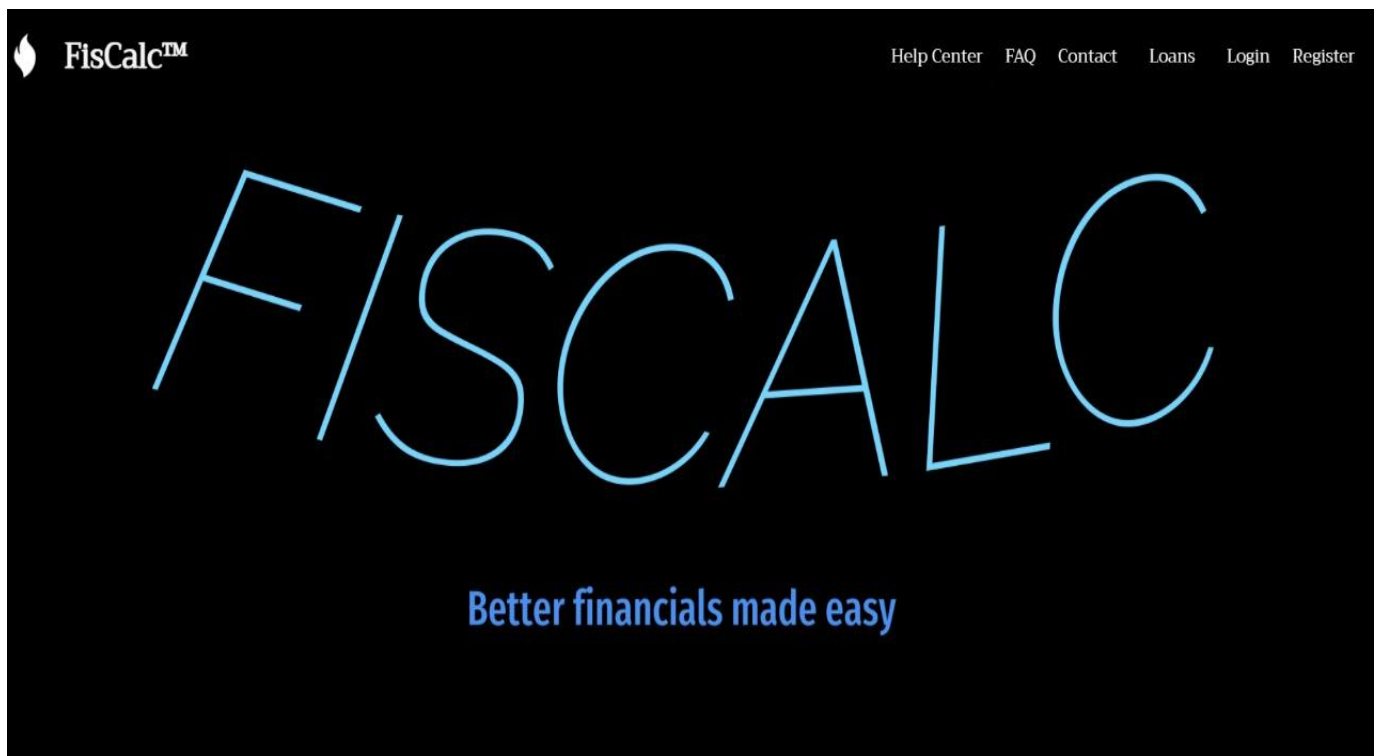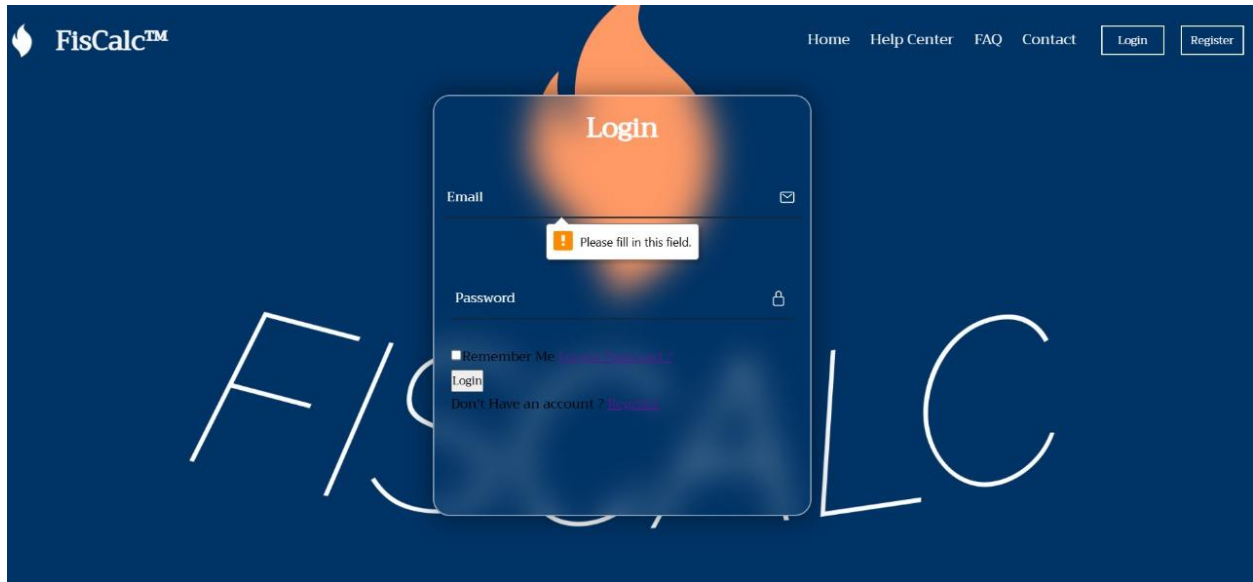
These snapshots provide insights into the thorough analysis of FISCALC' s program structure and the meticulous