# Economic Analysis System based on React 18 and TypeScript

Group Members：

24129061 Tiago GUTERRES RODRIGUES

24121434 彭超

24129097 Damien BIREMBAUT

24129057 Alan Piedbois

## 一、Technical Architecture

This system is built based on today's most advanced front-end and full-stack development framework, aiming to provide an efficient, scalable, and easy-to-maintain user interface and data-driven analysis environment.

* Next.js 14 and React 18: Next.js is a full-stack framework for building high-performance web applications. It supports static generation and server-side rendering. It works with the Concurrent Rendering and Suspense features of React 18 to implement data preloading, lazy loading components, and automatic concurrent processing. This keeps the page responsive and stable in data-intensive economic analysis scenarios.
* TypeScript: The entire system is written in TypeScript, which provides strict type checking, interface definition, generic support, and code completion functions, which helps to find potential errors in advance during collaborative development and improve code quality and readability. Its static type mechanism also greatly enhances the IDE's smart prompt capabilities and reduces maintenance costs.
* Tailwind CSS: Tailwind is a CSS tool framework that prioritizes practicality. Developers can quickly implement complex responsive layouts and UI styles by combining class names. Compared with traditional CSS or CSS-in-JS solutions, Tailwind provides higher development efficiency and a more unified design language. It also supports Dark Mode and custom theme extensions, making the entire system interface consistent across different devices and themes.
* Componentized architecture and module layered design: Based on React's componentization concept, the system UI is divided into functional components of reasonable granularity (such as input forms, chart panels, pop-up window prompts, etc.). Each component has independent state management and life cycle control, which is easy to reuse and test. The page layer automatically associates the entry points of each module through the routing mechanism of Next.js, and the service layer encapsulates data requests and algorithm logic to ensure clear responsibilities between the front and back ends.
* State management and data-driven interaction: Use React Context and Hooks (such as useReducer and useContext) to implement internal state sharing and responsive interaction in modules, and use data request libraries such as SWR or React Query to manage remote data caching, retry, pre-fetching and other logic to ensure real-time feedback of user operations and data synchronization.

The system is divided into four functional modules: cost estimation, budget and financial analysis, risk management, and resource allocation.

## 二、Cost Estimating Module

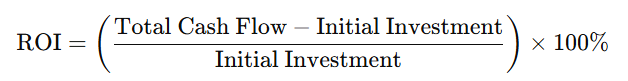
This module combines the COCOMO model with the function point analysis method to accurately evaluate the human workload, time duration and required team size of project development. In the COCOMO model, the system automatically calls the corresponding empirical coefficients (such as a, b, c, d) based on the KLOC (thousands of lines of code) entered by the user and the preset project type (organic, semi-separated, embedded), and calculates the estimated workload through formula (1), and further derives the project duration and required development team size. In order to enhance the accuracy of the estimate, the system also integrates the function point analysis algorithm. Users can define system components (such as external input/output, internal files, etc.) and their complexity levels through a graphical interface. The system automatically calculates the unadjusted function points (UFP) and then combines the complexity adjustment factor (VAF) to obtain the total function point value.

(1)

Subsequently, function points are converted to LOC based on the average productivity of different languages, thus providing a more reliable input for the COCOMO model. Technically, the system uses TypeScript to write dynamic estimation logic, and uses Chart.js to visualize the changing trends of LOC, workload, duration, and team size to achieve result visualization.

## 三、Budget and Financial Analysis Module

The budget analysis module integrates classic investment decision analysis models, including ROI (return on investment), NPV (net present value), IRR (internal rate of return) and payback period analysis. The system allows users to enter the initial investment amount, expected annual cash flow and discount rate, and calculate the rate of return using formula (2); the NPV formula (3) is used to evaluate the project net present value to take into account the time value of money.

(2)

(3)

In terms of internal rate of return calculation, the system implements the Newton-Raphson numerical iteration algorithm, which automatically approximates the discount rate value that makes NPV zero through derivative approximation and multiple rounds of iteration. The payback period is determined by calculating the time point when the cumulative cash flow value first reaches or exceeds the initial investment, and the system automatically outputs the critical year and the break-even chart. In the front-end implementation, the system binds user input through React state management, and uses the ECharts library to dynamically present cash flow trends, payback period intervals, and net present value curves to help users quickly and intuitively understand the financial health of the project.

