# LIVE PROJECT-WATERFALL

**1) Business case document template**

# **ANS)**

# Business Case Document: Cotrack - Insurance Policy Management Software

## ➢ Why is this project initiated?

The Cotrack project is initiated to address inefficiencies in insurance policy management for financial institutions. Current manual processes lead to delays, errors, and compliance risks. Cotrack aims to centralize policy data, enhance communication, and improve operational efficiency.

## ➢ What are the current problems?

* Financial institutions face challenges such as:
* Inefficient manual processes
* Data inaccuracies and missing information
* Difficulty in handling large policy volumes
* Poor customer communication and tracking
* Compliance risks and high operational costs
* Fragmented data storage and limited scalability

## ➢ With this project how many problems could be solved?

Cotrack will resolve key challenges by:

* Automating policy management workflows
* Reducing data errors through validation mechanisms
* Providing real-time updates and tracking
* Enhancing customer-agent communication
* Ensuring compliance with industry standards
* Improving overall efficiency and scalability

## ➢ What are the resources required?

To successfully implement Cotrack, the following resources are needed:

**Development Team:** Software engineers, UI/UX designers

**Business Analysts:** Requirements gathering and validation

**Testing Team:** Quality assurance and debugging

**Infrastructure:** Servers, databases, cloud services

**Training & Support Personnel:** User onboarding and assistance

## ➢ How much organizational change is required to adopt this technology?

Moderate organizational change is required. Training programs will be provided to ensure seamless adoption by analysts and agents. Resistance to new systems will be mitigated through user-friendly design and phased implementation.

## ➢ Time frame to recover ROI?

The estimated time frame to recover ROI is within **12-18 months** post-implementation. The efficiency gains, error reduction, and improved compliance will lead to cost savings and increased productivity.

## ➢ How to identify Stakeholders?

A RASCI matrix helps in identifying and defining stakeholder roles:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Stakeholder** | **Responsible (R)** | **Accountable (A)** | **Supportive (S)** | **Consulted (C)** | **Informed (I)** |
| Financial Institutions |  | YES |  | YES | YES |
| Insurance Companies |  | YES |  | YES | YES |
| Agents & Brokers | YES |  | YES |  | YES |
| Policyholders |  |  |  | YES | YES |
| IT & Compliance Teams | YES |  | YES | YES | YES |
| Development Team | YES |  |  | YES | YES |
| Business Analysts | YES |  |  | YES | YES |
| Testing Team | YES |  |  | YES | YES |
| Training & Support Staff | YES |  |  | YES | YES |

**Responsible (R):** Those who perform the tasks and execute the work.

**Accountable (A):** The final decision-makers and owners of the project.

**Supportive (S):** Those who assist in project execution.

**Consulted (C):** Those whose input is required for key decisions.

**Informed (I):** Those who need updates but do not directly contribute.

## Conclusion

Cotrack is designed to significantly enhance insurance policy management, leading to operational efficiency, compliance, and cost savings. The structured implementation and stakeholder involvement will ensure a successful rollout by June 2025.

**2) Document 2: BA Strategy**

# ANS) Business Analyst Approach Strategy for Cotrack - Insurance Policy Management Software

As a **Business Analyst** (BA), my primary role would be to ensure that we define clear business goals, understand user needs, map out processes, and create a solution that not only meets current challenges but is scalable for the future. Here’s my approach to this project:

### ****1. Requirements Gathering and Analysis (Phase 1)****

As a **Business Analyst**, the first and most critical step is to gather and analyze the **business requirements** thoroughly. I would:

**Engage with stakeholders:** This includes interviewing insurance company representatives, policy agents, customers, and compliance officers to understand their pain points and needs. For example, understanding the issues like slow manual work, errors in data, and poor communication will help me map the software features accurately.

**Document the requirements:** I would create a detailed **Business Requirements Document (BRD)** to capture both **functional** and **non-functional requirements**:

* + **Functional Requirements** would cover key functionalities such as automated policy creation, real-time updates, customer-agent communication, and compliance tracking.
  + **Non-Functional Requirements** would cover performance metrics like system scalability, security standards (for data protection), and compliance with industry regulations (e.g., GDPR).

**Approval from stakeholders:** Once the requirements are gathered and documented, I would review them with key stakeholders (insurance companies, agents, etc.) and get formal **approval**. This ensures that we are aligned on the project’s scope before moving on to the design phase.

### ****2. System and Software Design (Phase 2)****

Once the requirements are finalized, my role as the BA in the **design phase** is to work with the technical team to ensure that the solution being designed fully aligns with the business needs.

