EY Catalyst Business Continuity Management (BCM) System

Source Code & API Documentation

Document Version: 1.0
Date: November 30, 2025
Organization: R.V. College of Engineering
Status: Development Stage

EXECUTIVE SUMMARY

This document provides the authoritative source code mapping, module-wise architecture, and API endpoint documentation for the **EY Catalyst Business Continuity Management (BCM) System**. It is designed to enable any engineer to:

1. **Clone and run the system locally** in 5 commands
2. **Understand module organization** and locate specific functionality
3. **Integrate with backend APIs** using real endpoint examples
4. **Extend or debug modules** independently with confidence
5. **Navigate the codebase** efficiently with clear file paths and component descriptions

All information in this document has been **verified against the current codebase** and reflects the actual implementation as of November 28, 2025.

TECH STACK SUMMARY

Layer	Technology	Purpose
Frontend	React + Vite	Modern UI framework with fast development
	Axios	HTTP client for API communication
	Chart.js / Recharts	Data visualization dashboards
Backend	FastAPI (Python)	High-performance REST API framework
	SQLAlchemy ORM	Database abstraction layer
	Pydantic	Request/response validation
Databases	PostgreSQL (Supabase)	Production database
	SQLite	Development database
	MongoDB	Document storage (procedures, uploads)
Authentication	JWT (python-jose)	Stateless token authentication

Layer	Technology	Purpose
	LDAP3	Active Directory integration
AI Integration	Groq API	LLM for content generation

QUICKSTART – GET RUNNING IN 5 COMMANDS

Backend Setup

```
git clone <REPO_URL>
cd BCM_Plan_Procedures_Crisis_Recovery/backend_brt
python -m venv venv
# source venv/bin/activate
# Windows: venv\Scripts\activate
pip install -r requirements.txt
cp .env.example .env
python main.py # Server runs on http://localhost:8000
```

Frontend Setup

```
cd ../EY-Catalyst-front-end
npm install
npm run dev # App runs on http://localhost:5173
```

Verify Installation:

Open http://localhost:5173 in browser

Navigate to http://localhost:8000/docs for API Swagger UI

Login with credentials from `.env` (ADMIN_USER, ADMIN_PASSWORD)

MODULE 1: AUTH & RBAC (Authentication & Authorization)

Purpose: Handles user authentication via Active Directory, JWT token generation, and role-based access control (RBAC). All users are synchronized from AD and assigned roles based on AD group membership.

Code Paths

Component	File Path
Frontend – Login Page	src/modules/auth/components/Login.jsx
Frontend – User Management	src/modules/admin/components/UsersManagement.jsx
Frontend – Module Approvals	src/modules/admin/components/ModuleApprovals.jsx
Backend – Auth Router	backend_brt/app/routers/auth_router.py
Backend – RBAC Router	backend_brt/app/routers/rbac_router.py
Backend – RBAC Models	backend_brt/app/models/rbac_models.py
Backend – RBAC Service	backend_brt/app/services/rbac_service.py

Key Components

Auth Router (auth_router.py):

Authenticates users against Active Directory using LDAP

Generates JWT tokens with user role and organization context

Implements token refresh mechanism for extended sessions

RBAC Router (rbac_router.py):

Manages Users, Roles, Permissions, Organizations, Departments,

Processes Provides hierarchical permission checks

Syncs users from AD on demand

API Endpoints

Method	Path	Purpose	Auth Required	Request	Response
POST	/auth/token	Login & get JWT token	No	username, password (form-data)	access_token, token_type, user_id, role
POST	/auth/refresh	Refresh expired JWT token	Yes	Refresh token	New access_token
GET	/auth/me	Get current authenticated user	Yes	None	User object (id, username, email, role, org_id)
GET	/rbac/users/	List all users	Yes (Admin)	None	Array of User objects
POST	/rbac/users/	Create new user	Yes (Admin)	UserCreate schema	Created User object
GET	/rbac/roles/	List all roles	Yes	None	Array of Role objects
POST	/rbac/roles/assign/	Assign role to user	Yes (Admin)	user_id, role_id	Assignment status
POST	/rbac/init	Initialize default RBAC	No	None	Initialization message

Example: Login Request/Response

Request:

```
POST /auth/token
Content-Type: application/x-www-form-urlencoded

username=Administrator&amp;password=password123
```

Response:

```
{
  "access_token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJBZG1pbmlzdHJhdG9yIiwiZ
  "token_type": "bearer",
  "user_id": 1,
  "username": "Administrator",
  "email": "admin@company.com",
  "role": "System Admin",
  "is_admin": true,
  "organization_id": 1
}
```

How to Test

```
# Test login
curl -X POST http://localhost:8000/auth/token \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -d "username=Administrator&amp;password=password123"

# Use token for subsequent requests
TOKEN="&lt;access_token_from_response&gt;"
curl -H "Authorization: Bearer $TOKEN" http://localhost:8000/auth/me
```

MODULE 2: ADMIN & DASHBOARD (System Administration)

Purpose: Provides System Administrators with tools to onboard organizations, synchronize users from Active Directory, manage licenses, and view system-wide metrics. Displays high-level statistics on BIA completion, critical processes, and system health.

Code Paths

Component	File Path
Frontend – Admin Dashboard	src/modules/admin/components/AdminDashboard.jsx
Frontend – Organizations Management	src/modules/admin/components/OrganizationsManagement.jsx
Frontend – License Management	src/modules/admin/components/LicenseManagement.jsx
Frontend – User Management	src/modules/admin/components/UsersManagement.jsx
Backend – Admin Router	backend_brt/app/routers/admin_router.py
Backend – BCM Router (Stats)	backend_brt/app/routers/bcm_router.py
Backend – Global Models	backend_brt/app/models/global_models.py

Key Components

Admin Router (admin_router.py):

Organization setup with automated AD OU and group creation

User sync from Active Directory

License expiry tracking and updates

BCM Router (Stats Module) (bcm_router.py):

Dashboard statistics aggregation (BIA completion rates, critical process

count) Department-level summaries

API Endpoints

Method	Path	Purpose	Auth Required
GET	/admin/users/	Fetch users from AD	Yes (Admin)
POST	/admin/organizations/setup	Create new org structure in AD & DB	Yes (Admin)
GET	/admin/organizations/	List all organizations	Yes (Admin)
PUT	/admin/organizations/{id}/license	Update license expiry	Yes (Admin)
GET	/bcm/dashboard/stats	Get system-wide statistics	Yes
GET	/bcm/departments/	Get departments with BIA stats	Yes

Example: Dashboard Stats Response

Request:

```
GET /bcm/dashboard/stats
Authorization: Bearer &lt;TOKEN&gt;
```

Response:

```
{
  "total_processes": 45,
  "completed_bia": 30,
  "pending_bia": 15,
  "critical_processes": 12,
  "total_departments": 5,
  "completion_rate": 67,
  "last_updated": "2025-11-27T10:30:00Z"
}
```

How to Test

```
# Setup organization
curl -X POST http://localhost:8000/admin/organizations/setup \
  -H "Authorization: Bearer &lt;TOKEN&gt;" \
  -H "Content-Type: application/json" \
  -d '{"organization_name":"Test Corp","client_head_email":"head@testcorp.com"}'

# Get dashboard stats
curl -H "Authorization: Bearer &lt;TOKEN&gt;" \
  http://localhost:8000/bcm/dashboard/stats
```

MODULE 3: BCM (Business Continuity Plans)

Purpose: Manages Business Continuity Management plans at organizational and departmental levels. Supports plan generation from templates, in-place editing, versioning, PDF export, and risk assessment procedure management.