## 四、Risk Management Module

### 4.1 Sensitivity Analysis

The sensitivity analysis module uses variable perturbation simulation to evaluate the impact of key input parameters of the project (such as labor costs, development cycle, KLOC, etc.) on economic results. The system supports users to set floating ranges for different parameters and generate three scenario models: pessimistic, neutral and optimistic scenarios. The corresponding ROI, NPV and IRR values will be recalculated in each scenario to evaluate the fluctuation range and trend of these indicators. In terms of technical implementation, the front-end uses TypeScript-based responsive data binding and D3.js to generate change trend charts. The back-end nests multiple rounds of calculations and automatically executes the cost model and financial model iteration process. During the simulation process, the parameter set is dynamically replaced, and sensitivity simulation is performed in the form of linear changes and step perturbations. The system finally outputs the comparison curve and relative influence ranking of the indicators in the form of interactive charts, thereby helping users identify the most influential variables and improving the controllability and foresight of project management decisions.

### 4.2 Monte Carlo Simulation

The Monte Carlo simulation module uses statistical modeling methods to systematically evaluate the uncertainty of input parameters. Users can set the distribution type (supports normal distribution, uniform distribution, etc.) and parameters (such as expected value, standard deviation, minimum and maximum range) for any variable, and the system will automatically perform thousands of rounds of random sampling simulations. In each round of simulation, a set of random samples is extracted from each distribution, and the cost estimates and financial results (ROI, NPV, IRR) are recalculated, and finally the distribution images of these indicators are generated, such as probability density function (PDF), cumulative distribution function (CDF) and confidence interval graph. In terms of implementation, the front end uses Web Workers to implement parallel computing to avoid interface blocking, and Chart.js is used to render probability histograms and curve graphs; the back end uses a pseudo-random number generator with parameter replacement and batch execution mechanism to improve simulation efficiency and reliability. This feature provides projects with visual insights into risk exposure, failure probability, and worst-case expectations, helping managers make robust decisions.

## 五、Resource Configuration Module

The resource allocation module aims to solve the optimal matching problem between personnel, skills and tasks in order to improve resource utilization and control development costs. Users can define the priority of project tasks, required skill tags and estimated working hours in the interface, and enter the skill sets and hourly costs of team members. The system builds a task-personnel matching matrix based on the greedy algorithm or the Hungarian algorithm to maximize skill matching while minimizing overall resource costs. For task conflicts or tasks with higher priorities, the system applies linear programming methods to optimize task sorting and member scheduling to ensure that key tasks are completed first. The front-end interactive design uses a drag-and-drop resource allocation interface, combined with responsive charts to display each member's current working hour utilization, skill coverage and task progress; the back-end scheduling logic is encapsulated as a reusable strategy module that supports parameterized calls and result backtracking. Through this module, users can clearly control the overall situation of human resource allocation and quickly respond to personnel adjustments and changes in project rhythm.

## 六、Front-end Interactive Component Design

1. Dashboard Home Page

* Overview charts (line chart, pie chart, bar chart).
* Quick access to four core modules.

1. Cost Estimation Module Page

* COCOMO and Function Point Entry Form and Calculator.
* KLOC and workload trend chart.

1. Budget and Finance Module Page

* Investment input and result indicator chart (ROI, NPV, IRR).
* Cash flow distribution line chart and break-even point display.

1. Risk Management Module Page

* Scenario simulation parameter slider and result distribution chart.
* Monte Carlo simulation interactive panel and probability plots.