**High-Level System Design:** I would collaborate with system architects and developers to ensure that the design addresses the requirements from a business perspective. For example, I would make sure that the design accounts for automated policy management, error prevention features, and an intuitive user interface.

**Detailed Design Specifications:** I would ensure that we have detailed **design documents** for:

* + **User interfaces (UI):** Clear designs for the policy management system (including dashboards for agents and clients).
  + **Process flows:** Visual diagrams showing how tasks like policy updates or claims processing should work from a user perspective.

**Integration Design:** If CoTrack needs to integrate with other systems (e.g., CRM, payment systems), I would make sure these integrations are well-defined in the design.

**Approval from stakeholders:** After the design is created, I would ensure the stakeholders review and **approve** the design before moving to development.

### ****3. Development (Phase 3)****

During the **development phase**, my role as a BA is to ensure the development team is clear on the requirements and designs, and that everything is being built according to plan.

**Support the development team:** I would be available to answer any questions the developers have about the requirements or design. For example, if they need clarification on how policy renewal should be automated, I’d provide that.

**Track progress:** I would monitor the progress of development to ensure that the features are being implemented as per the approved design. If there are any scope changes or concerns, I would raise them with the project manager and stakeholders immediately.

**Communication with stakeholders:** Throughout the development process, I would provide regular updates to stakeholders about the progress and any potential risks or delays.

### ****4. Testing (Phase 4)****

Once the system is developed, the **testing phase** is crucial to ensure the solution works as intended. My approach as a BA would be to:

**Test Planning:** I would work with the QA team to define test cases based on the **Business Requirements Document (BRD)**. These test cases would include:

* + Functional testing to verify policy automation.
  + Integration testing to ensure the system communicates correctly with other platforms.
  + Security testing to validate that customer and policy data is protected.

**User Acceptance Testing (UAT):** I would coordinate UAT with stakeholders, such as agents and insurance representatives. I would ensure that the system works as expected in real-world scenarios. For example, testing the policy renewal process or how errors are handled by the system.

**Defect Tracking and Resolution:** During testing, I would help track any issues and ensure they are documented and prioritized for resolution. I would also work with the developers to ensure that critical defects are addressed before moving to the next phase.

### ****5. Deployment (Phase 5)****

Once the system is fully tested and validated, it’s time to deploy it. My approach would involve ensuring the deployment goes smoothly:

**Deployment Planning:** I would ensure that a detailed **deployment plan** is in place, including:

* + Migration of data (if any existing data needs to be transferred).
  + Steps for transitioning the system to the live environment.
  + A rollback strategy in case any issues arise during deployment.

**Support during Go-Live:** I would ensure that the deployment team is supported, and I’d be available for any last-minute clarification needed during the **go-live** process.

**User Communication:** I would ensure that users (insurance agents and customers) are informed and trained on how to use the new system.

### ****6. Maintenance and Support (Phase 6)****

After the system goes live, I would be involved in ensuring that the system continues to meet business needs over time:

**Post-Go-Live Support:** I would ensure that support teams are available to address any issues that arise post-deployment. This may involve troubleshooting or handling minor defects.

**Collecting User Feedback:** I would gather feedback from users, including agents and customers, to see how the system is performing and whether there are any areas for improvement.

**Ongoing Maintenance:** I would work with the development team to address any necessary **updates** or **patches** to keep the system compliant and functioning well as the business needs evolve.

### ****Key Points to Focus on:****

**Clear Documentation:** Waterfall relies heavily on documentation, so I would ensure that every phase (from requirements to deployment) is thoroughly documented, providing a clear record of decisions and approvals for stakeholders.

**Stakeholder Communication:** Regular communication with stakeholders would be essential throughout the project to ensure that the system meets their needs and expectations.

**No Scope Creep:** In the Waterfall model, the scope is locked after the requirements phase. As a BA, I would make sure that no significant changes are made without formal approval from stakeholders, ensuring that the project stays on track.

**Attention to Detail:** Given the linear nature of the Waterfall model, it’s important to catch potential issues in the early stages (requirements gathering and design). I would work meticulously to ensure nothing is overlooked.