Code Paths

Component	File Path
Frontend	src/modules/bcm/OrganizationBCMPlan.jsx
Frontend – Dept BCM Plan	src/modules/bcm/DepartmentalBCMPlan.jsx
Frontend – BCM Dashboard	src/modules/bcm/BCMDashboard.jsx
Backend – BCM Router	backend_brt/app/routers/bcm_router.py
Backend – Procedures Router	backend_brt/app/routers/procedures_router.py
Backend – BCM Models	backend_brt/app/models/bcm_models.py
Backend – BCM Service	backend_brt/app/services/bcm_plan_service.py

Key Components

BCM Router (bcm_router.py):

- Organization and departmental plan CRUD operations
- Plan seeding for demo/testing
- PDF generation and export

Procedures Router (procedures_router.py):

- Risk Assessment Procedure management
- Other SOP types (BIA, Crisis Communication, Training, Non-conformity)
- Version history tracking

API Endpoints

Method	Path	Purpose
GET	/bcm/organization-plan/{id}	Retrieve organization-level BCM plan
PUT	/bcm/organization-plan/{id}	Update organization-level plan
GET	/bcm/department-plan/{id}	Retrieve department-level BCM plan
PUT	/bcm/department-plan/{id}	Update department-level plan
POST	/bcm/seed-plans	Generate demo BCM plans
POST	/bcm/organization-plan/{id}/export-pdf	Export plan to PDF file
GET	/procedures/risk-assessment-procedure/{org_id}	Get Risk Assessment procedure

Method	Path	Purpose
POST	/procedures/risk-assessment-procedure/{org_id}	Create/update Risk Assessment

Example: BCM Plan Response

Request:

```
GET /bcm/organization-plan/org_123
Authorization: Bearer &lt;TOKEN&gt;
```

Response:


```
{
  "id": "plan_123",
  "organization_name": "Demo Organization",
  "plan_type": "organization_level",
  "plan_version": "1.0",
```

```
  "introduction": "This Business Continuity Plan defines the organization's strategy...",
  "scope": "All critical business processes",
  "governance": "BCM Steering Committee oversees...",
  "recovery_strategies": "Data backup, alternate site recovery...",
  "last_updated": "2025-11-27T10:00:00Z",
  "created_by": "Administrator"
}
```

MODULE 4: BUSINESS IMPACT ANALYSIS (BIA)

Purpose: Enables process owners to analyze impact of disruptions on business processes. Users define Recovery Time Objectives (RTO), Maximum Tolerable Period of Disruption (MTPD), and assess criticality using impact matrices.

Code Paths

Component	File Path
Frontend – BIA Form	src/modules/bia/components/BusinessImpactAnalysis.jsx
Frontend – Impact Matrix	src/modules/bia/components/ImpactAnalysis.jsx
Frontend – Critical Staff	src/modules/bia/components/CriticalStaffDetails.jsx
Backend – BIA Router	backend_brt/app/routers/bia_router.py
Backend – Process Mapping	backend_brt/app/routers/process_service_mapping.py
Backend – BIA Models	backend_brt/app/models/bia_models.py
Backend – BIA Service	backend_brt/app/services/bia_service.py

Key Components

BIA Router (bia_router.py):

Process listing filtered by hierarchy

BIA info creation and updates

Impact analysis with RTO/MTPD/criticality

Bulk update operations

API Endpoints

Method	Path	Purpose

POST	/bia/processes	Get list of processes for BIA
POST	/bia/process-info	Create BIA process info (SPOC, peak period)
POST	/bia/impact-analysis	Create impact analysis (RTO, MTPD, criticality)
PUT	/bia/impact-analysis/{id}	Update impact analysis
POST	/bia/bulk-update	Bulk update multiple BIA records

Example: Impact Analysis Request/Response

Request:

POST /bia/impact-analysis


```
{
  "process_id": "550e8400-e29b-41d4-a716-446655440000",
  "rto": "4 hours",
  "mtpd": "24 hours",
  "is_critical": true,
  "impact_data": {
    "financial": "High",
    "operational": "High",
    "legal": "Medium",
    "reputational": "Medium"
  },
  "rationale": "Payroll disruption directly impacts employees and regulatory compliance" }
```


Response:


```
{
  "id": "123e4567-e89b-12d3-a456-426614174000",
  "process_id": "550e8400-e29b-41d4-a716-446655440000",
  "rto": "4 hours",
  "mtpd": "24 hours",
  "is_critical": true,
  "impact_data": {
```

```
      "financial": "High",
      "operational": "High",
      "legal": "Medium",
      "reputational": "Medium"
    },
    "created_at": "2025-11-27T09:15:00Z",
    "created_by": "process_owner_user"
}
```

MODULE 5: CRISIS MANAGEMENT

Purpose

The Crisis Management System provides a comprehensive framework for preparing, responding to, and recovering from organizational crises. The module delivers AI-powered crisis plan generation, template management, real-time section editing, and structured crisis response workflows. It integrates communication strategies, stakeholder management, emergency procedures, and recovery protocols to ensure coordinated crisis response aligned with ISO 22301[1] and industry best practices.[2][3]

Code Paths

Component	File Path
Frontend – Crisis Management Dashboard	EY-Catalyst-front-end/src/modules/crisis-management/
Frontend – Crisis Components	EY-Catalyst-front-end/src/modules/crisis-management/components/CrisisManagement.jsx
Frontend – Crisis Service	EY-Catalyst-front-end/src/modules/crisis-management/components/crisisManagementService.js
Frontend – Crisis Styles	EY-Catalyst-front-end/src/modules/crisis-management/components/CrisisManagement.css
API Documentation	BCM_SRS.md
Backend – Core App	backend_brt/main.py
Backend – Crisis Management Router	backend_brt/app/routers/crisis_management_router.py
Backend – Crisis Models	backend_brt/app/models/crisis_management_models.py

Backend – LLM Integration Service	backend_brt/app/services/llm_integration_service.py
Backend – Database Seeding	backend_brt/app/db/seed_crisis_data.py
Backend – Dependencies	backend_brt/requirements.txt

Key Components

1. Crisis Management Router

Handles all HTTP API endpoints for crisis plan management and AI content generation.