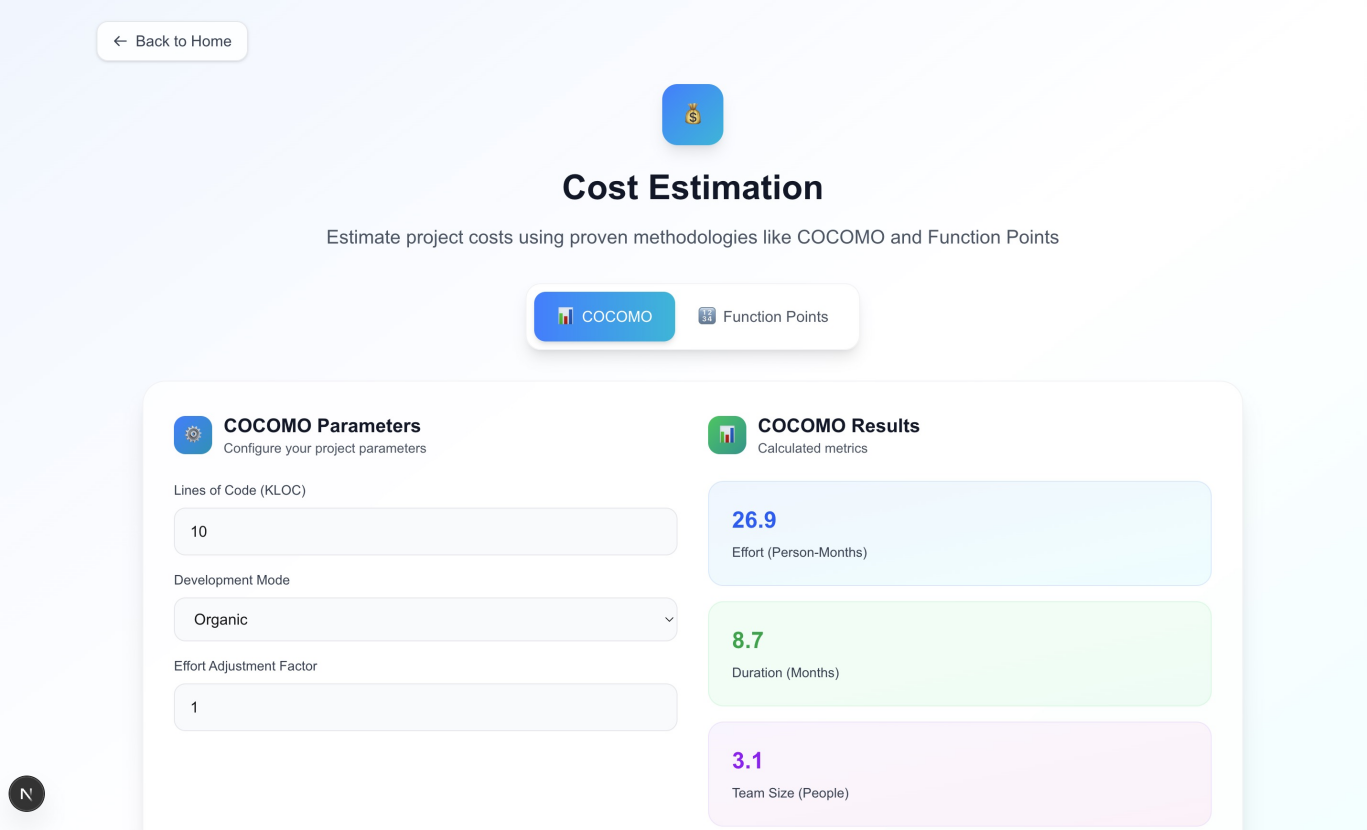
1. Resource Configuration Module Page

* Member management interface (add skills, set costs).
* Automatic allocation result display and manual adjustment support.