**3) Document 3- Functional Specifications**

### ANS)

### ****Functional Specifications Document (FSD)****

#### ****Project Overview****

**Project Name:**  
CoTrack – Insurance Policy Management Software

**Customer Name:**  
SWBC

**Project Version:**  
1.0

**Project Sponsor:**  
SWBC

**Project Manager:**  
Cynthia

**Project Initiation Date:**  
15/02/2025

#### ****Functional Requirement****

| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| --- | --- | --- | --- |
| **FR0001** | Login | The user should be able to login to the application to perform inventory operations. | 10 |
| **FR0002** | Policy Creation | The system should allow users to create a new insurance policy with necessary details such as customer information, coverage type, and policy period. | 10 |
| **FR0003** | Policy Update | Users should be able to update existing policy information, such as coverage changes or policy status. | 9 |
| **FR0004** | Policy Renewal | The system should automatically notify users when a policy is due for renewal and provide an option for automatic renewal. | 8 |
| **FR0005** | Error Handling | The system must handle errors gracefully by providing clear error messages and allowing users to correct them. | 10 |
| **FR0006** | User Roles Management | The system should allow administrators to define different user roles (e.g., agent, supervisor) and set permissions accordingly. | 7 |
| **FR0007** | Customer Communication | The system should enable communication between agents and customers via email or in-app messaging. | 6 |
| **FR0008** | Compliance Check | The system must verify that all policies meet regulatory and compliance standards before approval. | 10 |
| **FR0009** | Data Reporting | The system should provide real-time reports on policy status, renewals, and claims. | 8 |

#### ****Detailed Descriptions of Functional Requirements****

**FR0001 - Login:**

* + **Description:** The user must be able to securely log in to the application by entering a username and password. Once logged in, the user will have access to the features they are authorized to use (e.g., creating policies, viewing reports, etc.).
  + **Acceptance Criteria:**
    - Users should see a login screen with fields for username and password.
    - The system should validate user credentials and log in successfully if valid.
    - If invalid credentials are entered, the user should see an error message.
    - The login process should be secure and use encryption for passwords.

**FR0002 - Policy Creation:**

* + **Description:** The system should allow agents to create new insurance policies. The user should be prompted to enter information such as customer details, policy type, coverage limits, and policy term. Upon successful creation, the policy should be saved in the database.
  + **Acceptance Criteria:**
    - Fields for customer name, address, policy type, coverage amount, and policy duration should be available.
    - The system should validate that all required fields are filled before allowing the policy to be created.
    - The policy should be saved in the system’s database and visible to the user.

**FR0003 - Policy Update:**

* + **Description:** Users must be able to edit existing policies. This includes updating customer details, modifying coverage, or changing policy status.
  + **Acceptance Criteria:**
    - The system should provide an interface to view and update policy details.
    - Changes should be tracked with timestamps and user identification.
    - The updated policy should reflect the changes in the system immediately after saving.

**FR0004 - Policy Renewal:**

* + **Description:** The system should automatically send reminders for policy renewal and allow customers or agents to renew the policy with just a few clicks.
  + **Acceptance Criteria:**
    - The system should send email notifications to the agent and customer 30 days before the policy expires.
    - The user should be able to renew the policy directly within the system.

#### ****Prioritization of Requirements****

In this section, I will define the **priority** of each functional requirement. It helps determine the order in which the features will be implemented.

* **Priority 10**: Must-have features that are critical for the system’s operation (e.g., login, policy creation, compliance checks).
* **Priority 9**: Important features but can be added after the core features are implemented (e.g., policy update, reporting).
* **Priority 8**: Features that improve user experience but are not essential for system functionality (e.g., customer communication, reporting).
* **Priority 7 and below**: Optional features that can be considered after the core functionality is in place (e.g., role management).

**Document4)Requirement Traceability Matrix**

ANS)

The RTM helps to ensure that each requirement is addressed in the design, development, testing, and final delivery stages. It acts as a reference tool to validate that the system is meeting the specified requirements and ensures that no requirement is overlooked.

### ****Approach to Creating the Requirement Traceability Matrix (RTM)****

The **RTM** typically includes the following columns:

1. **Req ID**: Unique identifier for each requirement.
2. **Req Name**: A short name or title for the requirement.
3. **Req Description**: A detailed description of the requirement.
4. **Design (D1, D2, etc.)**: Indicates which design document or section addresses the requirement.
5. **T1, T2 (Development/Code Implementation)**: Indicates which development task or component corresponds to the requirement.
6. **UAT (User Acceptance Testing)**: Indicates whether the requirement has been validated during the UAT phase (testing by stakeholders/end-users).