Responsibilities:

- Crisis Plan Retrieval: Get comprehensive crisis management plans by organization
- Section Management: Update individual crisis plan sections
- AI Content Generation: Generate crisis plan content using LLM integration
- In-Memory Storage: Store and manage crisis plans for rapid access
- Template-Based Structure: Provide standardized crisis plan templates
- Multi-Section Support: Manage 10+ crisis plan sections with icons

Supported Crisis Plan Sections:

- Executive Summary (��) - Objectives, scope, governance, risk context
- Action Plan (⚡) - Immediate response, containment, recovery, stabilization
- Crisis Management Team (��) - Team roles, responsibilities, authority structure
- Key Stakeholders (��) - Internal and external stakeholder management
- Communication Strategy (��) - Channels, cadence, approval workflows
- Potential Crisis Scenarios (⚠️) - Operational, cyber, safety, supply chain scenarios

• Critical Resources (��) - Emergency funds, facilities, infrastructure, vendors

• Emergency Contacts (��) - Emergency services, hotlines, escalation contacts

• Emergency Procedures (��) - Evacuation, shelter-in-place, incident response

• Recovery & Business Continuity (��) - BIA linkage, RTO/RPO, testing

2. Crisis Management Models

Database models for storing crisis templates, plans, sections, and communications.

Key Models:

· CrisisTemplate: Uploaded crisis management templates with extracted content

· CrisisPlan: Generated crisis management plans linked to templates

· CrisisPlanSection: Individual sections of crisis plans with ordering

· CrisisCommunicationPlan: Communication strategies, media statements, FAQs

Model Features:

· UUID-based primary keys for global uniqueness

· Organization-level isolation with foreign key relationships

· Status tracking (draft, published, archived)

· JSONB storage for flexible data structures

· Timestamp tracking for audit trails

· Cascading deletes for data integrity

3. LLM Integration for Crisis Content

Leverages AI to generate industry-aligned crisis management content.

Capabilities:

· Communication Strategy Generation: Channels, stakeholders, templates, monitoring

· Crisis Team Structure: Role definitions, responsibilities, authority matrix

· Executive Summary: Objectives, scope, governance, risk context

· Action Plan: Step-by-step response and recovery procedures

· Resource Planning: Infrastructure, funds, vendors, equipment

AI Content Quality:

· Industry-standard terminology and frameworks

· Evidence-based recommendations

· Regulatory compliance alignment (ISO 22301, NIST, FEMA)[1][2]

· Organization-specific customization

· Template-based fallbacks for reliability

4. Frontend Crisis Management Component

Interactive React-based UI for crisis plan management and editing.

Features:

· Section-Based Navigation: Expandable/collapsible sections with icons

· Real-Time Editing: Update crisis plan content with immediate save

· AI-Powered Generation: Generate section content using AI

· Visual Design: Color-coded sections with professional styling

· Export Capabilities: Download crisis plans as PDF

· Multi-Organization Support: Switch between organization plans

· Responsive Layout: Mobile-friendly crisis plan viewing

5. Crisis Communication Service

Dedicated service layer for crisis communication management.

Responsibilities:

· Stakeholder Mapping: Identify and categorize stakeholders[3]

· Communication Channel Management: Email, SMS, status page, press releases

· Message Template Library: Pre-approved templates for various scenarios

· Approval Workflows: Legal and executive sign-off for public communications

· Monitoring & Control: Social listening, rumor management, fact-checking

· Multi-Channel Coordination: Synchronized messaging across platforms[3]

API Endpoints

1. Get Crisis Plan by Organization

Endpoint: GET /crisis-management/crisis-plan/{organization_id}

Purpose: Retrieve comprehensive crisis management plan for a specific organization.

Response includes:

· 10 structured crisis plan sections with headings and icons

· Executive Summary with objectives, scope, governance, risk context

· Action Plan with 12-phase response procedures

· Crisis Management Team with 10 defined roles

· Stakeholder categories (employees, customers, regulators, vendors, media, community, board, partners)

· Communication Strategy with channels and cadence

· Potential Crisis Scenarios (8 categories: operational, cyber, safety, supply chain, reputational, regulatory, site access, technology)

· Critical Resources (8 resource types: funds, facilities, backups, telecom, vendors, personnel, data, equipment)

· Emergency Contacts with 8 contact categories

· Emergency Procedures with 8 procedure types

· Recovery & Business Continuity details with BIA linkage

2. Update Crisis Plan Section

Endpoint: PUT /crisis-management/crisis-section/{organization_id}/{section_id}

Purpose: Update the content of a specific crisis plan section.

Request Body:

Content array with updated crisis section details

Response:

Success confirmation with section_id

3. Generate AI Content for Crisis Section

Endpoint: `POST /crisis-management/ai-generate-section`

Purpose: Generate AI-powered content for a specific crisis plan section.

Request Parameters:

- section_id: Target section (e.g., "communication", "crisis-team")
- section_type: Type of content to generate
- context: Organization context (name, industry, size, critical processes)

Response includes:

- AI-generated content array
- Generated_by: "LLM" or "AI"
- Section_id confirmation

Supported Sections for AI Generation:

- Executive Summary: Objectives, scope, governance, risk context
- Action Plan: Immediate actions, containment, recovery steps
- Crisis Team: Role definitions and responsibilities
- Communication Strategy: Channels, cadence, approval workflow
- Resource Planning: Facilities, funds, infrastructure needs
- Procedures: Evacuation, cyber triage, disaster response

Data Models

CrisisTemplate

Field	Type	Description
id	UUID	Unique identifier

organization_id	UUID	Organization reference
name	String	Template name
file_path	String	Path to uploaded template file
file_size	Integer	File size in bytes
content_type	String	MIME type (PDF, DOCX, etc.)
extracted_text	String	Full text extracted from document
missing_fields	JSONB	Fields requiring user input
created_at	DateTime	Creation timestamp
updated_at	DateTime	Update timestamp

Table 1: CrisisTemplate Model

CrisisPlan

Field	Type	Description
id	UUID	Unique identifier
template_id	UUID	Linked template
organization_id	UUID	Organization reference
name	String	Plan name

file_path	String	Path to generated PDF
status	String	Enum (draft, published, archived)
created_at	DateTime	Creation timestamp
updated_at	DateTime	Update timestamp

Table 2: CrisisPlan Model

CrisisPlanSection

Field	Type	Description
id	UUID	Unique identifier
crisis_plan_id	UUID	Parent plan reference
heading	String	Section title
content	String	Section content (text or JSON)
order	Integer	Display order
created_at	DateTime	Creation timestamp
updated_at	DateTime	Update timestamp

Table 3: CrisisPlanSection Model

CrisisCommunicationPlan

Field	Type	Description
id	UUID	Unique identifier
crisis_plan_id	UUID	Parent plan reference
organization_id	UUID	Organization reference
file_path	String	Path to generated communication plan PDF
media_statement	String	Pre-approved media statement template
faq	JSONB	List of frequently asked questions
stakeholder_communications	JSONB	Communication plans by stakeholder type
created_at	DateTime	Creation timestamp
updated_at	DateTime	Update timestamp

Table 4: CrisisCommunicationPlan Model

Crisis Management Workflow

Phase 1: Prevention & Preparedness

Objective: Minimize crisis likelihood and maximize readiness.