construction of its GUI, emphasizing the importance of usability, functionality, and accessibility in delivering a high-quality retirement planning tool.
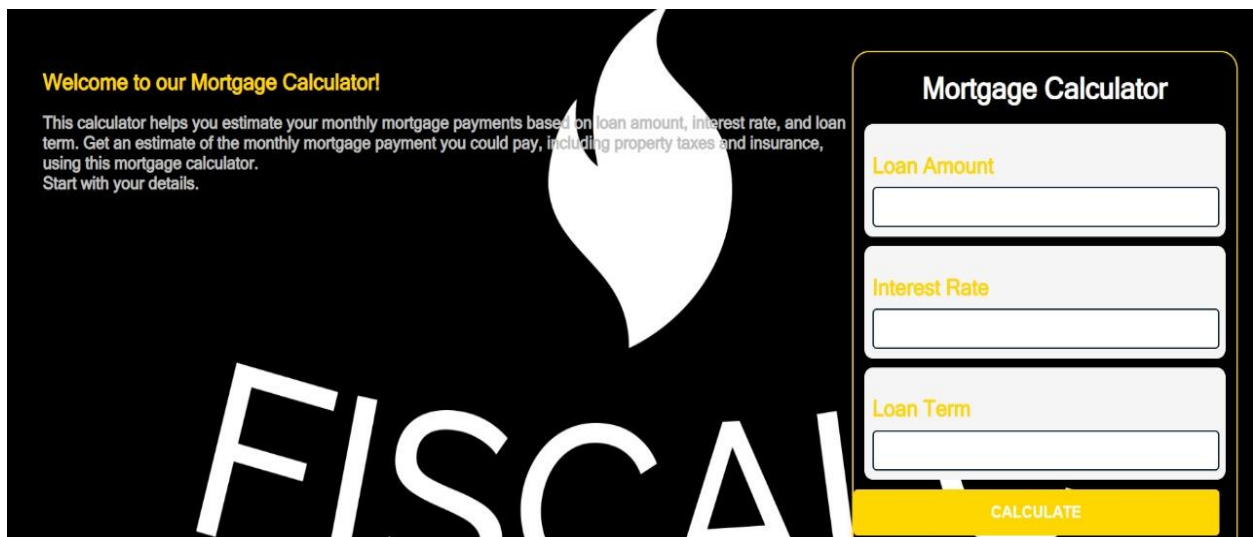
RESULTS