### ****Sample Requirement Traceability Matrix (RTM)**** for the ****CoTrack Project****:

| **Req ID** | **Req Name** | **Req Description** | **Design** | **D1** | **T1** | **D2** | **T2** | **UAT** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FR0001** | Login | User must be able to log in to the application to access the system and perform inventory operations. | Yes | Yes | Yes | No | Yes | YES |
| **FR0002** | Policy Creation | Users should be able to create insurance policies with customer data and coverage details. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0003** | Policy Update | Users should be able to update existing policies, including coverage and details. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0004** | Policy Renewal | The system should automatically send renewal reminders and enable policy renewal. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0005** | Error Handling | The system must handle errors with clear error messages and provide corrective options. | Yes | Yes | Yes | No | Yes | YES |
| **FR0006** | User Roles | The system should allow administrators to define and manage user roles and permissions. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0007** | Customer Communication | Users must be able to send communication between agents and customers through email or messaging features. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0008** | Compliance Check | The system must check compliance for each policy to ensure it adheres to regulatory standards before approval. | Yes | Yes | Yes | Yes | Yes | YES |
| **FR0009** | Data Reporting | The system must generate reports for policy status, renewals, and claims. | Yes | Yes | Yes | Yes | Yes | YES |

### ****Explanation of Columns in RTM****

**Req ID**:

This is the **unique identifier** for each requirement, such as **FR0001**, **FR0002**, etc. It’s important to use consistent and systematic naming to make it easy to track each requirement.

**Req Name**:

This is the **short name** or **title** of the requirement. It provides a quick reference to the requirement’s purpose or functionality (e.g., Login, Policy Creation, etc.).

**Req Description**:

The detailed **explanation** of what the requirement is and how it fits into the system. This clarifies what is expected from the software solution.

**Design (D1, D2, etc.)**:

This column tracks the **design documentation** or **design elements** that correspond to the requirement.

"Yes" indicates that the requirement is addressed in a particular design document.

If marked "Pending," it means the design work for that requirement is not yet completed.

**Development (T1, T2)**:

These columns correspond to **specific development tasks** or **components** that address the requirement.

"Yes" indicates the development task/component addresses the requirement.

If marked "No," it indicates that the development task does not address this requirement.

**UAT (User Acceptance Testing)**:

This column is used to confirm whether the requirement has been included in the **User Acceptance Testing (UAT)** phase.

"YES" means the requirement has been tested by the end-users and stakeholders during UAT.

If marked "Pending" or "No," it indicates that the requirement has not yet been tested during UAT or the testing is yet to be conducted.

**Document 5) - BRD Template**

* ANS)

## ****Business Requirements Document (BRD) for CoTrack Project****

### ****1. Document Control****

* **Version:** 1.0
* **Prepared by: SWATI RAI**
* **Date:-04/02/2025**
* **Approved by:** SWBC
* **Version History:**

| **Version** | **Date** | **Description** | **Author** |
| --- | --- | --- | --- |
| 1.0 | 04/02/2025 | 04/02/2025 | SWATI |

### ****2. Executive Summary****

The **CoTrack** project is being initiated to develop an **Insurance Policy Management Software** that will automate the management of policies, improve data accuracy, reduce operational costs, and streamline communication between customers and agents. The objective is to modernize the current system, resolve inefficiencies, and meet compliance requirements while improving customer satisfaction.

**Project Goals:**

* Automate policy management to eliminate slow and manual work.
* Reduce errors and data inconsistencies.
* Improve communication between agents and customers.
* Ensure compliance with regulations and industry standards.
* Enable scalability for future growth.

### ****3. Business Objectives****

The **CoTrack** system must address several key business objectives:

1. **Increase Efficiency**: Automate policy creation, renewal, and updates to reduce manual errors and the time spent on administrative tasks.
2. **Enhance Data Accuracy**: Implement robust validation and error-checking mechanisms to ensure data consistency.
3. **Improve Customer Satisfaction**: Provide real-time tracking and updates for policyholders, allowing for better customer service and communication.
4. **Ensure Compliance**: Adhere to legal and industry regulations, ensuring that policies meet all necessary compliance standards.
5. **Support Growth**: Implement a scalable solution that can handle an increasing number of policies, users, and customers as the business grows.

### ****4. Scope****

#### ****In-Scope****

The following activities are within the scope of the CoTrack project:

* Development of a web-based application for policy management.
* Features for policy creation, policy update, and policy renewal.
* Automatic communication system between agents and customers (email notifications, in-app messages).
* User role management for different access levels (e.g., agents, administrators).
* Integration with existing CRM and payment systems.
* Reporting and dashboard for policy status, renewals, and claims.
* Compliance tracking and validation.
* Support for mobile access (optional for future release).

#### ****Out-of-Scope****

The following items are excluded from the scope of the project:

* Development of mobile applications (for this phase).
* Integration with external payment gateways or banking systems (unless specified by the client).
* Development of non-policy-related functionalities (e.g., claim management, reporting for other services).
* Data migration from legacy systems (if needed, can be considered in a later phase).