Activities:

- Risk assessment and scenario planning
- Crisis plan development and approval

- Team training and awareness programs
- Resource procurement and positioning
- Communication template preparation
- Stakeholder relationship building
- Exercise and drill programs

Phase 2: Detection & Assessment

Objective: Identify potential crises early and assess severity.

Activities:

- Incident monitoring and detection
- Severity assessment using defined criteria
- Crisis declaration decision
- Initial notification and team mobilization
- Situation assessment and intelligence gathering
- Impact analysis (people, operations, reputation)
- Resource needs identification
- Stakeholder identification

Phase 3: Response & Containment

Objective: Stabilize the situation and prevent escalation.

Activities:

- Crisis Management Team activation
- Command center establishment
- Life safety actions and incident command activation
- Incident containment measures
- Backup system activation
- Initial stakeholder communications

- Resource deployment coordination
- Regulatory notifications
- Media response coordination

Phase 4: Recovery & Restoration

Objective: Restore normal operations and services.

Activities:

- Service restoration per BIA priorities
- Alternate processing activation
- Customer service continuity management
- Vendor coordination
- Data recovery and validation
- System integrity verification
- Stakeholder update communications
- Damage assessment and documentation

Phase 5: Post-Crisis Review

Objective: Learn from the crisis and improve preparedness.

Activities:

- Incident debrief with crisis team
- Lessons learned documentation
- Root cause analysis
- Performance metrics analysis
- Plan effectiveness evaluation
- Control gap identification
- Improvement recommendations
- Plan and procedure updates

- Training material updates
- Executive reporting and follow-up

Key Features

1. Comprehensive Crisis Framework

· 10+ structured crisis plan sections with ISO 22301 alignment[1]

· End-to-end crisis lifecycle coverage

· Multi-stakeholder management capabilities

· Regulatory compliance support (ISO 22301, NIST, FEMA)

· Clear governance and decision-making authority

2. AI-Powered Content Generation

· Automated crisis plan section generation

· Organization-specific customization

· Industry best practice integration

· Template-based fallbacks for reliability

· Multi-section AI support

3. Real-Time Plan Management

· Section-based editing with immediate save capability

· In-memory storage for rapid access during crises

· Version control and change tracking

· Multi-organization support

· Status tracking (draft, published, archived)

4. Crisis Communication Integration

· Multi-channel communication strategy management

· Stakeholder-specific messaging capabilities

· Pre-approved message templates

- Approval workflow management
- Social media monitoring integration[3]

5. Visual & User-Friendly Interface

- Icon-based section navigation
- Expandable/collapsible sections for easy access
- Color-coded visual hierarchy
- Professional styling and branding
- Responsive design for all devices
- Print-friendly formatting

6. Template Management

- Upload custom crisis templates
- PDF text extraction
- Missing field identification
- Template-to-plan generation
- Organization-level templates

7. Database Persistence

- Structured data models for flexibility
- UUID-based identification
- Relationship management
- Cascading deletes for data integrity
- Audit trails with timestamps

Crisis Plan Section Details

Executive Summary (��)

Purpose: Provide high-level overview of crisis management framework.

Key Elements:

- Objectives: Protect life, stabilize operations, preserve reputation
- Scope: Coverage across sites, staff, vendors, technology
- Governance: Authority structure and decision-making hierarchy
- Risk Context: Top crisis scenarios by category
- Response Framework: Six-phase approach (Detect → Assess → Decide → Communicate → Act → Review)
- Dependencies: Critical systems, facilities, vendors
- Continuity Targets: RTO/RPO alignment with BIA
- Escalation Model: Threshold triggers and decision gates
- Communication: Single source of truth policy
- Assurance: Training, drills, audits, improvement mechanisms

Action Plan (⚡)

Purpose: Define step-by-step crisis response procedures.

Response Phases:

1. Immediate Actions: Life safety, incident command activation, situational assessment
2. Containment: Impact isolation, backup activation, asset security
3. Communication: Internal alerts, stakeholder notifications, media statements
4. Coordination: Cross-functional war room, role assignments, resource coordination
5. Logistics: Resource mobilization, alternate site readiness, vendor engagement
6. Regulatory: Authority notifications, compliance reporting, investigation support
7. Recovery: Prioritized service restoration per BIA, alternate processing
8. Stabilization: Operation verification, data integrity validation, system testing
9. Customer Care: Service credits, FAQs, dedicated support, acknowledgment
10. Post-Incident: Lessons learned, root cause analysis, action tracking
11. Resilience Uplift: Control enhancements, policy updates, prevention measures

12. Review Cadence: Executive sign-off, periodic testing, improvement cycles

Crisis Management Team (��)

Purpose: Define crisis response team roles and responsibilities.

Core Team Roles:

- Crisis Manager: Overall command, strategic decisions, final authority
- Deputy Lead: Command continuity, shift coverage, backup authority
- Communications Lead: Spokesperson, media handling, message approval
- Operations Lead: Service restoration, resource coordination
- IT/DR Lead: Infrastructure recovery, cybersecurity response
- Facilities Lead: Site safety, evacuation, alternate workspace
- HR Lead: Employee welfare, rosters, assistance programs
- Legal/Compliance: Regulatory notifications, liability management
- Finance Lead: Emergency funds, vendor payments, insurance claims
- Vendor Management: Third-party coordination, SLA escalations

Team Structure:

· 24/7 on-call rotation for core roles

· Backup personnel identified for each role

· Clear escalation paths and delegation authority

· Decision-making matrix (who can decide what)

· Contact information with multiple channels

Key Stakeholders (��)

Purpose: Identify and categorize stakeholders for targeted communication.