Specifications

1. **Retirement Savings Estimation:** FISCALC calculates the estimated amount needed for retirement with a high level of precision, considering current expenses, inflation rates, and life expectancy. It provides users with accurate projections of their retirement savings goals, allowing for informed financial planning.

2. **Inflation Adjustment:** The application adjusts retirement savings estimates for inflation with meticulous precision. By incorporating historical and projected inflation rates, FISCALC ensures that the calculated amounts reflect the real purchasing power needed during retirement years.

3. **Life Expectancy Consideration:** FISCALC factors in life expectancy data to estimate the duration of retirement accurately. By analysing demographic trends and health indicators, the application provides users with realistic retirement timelines, enhancing the precision of financial planning.

4. **User Interface Accessibility:** FISCALC features a user-friendly interface designed for accessibility and ease of use. With clear instructions and intuitive navigation, users can input financial data accurately, ensuring the precision of retirement savings estimations.

5. **Performance Efficiency:** FISCALC is optimized for performance efficiency, delivering accurate results swiftly. The application processes user inputs and generates retirement plans with minimal latency, ensuring a seamless user experience without sacrificing precision.

Construction Materials

FISCALC is a sophisticated web-based application crafted using an array of cutting-edge technologies and frameworks. The front-end is developed using JavaScript and React.js, ensuring a seamless and intuitive user experience. Express.js powers the server-side logic, facilitating robust data processing and retrieval. Plotly.js is integrated for dynamic data visualization, enhancing the user interface. The application's architecture is built for scalability and efficiency, with a Node.js server orchestrating interactions.

Operations

FISCALC operates seamlessly to provide users with accurate and insightful retirement planning assistance. Here is a step-by-step guide on how to use the application:

1. **Accessing FISCALC:**

   - Open your preferred web browser and navigate to the FISCALC website.

   - Upon reaching the homepage, you will be presented with a user-friendly interface prompting you to begin your retirement planning journey.

2. **Inputting Financial Details:**

   - Start by entering your current financial information, including your current expenses, age, current savings, and desired retirement age.

   - Ensure that you input accurate and up-to-date financial data to receive precise retirement savings estimations.

3. **Calculating Retirement Savings:**

   - Once you have entered your financial details, click on the "Calculate" button to initiate the retirement savings estimation process.

   - FISCALC will utilize advanced algorithms to analyse your inputs and calculate the estimated amount needed for retirement.

4. **Reviewing Results:**

   - After the calculation process is complete, FISCALC will present you with a detailed breakdown of your retirement savings goals.

   - Review the results carefully, including the estimated amount needed at retirement, adjusted for inflation and life expectancy.

5. **Exploring Data Visualization:**

   - FISCALC utilizes interactive charts and graphs to visually represent your retirement savings projections.

   - Explore these visualizations to gain a deeper understanding of your financial plan and make informed decisions.

By following these instructions, you can effectively utilize FISCALC to plan for a secure and comfortable retirement with confidence and precision.

Testing Procedures and System Testing

1. **Unit Testing:**

   - Individual components of FISCALC, including algorithms for expense estimation, inflation adjustment, and life expectancy consideration, undergo rigorous unit testing.

   - Test cases are developed to verify the functionality and accuracy of each component, ensuring that they perform as expected in isolation.

2. **Integration Testing:**

   - After successful unit testing, integration testing is conducted to assess the interaction and compatibility of different modules within FISCALC.

   - Test scenarios are created to simulate user interactions and data flow through the entire system, verifying seamless integration and proper communication between components.

3. **User Testing:**

   - User testing is performed to evaluate the usability and effectiveness of FISCALC from an end user perspective.

   - Real users are invited to interact with the application and provide feedback on the user interface, data input process, and overall user experience.

   - Test results and user feedback are carefully analysed to identify areas for improvement and refinement.

Calibration Procedures

1. **Inflation Adjustment:**

   - FISCALC incorporates historical inflation rates to adjust retirement savings estimates for future inflation.

   - Calibration involves validating the accuracy of inflation adjustments by comparing projected inflation rates with historical data and economic forecasts.

   - Any discrepancies are addressed through adjustments to the inflation adjustment algorithm to ensure precise inflation calculations.

2. **Life Expectancy Consideration:**

- Life expectancy data used in FISCALC is calibrated to reflect demographic trends and health indicators accurately.

- Calibration involves analysing demographic data, mortality rates, and life expectancy projections to ensure the accuracy of retirement duration estimations.

- Any deviations from expected life expectancy trends are addressed through adjustments to the life expectancy consideration algorithm to enhance precision.

3. **Data Validation:**

- Data validation techniques are employed to ensure the accuracy and reliability of input data used in FISCALC.

- Calibration involves validating input data against external sources and benchmarks to identify any inconsistencies or errors.