### ****5. Stakeholders****

The key stakeholders for the **CoTrack Project** include:

| **Role** | **Name** | **Responsibility** |
| --- | --- | --- |
| **Project Sponsor** | SWBC | Overall project direction, funding, and approval. |
| **Project Manager** | CYNTHIA | Oversee project execution, timelines, and resources. |
| **Business Analyst** | SWATI | Requirements gathering, documentation, and ensuring project alignment with business needs. |
| **Product Owner** | NEIL | Provide ongoing business input and feedback. |
| **Development Team** | SONAL,ARBAAZ,AJAY,SONU | Design, develop, and deploy the system. |
| **QA/Test Team** | JENNIFER | Test the system, perform UAT, and ensure quality. |
| **Compliance Officer** | GAURAV | Ensure system complies with legal and regulatory standards. |
| **End-Users (Agents)** | NAVJIT,NAVED,DIVYA | Use the system for policy management and interact with customers. |
| **Customers (Policyholders)** | RUSHMORE,CARRINGTON AND STATE FARM CUSTOMERS | End-users interacting with the system for policy management and communication. |

### ****6. Functional Requirements****

The **functional requirements** define the features and functionality that the system must support. Below are some examples of functional requirements:

| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| --- | --- | --- | --- |
| **FR0001** | Login | The user must be able to log in securely to access the application for inventory operations. | 10 |
| **FR0002** | Policy Creation | The system should allow users to create a new insurance policy with necessary details such as customer information, coverage type, and policy period. | 10 |
| **FR0003** | Policy Update | Users should be able to update existing policy information, such as coverage changes or policy status. | 9 |
| **FR0004** | Policy Renewal | The system should automatically notify users when a policy is due for renewal and provide an option for automatic renewal. | 8 |
| **FR0005** | Error Handling | The system must handle errors gracefully by providing clear error messages and allowing users to correct them. | 10 |
| **FR0006** | User Roles Management | The system should allow administrators to define different user roles (e.g., agent, supervisor) and set permissions accordingly. | 7 |
| **FR0007** | Customer Communication | The system should enable communication between agents and customers via email or in-app messaging. | 6 |
| **FR0008** | Compliance Check | The system must verify that all policies meet regulatory and compliance standards before approval. | 10 |
| **FR0009** | Data Reporting | The system should provide real-time reports on policy status, renewals, and claims. | 8 |

### ****7. Non-Functional Requirements****

Non-functional requirements focus on how the system should perform rather than what it should do. Some of the **non-functional requirements** for **CoTrack** include:

1. **Performance:** The system should be able to handle a minimum of 1,000 concurrent users without significant degradation in response times.
2. **Scalability:** The system must be designed to accommodate growth in both data volume and user base, capable of supporting up to 10,000 users and handling an increased number of policies annually.
3. **Security:** The system should implement **role-based access control (RBAC)** to restrict access to sensitive data and operations based on the user's role.
4. **Availability:** The system should be available 99.9% of the time, with planned downtimes for maintenance communicated in advance.
5. **Compliance:** The system must comply with **GDPR**, **HIPAA**, and industry-specific regulations related to insurance and data protection.

### ****8. Assumptions****

* The project will be developed using a **web-based platform** (with a future mobile application considered later).
* Existing data will be cleaned and formatted before the new system is integrated.
* All required legal and compliance standards will be provided by the **Compliance Officer**.

### ****9. Constraints****

* The initial budget for the project is limited to $290,000 .
* The first version of the system must be delivered within 6 months to align with the insurance company's annual policy review cycle.
* Integration with legacy systems must be completed without disrupting the daily operations of the business.

### ****10. Risks****

1. **Data Migration Challenges**: There may be challenges in migrating data from the legacy system to the new system, which could delay the project.
2. **Compliance Complexity**: Frequent regulatory changes in the insurance industry could require rapid adjustments to the system.
3. **User Adoption**: Ensuring that end-users (agents and customers) are fully trained and comfortable with the new system could take longer than anticipated.

### ****11. Approval and Sign-Off****

This section confirms that the business requirements have been reviewed and approved by all necessary stakeholders. It serves as a formal agreement on what is expected from the project.

| **Name** | **Role** | **Signature** | **Date** |
| --- | --- | --- | --- |
| SWBC | Project Sponsor | - | - |
| SWBC | Product Owner | - | - |
| NELLY | Development Lead | - | - |

This document will be continuously reviewed and updated as needed to reflect any changes or new insights throughout the project's lifecycle.

**CONFIDENTIAL** | Cotrack - Insurance Policy Management Software