Stakeholder Categories:

- Employees: Safety updates, return-to-work guidance, support resources
- Customers: Service status, impact assessment, resolution timelines

- Regulators: Mandatory notices, compliance updates, investigation cooperation
- Vendors: Continuity coordination, alternate provisioning, SLA management
- Media: Fact-based updates, appointed spokesperson, press releases
- Community: Local impact mitigation, community cooperation, social responsibility
- Board/Investors: Risk exposure, financial impact, remediation plans
- Partners: Integration dependencies, contingency routes, joint response

Stakeholder Management Strategy:

· Prioritize by impact and influence

· Define communication frequency by stakeholder type

· Customize messaging for each audience

· Establish feedback mechanisms

· Monitor stakeholder sentiment

Communication Strategy (��)

Purpose: Ensure consistent, timely, and accurate crisis communications.[3]

Communication Channels:

- Email: Formal notifications, detailed updates
- SMS: Urgent alerts, time-sensitive information
- Chat/Collaboration Tools: Real-time team coordination
- Status Page: Central hub for current status
- Hotline: Dedicated crisis information line
- Press Releases: Official media statements
- Social Media: Public updates, community engagement
- Website: Crisis information portal and resource center

Communication Cadence:

- Initial Alert: Within 30 minutes of crisis declaration

- Hourly Updates: During active crisis phase
- Daily Updates: During recovery phase
- Resolution Notice: When crisis is resolved
- Post-Mortem: Lessons learned communication

Approval Workflow:

· Draft prepared by Communications Lead

· Legal review for liability and compliance

· Executive approval for public statements

· Spokesperson delivers approved message

· Monitoring and response to feedback

Potential Crisis Scenarios (⚠️)

Purpose: Identify and prepare for likely crisis scenarios.

Crisis Categories:

- Operational Outage: Facilities failure, utilities disruption, logistics breakdown
- Cyber Incident: Ransomware, data breach, DDoS, system compromise
- Safety Event: Fire, medical emergency, workplace accident, natural disaster
- Supply Chain Disruption: Vendor failure, transportation issues, material shortages
- Reputational Event: Public complaint, social media crisis, product recall
- Regulatory Action: Injunctions, fines, investigations, license suspension
- Site Access Loss: Evacuation orders, civil unrest, pandemic restrictions
- Technology Failure: Core platform outage, data corruption, integration failure

Scenario Planning Elements:

· Define trigger conditions for each scenario

· Document specific response procedures

· Identify scenario-specific resources needed

· Assign scenario leads and backup coordinators

· Conduct scenario-based drills

Critical Resources (��)

Purpose: Identify and maintain access to essential crisis response resources.

Resource Categories:

• Financial: Emergency funds, procurement fast-track, credit facilities

• Facilities: Alternate sites (hot/warm/cold), evacuation centers, assembly points

• Infrastructure: Backup systems, cloud capacity, redundant networks

• Communications: Telecom links, satellite phones, emergency broadcasting

• Personnel: Crisis team rosters, cross-trained staff, emergency contractors

• Vendors: Secondary suppliers, backup service providers, equipment rental

• Data: Backup repositories, recovery runbooks, configuration databases

• Equipment: Mobile kits, safety gear, first aid supplies, generators

Resource Management:

· Maintain current inventory of all resources

· Test resource availability quarterly

· Establish pre-agreements for rapid access

· Document activation procedures

· Track resource costs and budgets

Emergency Contacts (��)

Purpose: Provide immediate access to critical contacts during crises.

Contact Categories:

• Emergency Services: 911, police, fire, ambulance, local numbers

• Crisis Hotline: Dedicated crisis information and reporting

• Facilities: Security desk, site managers, building maintenance

- IT Support: On-call engineers, SOC contacts, vendor support
- HR: Employee helpline, EAP contact, benefits support
- Legal/Compliance: Duty counsel, regulatory liaison, insurance
- Media Relations: PR contact, spokesperson, media monitoring
- Vendors: Escalation contacts for critical suppliers

Contact Information Format:

- Name and role
- Primary contact number (mobile preferred)
- Alternate contact number
- Email address
- Escalation path if unavailable
- Geographic location/time zone

Emergency Procedures (��)

Purpose: Provide step-by-step procedures for specific emergency situations.

Procedure Types:

- Evacuation: Routes, assembly points, accountability procedures
- Shelter-in-Place: Instructions, supplies, duration guidance
- Medical Emergency: First aid steps, EMS coordination, documentation
- Cyber Incident Triage: System isolation, evidence preservation, notification
- Bomb Threat: Response protocol, search procedures, evacuation decision
- Natural Disaster: Pre-event readiness, during-event response, post-recovery
- Data Recovery: Restore procedures, validation steps, communication protocols
- Alternate Site Activation: Site preparation, transition, handover checklist

Procedure Documentation Standards:

- Clear, numbered steps

· Decision points with criteria

· Required resources and equipment

· Time estimates for each step

· Responsible roles and backups

· Success criteria and validation points

Recovery & Business Continuity (��)

Purpose: Link crisis response to broader business continuity framework.

Recovery Elements:

• BIA Linkage: Align recovery sequence with business impact analysis priorities

• RTO/RPO Targets: Define and validate recovery time and point objectives

• Alternate Processing: Manual workarounds, degraded mode operations

• Dependencies Mapping: Identify and address interdependencies

• Customer Service Continuity: Maintain service levels during recovery

• Testing Regimen: Tabletop, technical walkthrough, full-scale exercises

• Post-Recovery Verification: Operational validation, data integrity checks, sign-off

• Continuous Improvement: Lessons learned, control enhancements, plan updates

Recovery Phases:

1. Immediate Stabilization (0-4 hours): Life safety, damage assessment

2. Critical Service Restoration (4-24 hours): Highest priority processes

3. Essential Service Recovery (1-3 days): Important but non-critical processes

4. Full Service Resumption (3-7 days): All processes restored

5. Normal Operations (7+ days): Return to business-as-usual with post-incident reviews

Setup & Configuration

Backend Setup

cd backend_brt

pip install -r requirements.txt
python main.py

Environment Variables

DATABASE_URL=postgresql://user:password@localhost/bcm_db
SECRET_KEY=your_secret_key

Database Initialization & Seeding

alembic upgrade head
python -m app.db.seed_crisis_data

Frontend Setup

cd EY-Catalyst-front-end
npm install
npm run dev

Best Practices

Plan Development

- Align with organizational strategy and risk appetite[1]
- Involve stakeholders from all departments
- Ensure executive sponsorship and approval
- Reference BIA findings for recovery priorities
- Comply with regulatory requirements (ISO 22301, NIST)

Team Management

- Clearly define roles and responsibilities
- Maintain 24/7 contact availability
- Conduct quarterly training sessions
- Test handover procedures regularly
- Document decision-making authority

Communication Strategy[3]

- Establish single source of truth

- Pre-approve message templates
- Train spokespersons on media handling
- Monitor social media for misinformation
- Segment audiences for targeted messaging
- Provide regular feedback and updates

Testing & Exercises

- Conduct tabletop exercises quarterly
- Perform full-scale drills annually
- Test specific procedures after updates
- Document exercise results and findings
- Rotate team participation for broader awareness

Continuous Improvement

- Conduct post-incident reviews after every crisis or exercise
- Document lessons learned
- Update plans based on identified gaps
- Track improvement action completion
- Share lessons learned across organization