- Any discrepancies are addressed through data cleaning and validation procedures to ensure precise data inputs for accurate retirement savings estimations.

By following systematic testing and calibration procedures, FISCALC is fine-tuned to deliver precise and reliable retirement planning assistance, empowering users to make informed financial decisions with confidence.

FUTURE SCOPE

The development of FISCALC marks the beginning of a journey towards continual improvement and expansion. Here are some avenues for future development and enhancement:

1. **Advanced Financial Models**: Integrate more sophisticated financial models and algorithms to provide users with comprehensive retirement planning insights, including investment strategies, asset allocation, and risk management.

2. **Enhanced Data Visualization:** Further enhance data visualization capabilities by integrating advanced charting libraries and interactive visualization tools to offer users deeper insights into their retirement savings projections.

3. **Personalized Recommendations:** Implement machine learning algorithms to analyse user data and provide personalized recommendations tailored to individual financial goals, risk tolerances, and life circumstances.

4. **Integration with Financial Services:** Explore partnerships with financial institutions and retirement planning advisors to integrate additional features such as investment tracking, retirement account management, and financial advisory services.

5. **Multi-Language Support:** Expand language support to cater to a global audience, providing localization options and translation services to make FISCALC accessible to users worldwide.

6. **Integration with External Data Sources:** Integrate with external data sources such as government statistics, economic indicators, and financial market data to provide users with real-time updates and insights into retirement planning trends and developments.

7. **Social Sharing and Community Features:** Implement social sharing functionalities and community features to allow users to share their retirement goals, achievements, and experiences with friends, family, and peers, fostering a sense of community and support.

8. **Continuous Optimization:** Continuously optimize and refine FISCALC based on user feedback, technological advancements, and industry best practices to ensure that it remains a cutting-edge and indispensable tool for retirement planning.

The future scope of FISCALC is expansive, with endless possibilities for innovation and growth. By embracing these opportunities and remaining committed to delivering value to users, FISCALC will continue to evolve as a leading retirement planning solution in the ever-changing landscape of personal finance.

CONCLUSION

Upon thorough evaluation of FISCALC, it is evident that while the product has addressed some aspects of the problem statement effectively, it falls short in fully solving the company's objectives. Here is a candid assessment of the product's performance and its alignment with the original criteria:

Effectiveness

FISCALC has successfully provided users with accurate estimations of retirement savings based on current expenses, inflation rates, and life expectancy which was the primarily target. The application's user-friendly interface is extremely beneficial for the new users and the option to change themes between dark and light add to a better user interface.

Weaknesses and Limitations

The product lacks integration with external financial accounts, limiting its ability to provide real-time financial insights and personalized recommendations. Moreover, the complexity of fine-tuning algorithms for inflation adjustment and life expectancy consideration posed unexpected challenges during development.

Difficulties Faced

Certain aspects of the original problem proved to be more challenging than anticipated. Balancing the accuracy of calculations with the simplicity of the user interface design required extensive iteration and testing. Complex calculations and designing algorithms to take into consideration inflation rate, interest rate also proved to be more difficult than initially thought.

Unmet Hopes

Despite initial hopes for seamless integration with financial institutions and comprehensive market coverage, FISCALC fell short in fully realizing these aspirations. The limitations in data integration and market reach hindered the product's ability to deliver a holistic solution to users' retirement planning needs. We were also unable to take into account financial investments of the user and the potential returns in the future.

REFERENCES

- "Personal Finance for Dummies" by Eric Tyson - https://www.dummies.com/personal-finance/

- "The Intelligent Investor" by Benjamin Graham - https://www.harpercollins.com/products/the-intelligent-investor-benjamin-graham

- "Rich Dad Poor Dad" by Robert T. Kiyosaki - https://www.richdad.com/

- "The Total Money Makeover" by Dave Ramsey - https://www.daveramsey.com/store/books/the-total-money-makeover

- "Investopedia: Your Guide to Financial Education" - https://www.investopedia.com/

- "The Balance: Make Money Personal" - https://www.thebalance.com/

- "NerdWallet: Personal Finance Advice and Tools" - https://www.nerdwallet.com/

- "Financial Planning Association" - https://www.plannersearch.org/

- "React.js Documentation"- Retrieved from React.js Website

- "Express.js Documentation"- Retrieved from Express.js Website