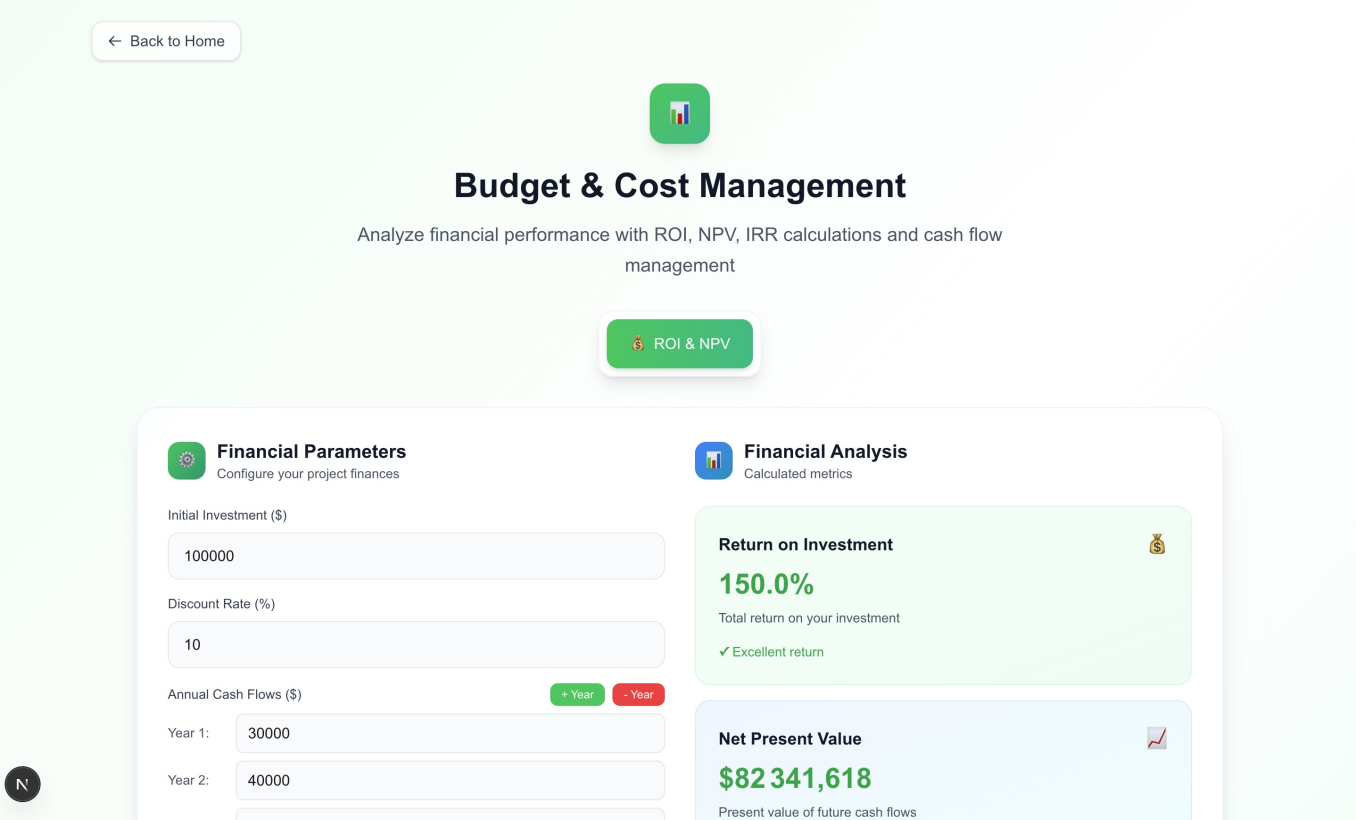
## 七、Conclusion

This system integrates modern front-end technology and economic modeling algorithms, has good user experience and calculation accuracy, and is suitable for project management and budget control scenarios. In the future, it can be expanded to intelligent modules such as AI-assisted prediction and automatic resource scheduling.