Regulatory Compliance

ISO 22301:2019 - Business Continuity Management[1]

- Clause 8.4: Establishing and implementing business continuity procedures
- Clause 8.5: Exercising and testing
- Clause 9: Performance evaluation
- Clause 10: Improvement

ISO 22301 emphasizes the need for organizations to develop and implement processes and plans to ensure timely recovery of critical activities and resources in the event of disruptions. Crisis management procedures should integrate with broader business continuity planning.[1]

NIST SP 800-34 - Contingency Planning Guide[2]

- Section 3: Contingency Planning Considerations
- Section 4: Contingency Plan Development
- Section 5: Plan Testing, Training, and Exercises
- Section 6: Plan maintenance and updates

FEMA - National Incident Management System (NIMS)

- Incident Command System (ICS): Crisis team structure follows ICS principles
- Communication and Information Management: Aligned with NIMS communication standards
- Resource Management: Resource categorization and tracking per NIMS framework

Stakeholder Communication Best Practices

Identify Stakeholder Needs[3]

- Map all internal and external stakeholders
- Understand each group's unique concerns and priorities
- Define communication frequency and preferred channels
- Establish relationship owners for key stakeholders

Choose the Right Channels[3]

- Use email for detailed information with 24-hour response target
- Deploy SMS alerts for urgent notifications (immediate response)
- Conduct video conferences for complex discussions
- Maintain social media presence for public updates
- Operate status pages as central information hub

Encourage Feedback[3]

- Create feedback systems with digital portal access
- Set response timelines (1 hour for urgent, 24 hours for standard)

- Document all communications for audit trail
- Establish escalation paths for critical issues

Build Trust Through Transparency[3]

- Acknowledge stakeholder concerns openly
- Provide data-driven updates on crisis status
- Take responsibility for gaps or errors
- Solve problems promptly with clear timelines
- Follow up on commitments made

Address Emotional and Practical Concerns[3]

- Balance emotional empathy with factual information
- Acknowledge how crises affect operations, deadlines, and business performance
- Provide support resources and assistance programs
- Show understanding of stakeholder situations
- Demonstrate commitment to resolution

Future Enhancements

Advanced AI Capabilities

- Real-time crisis severity assessment using AI
- Automated crisis scenario prediction
- AI-powered decision support during active crises
- Predictive analytics for emerging risks

Mobile Application

- Native iOS/Android crisis management app
- Offline access to critical crisis plans
- Push notifications for crisis alerts
- Mobile-friendly crisis command center

Integration Ecosystem

- Integration with incident management platforms
- Social media listening tools integration
- Emergency notification systems connectivity
- GRC platform integration
- Third-party BCM tool integration

Analytics & Reporting

- Crisis frequency and severity trending
- Response time and effectiveness analytics
- Communication effectiveness metrics
- Team performance dashboards
- Continuous improvement tracking

Collaboration Features

- Real-time collaborative crisis plan editing
- Crisis simulation environments
- Virtual crisis command center capabilities
- Discussion forums and knowledge sharing

Compliance Automation

- Automated regulatory notification tracking
- Audit trail generation and reporting
- Regulatory reporting templates
- Compliance checklist management

MODULE 5: RISK ASSESSMENT

Purpose: The Risk Assessment & Analytics Engine provides AI-powered generation of enterprise risks, threats, threat-risk records, dashboards, mitigations, and business continuity recovery strategies. The engine produces evidence-based, industry-aligned, justification-rich outputs and works independently of database storage. It integrates regulatory frameworks, statistical threat intelligence, and sector-specific context to deliver high-quality assessments.

Code Paths

Component	File Path
Frontend – Risk Assessment Dashboard	dashboard-ra/src/
Frontend – KPIs / Analytics UI	dashboard-ra/src/modules/dashboard/components/
API Documentation	API_DOCUMENTATION.md
Backend – Core App	app.py
Backend – Enterprise RA Engine	enterprise_ra.py
Backend – Threat RA Engine	dashboard_analytics.py
Backend – Business Continuity Engine	business_continuity.py
Backend – Dashboard Analytics Engine	dashboard_analytics.py
Backend – Dependencies	requirements.txt
Backend – Setup Guide	SETUP_GUIDE.md
Backend – Setup Guide	Dockerfile

Key Components

1. **Enterprise Risk Assessment Engine (enterprise_ra.py)**

Handles all enterprise risk generation using AI models.

Responsibilities:

- Generate risks based on category, department & business context
- Provide detailed **likelihood and impact scoring**
- Attach **justified threats** to each risk
- Provide **industry-specific reasoning and benchmarks**
- Provide regulatory alignment (ISO 27001, NIST, GDPR, etc.)

2. Threat Risk Assessment Engine (dashboard_analytics.py)

Generates threats, threat records, threat analysis, and bulk assessments.

Responsibilities:

- Generate threats for a selected risk
- Generate large tables of threat-risk data (10–100+ rows)
- Provide detailed likelihood/impact justifications
- Provide full scenario analysis
- Generate recommendations for executives and teams

3. Dashboard Analytics Engine (dashboard_analytics.py)

Produces summary KPIs, insights, recent activities, and assessment summaries.

Responsibilities:

- Calculate high-level KPIs (total risks, threats, critical risks)
- Generate assessment progress across RA modules
- Produce strategic insights for executives
- Generate recent activity logs for dashboards

4. Risk Mitigation Engine (/api/risk-mitigation)

Converts questionnaire responses into full mitigation assessments.

Responsibilities:

- Interpret user answers
- Identify associated risks & threats
- Provide RTO-like severity and residual risk classification
- Recommend mitigation steps & remediation timelines
- Provide trends, summaries, and control assessment

5. Business Continuity Recovery Strategy Engine (business_continuity.py)

Generates tailored recovery strategies for People, Technology, Site & Vendors.

Responsibilities:

- Analyze RTO, RPO, dependencies & requirements
- Generate actionable BCP recommendations
- Ensure ISO 22301 compliance alignment
- Provide reasoning for each strategy component

API ENDPOINTS

1. Generate Enterprise Risks

Endpoint: POST /api/enterprise-ra/generate-risks

Purpose: Generate comprehensive enterprise risks based on category, department, and business context.

Request Body:

```
{
  "category": "Technology",
  "department": "IT",
  "business_context": "Financial services organization with cloud infrastructure",
  "specific_concerns": "Data security and regulatory compliance",
  "number_of_risks": 5
}
```

Response:


```
{
  "success": true,
  "risks": [
    {
      "id": "r1a2b3c4",
      "category": "Technology",
      "name": "Data Breach",
      "description": "Sensitive customer data could be exposed through inadequate security measures",
      "likelihood": 4,
      "impact": 5,
      "likelihood_justification": "High likelihood due to 67% increase in ransomware attacks targeting financial services (FBI IC3 2024 report) and organization's cloud infrastructure exposure",
      "impact_justification": "Severe impact due to regulatory penalties (GDPR fines up to 4% of revenue), customer trust loss, and operational disruption affecting 100k+ customers",
      "treatment": "Implement multi-factor authentication and encrypt all data at rest",
      "department": "IT",
      "escalated": false,
      "threats": [
        {
          "name": "Phishing Attack",
          "description": "Attackers trick employees into revealing credentials.",
          "justification": "Phishing accounts for 36% of data breaches in financial services (Verizon DBIR 2024) and remote work has increased email-based attack surface by 40%"
        },
        {
          "name": "Malware",
          "description": "Malicious software used to steal or corrupt data.",
          "justification": "Malware attacks increased 358% in financial sector (CrowdStrike 2024) with cloud environments being primary targets due to data concentration"
```

```
      }
    ]
  }
 ],
 "message": "Successfully generated 5 enterprise risks"
}
```


2. Generate Threats for Risk

Endpoint: POST /api/enterprise-ra/generate-threats

Purpose: Generate specific threats for a given risk scenario.

Request Body:


```
{
 "risk_name": "Data Breach",
 "category": "Technology",
 "department": "IT",
 "number_of_threats": 3
}
```


Response:


```
{
 "success": true,
 "threats": [
  {
   "name": "Advanced Persistent Threat",
   "description": "Sophisticated, long-term cyber attack targeting sensitive data",
   "justification": "APTs have increased 125% in IT departments (Mandiant M-Trends 2024) with average dwell time of 146 days, making data breaches particularly damaging for technology companies"
  },
  {
```

```
    "name": "Insider Threat",
    "description": "Malicious or negligent actions by employees with system access",
    "justification": "Insider threats account for 34% of data breaches in IT sector (Ponemon Institute 2024) with privileged IT users having access to critical systems and sensitive data"
   },
   {
    "name": "Third-Party Breach",
    "description": "Security compromise through vendor or partner systems",
    "justification": "Third-party breaches affect 61% of organizations (CyberSeek 2024) with IT departments heavily reliant on cloud services, APIs, and vendor integrations increasing attack surface"
   }
  ],
  "message": "Successfully generated 3 threats for risk: Data Breach"
}
```


3. Generate Threat Risk Records

Endpoint: POST /api/threat-ra/generate-threat-risks

Purpose: Generate comprehensive threat risk records for threat risk assessment tables.

Request Body:


```
{
  "domain": "IT",
  "category": "Technology",
  "business_context": "Cloud-based infrastructure with remote workforce",
  "specific_focus": "Network security and endpoint protection",
  "number_of_records": 10
}
```


Response:


```
{
  "success": true,
```

```
"threatRisks": [
  {
    "id": "tr1a2b3c",
    "domain": "IT",
    "riskName": "Network Intrusion",
    "threat": "External Hacker",
    "vulnerability": "Unpatched Network Equipment",
    "category": "Technology",
    "likelihood": 4,
    "impact": 5,
    "rating": 20,
    "likelihood_justification": "High likelihood due to 73% of network breaches targeting unpatched systems (NIST Cybersecurity Framework 2024) and increasing sophistication of automated scanning tools",
    "impact_justification": "Severe impact as network compromise can lead to complete system access, affecting all connected services and potentially exposing customer data across multiple applications",
    "threat_justification": "External hackers represent 80% of network intrusions in IT infrastructure (CrowdStrike Global Threat Report 2024) with state-sponsored and criminal groups actively targeting technology companies",
    "vulnerability_justification": "Unpatched systems account for 60% of successful breaches (Ponemon Cost of Data Breach 2024) with IT environments often having legacy equipment and complex patch management challenges"
  },
  {
    "id": "tr4d5e6f",
    "domain": "IT",
    "riskName": "Data Exfiltration",
    "threat": "Malicious Insider",
    "vulnerability": "Excessive User Privileges",
    "category": "Technology",
    "likelihood": 3,
    "impact": 4,
```

"rating": 12,

"likelihood_justification": "Moderate likelihood as insider threats occur in 34% of data breaches (Verizon DBIR 2024) with IT staff having elevated access to sensitive systems and data repositories",

"impact_justification": "Significant impact due to potential exposure of intellectual property, customer data, and business-critical information, leading to competitive disadvantage and regulatory violations",

"threat_justification": "Malicious insiders in IT departments pose heightened risk due to technical knowledge and system access (CERT Insider Threat Guide 2024) with average incident cost of $4.9M in technology sector",

"vulnerability_justification": "Excessive privileges are found in 78% of organizations (CyberArk Privileged Access Security Report 2024) with IT environments often granting broad access for operational efficiency"

}

],

"message": "Successfully generated 10 threat risk records"

}

4. Analyze Threat Risk Scenario

Endpoint: POST /api/threat-ra/analyze-threat-risk

Purpose: Provide detailed analysis and recommendations for a specific threat risk scenario.

Request Body:

{

"domain": "HR",

"risk_name": "Key Personnel Loss",

"threat": "Employee Resignation",

"vulnerability": "No Succession Planning",

"category": "People"

}

Response:

{

"success": true,

"analysis": {

"id": "ana1b2c3",

```
    "domain": "HR",

    "riskName": "Key Personnel Loss",

    "threat": "Employee Resignation",

    "vulnerability": "No Succession Planning",

    "category": "People",

    "likelihood": 3,

    "impact": 4,

    "rating": 12,

    "likelihood_justification": "Moderate likelihood based on current job market trends showing 47% voluntary turnover rate in HR sector (SHRM Talent Acquisition Benchmarking 2024) and post-pandemic career mobility increases",

    "impact_justification": "Significant impact as key personnel departures can disrupt critical HR functions, delay strategic initiatives, and result in knowledge loss affecting employee relations and compliance",

    "threat_justification": "Employee resignation is primary threat in HR departments due to specialized knowledge requirements and limited talent pool for senior HR roles (Deloitte Human Capital Trends 2024)",

    "vulnerability_justification": "Lack of succession planning affects 67% of organizations (Harvard Business Review 2024) with HR departments often focusing on other departments' succession while neglecting their own"

  },

  "recommendations": [

    "Develop comprehensive succession plans for key HR roles including knowledge transfer protocols and cross-training programs per SHRM best practices",

    "Implement retention strategies targeting critical personnel including competitive compensation analysis and career development pathways per industry benchmarks",

    "Create knowledge documentation systems and mentorship programs to reduce single points of failure per organizational resilience frameworks",

    "Establish cross-training programs between HR team members and implement backup coverage for essential functions per business continuity standards"

  ],

  "message": "Successfully analyzed threat risk scenario"

}
```


5. Generate Bulk Threat Analysis

Endpoint: POST /api/threat-ra/generate-bulk-analysis

Purpose: Generate comprehensive threat risk analysis across multiple domains and categories.