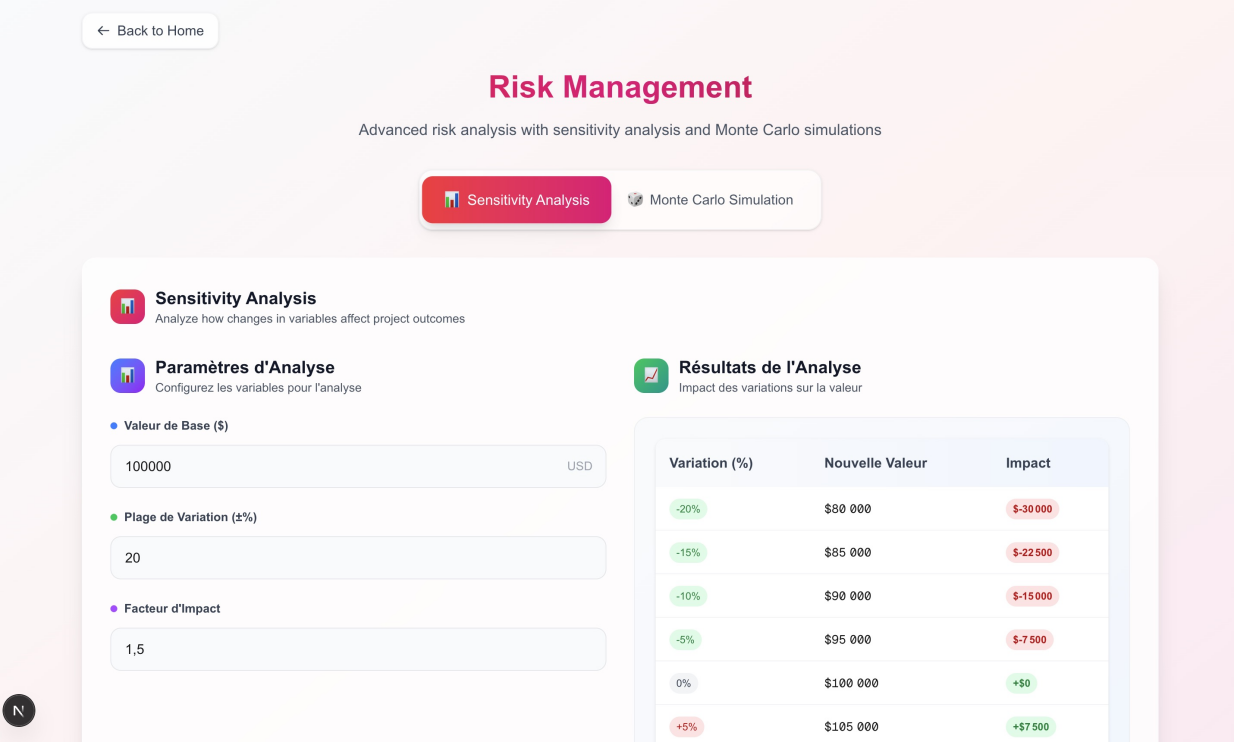
## System Screenshots



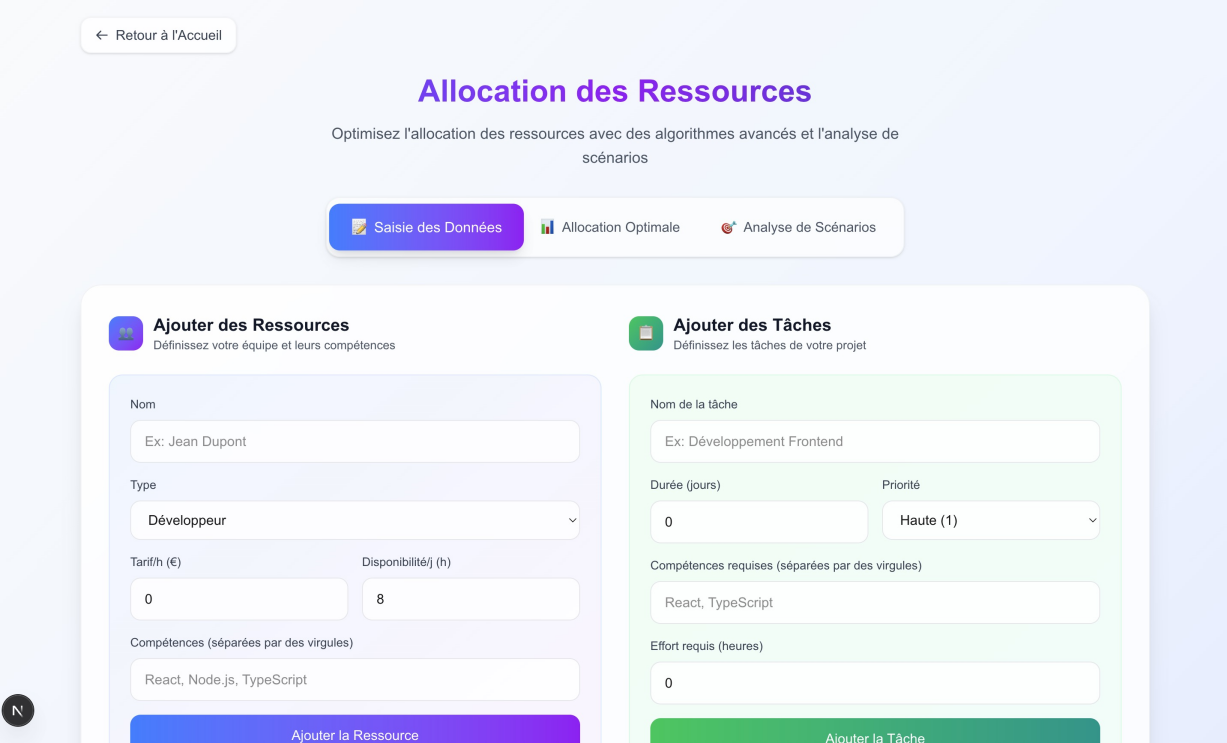
**Figure.1 Cost Estimation Module**



**Figure.2 Budget & Cost Management**



**Figure.3 Risk Management**



**Figure.4 Allocation des Ressource**