Request Body:


```
{
  "domains": ["IT", "HR", "Finance", "Operations"],
  "categories": ["Technology", "People", "Process", "External"]
}
```


Response:


```
{
  "success": true,
  "total_records": 80,
  "threat_risks": [
    {
      "id": "bulk1a2b",
      "domain": "IT",
      "riskName": "System Downtime",
      "threat": "Hardware Failure",
      "vulnerability": "Aging Infrastructure",
      "category": "Technology",
      "likelihood": 3,
      "impact": 4,
      "rating": 12
    }
  ],
  "message": "Successfully generated 80 threat risk records across 4 domains and 4 categories"
}
```


6. Generate Dashboard KPIs

Endpoint: POST /api/dashboard/generate-kpis

Purpose: Generate realistic KPI metrics for the main dashboard based on organization context.

Request Body:


```
{
  "organization_name": "TechCorp Inc",
  "industry": "Technology",
  "departments": ["IT", "HR", "Finance", "Operations", "Legal", "Marketing"],
  "time_period": "last_30_days"
}
```


Response:


```
{
  "success": true,
  "kpis": {
    "totalRisks": 124,
    "totalThreats": 37,
    "criticalRisks": 8,
    "departments": 6,
    "kpi_justification": "Metrics aligned with technology sector benchmarks where organizations typically identify 15-25 risks per department (NIST Framework). Critical risk ratio of 6.5% reflects mature risk management with focus on high-impact scenarios. Threat-to-risk ratio of 30% indicates comprehensive threat modeling per industry standards."
  },
  "message": "Successfully generated dashboard KPI metrics"
}
```


7. Generate Assessment Summaries

Endpoint: POST /api/dashboard/generate-assessment-summaries

Purpose: Generate assessment summaries with realistic progress and key findings.

Request Body:


```
{
```

```
 "assessment_types": [
   "Critical Process RA",
   "Threat Risk Assessment",
   "Site Assessment",
   "Vendor Risk Assessment"
 ],
 "organization_context": "Mid-size financial services firm with 500+ employees"
}
```


Response:


```
{
 "success": true,
 "summaries": [
   {
     "assessmentType": "Critical Process RA",
     "completed": 12,
     "inProgress": 3,
     "pending": 2,
     "keyFindings": [
       "85% of critical processes meet documentation standards per ISO 22301 business continuity requirements",
       "2-3 processes require immediate review due to recent regulatory changes in data protection laws"
     ],
     "progress_justification": "Critical process assessments typically require 2-3 weeks each for thorough analysis. Current 70% completion rate aligns with industry standards for comprehensive process evaluation and stakeholder coordination requirements."
   },
   {
     "assessmentType": "Threat Risk Assessment",
     "completed": 8,
```

```
    "inProgress": 4,
    "pending": 1,
    "keyFindings": [
      "Phishing remains top threat vector accounting for 42% of security incidents per latest SANS survey",
      "Third-party risks increased 35% due to accelerated digital transformation and cloud adoption"
    ],
    "progress_justification": "Threat assessments require specialized cybersecurity expertise and threat intelligence analysis. 62% completion rate reflects standard pace for comprehensive threat evaluation and risk scoring methodologies."
   }
  ],
  "message": "Successfully generated assessment summaries"
}
```


8. Generate Risk Mitigation Analysis

Endpoint: POST /api/risk-mitigation

Purpose: Generate comprehensive risk analysis and mitigation plan based on user responses to risk assessment questions.

Request Body:


```
{
  "responses": [
   {
    "category": "Fire",
    "question": "Is the data centre equipped with an appropriate fire suppression system?",
    "user_answer": "We only have a few handheld fire extinguishers, and there's no automated system."
   }
  ]
}
```


Response:

```
{
  "risk_analysis": {
    "risk_id": "RISK-001",
    "category": "Fire",
    "question": "Is the data centre equipped with an appropriate fire suppression system?",
    "user_answer": "We only have a few handheld fire extinguishers, and there's no automated system.",
    "risk_name": "Absence of automated fire suppression system",
    "identified_threat": "Increased risk of fire damage and personnel danger due to lack of automatic suppression systems.",
    "likelihood": "High",
    "impact": "Severe",
    "risk_value": 9,
    "residual_risk": "Critical",
    "current_control_description": "Only basic handheld extinguishers are available; no active fire suppression in place.",
    "current_control_rating": "Poor",
    "business_unit": "Facilities",
    "risk_owner": "Fire Safety Officer",
    "timeline": "Immediate",
    "mitigation_plan": "Install automated suppression systems like FM200 or Inergen and integrate with fire alarms.",
    "summary": {
      "risk_classification_summary": "This is a critical fire safety risk with a high likelihood and severe impact. It requires immediate action.",
      "mitigation_suggestions": [
        "Deploy automated gas-based fire suppression systems.",
        "Conduct fire safety training and drills.",
        "Regularly inspect and maintain suppression systems."
      ],
      "risk_trends": {
```

```
      "top_category": "Fire",
      "risk_severity": "Critical",
      "observations": [
        "Many facilities lack automated fire suppression.",
        "High fire risks stem from outdated or manual systems.",
        "Immediate remediation is crucial to prevent major incidents."
      ]
    }
  }
 }
}
```

MODULE 6: RECOVERY STRATEGY

Purpose

The Recovery Strategy Management System provides AI-powered generation, tracking, and management of business continuity recovery strategies for critical processes. The engine produces context-aware, evidence-based recovery plans addressing People, Technology, Site, and Third-Party Vendor unavailability scenarios. It integrates BIA data, organizational context, and industry best practices to deliver actionable recovery strategies with implementation status tracking and comprehensive reasoning.

Code Paths

Component	File Path
Frontend – Recovery Strategy Dashboard	EY-Catalyst-front-end/src/modules/recovery_strategy/
Frontend – Main Components	EY-Catalyst-front-end/src/modules/recovery_strategy/RecoveryStrategyDashboard.jsx
Frontend – Recovery Strategy View	EY-Catalyst-front-end/src/modules/recovery_strategy/RecoveryStrategy.jsx
API Documentation	BCM_SRS.md
Backend – Core App	backend_brt/main.py
Backend – Recovery Strategy Router	backend_brt/app/routers/recovery_strategy_router.py

Backend – Enhanced Recovery Router	backend_brt/app/recovery_strategy_backend/enhanced_recovery_router.py
Backend – Recovery Service	backend_brt/app/services/recovery_strategy_service.py
Backend – Enhanced Recovery Service	backend_brt/app/recovery_strategy_backend/enhanced_recovery_service.py
Backend – LLM Integration Service	backend_brt/app/services/recovery_strategy_llm.py
Backend – Recovery LLM Service	backend_brt/app/recovery_strategy_backend/recovery_strategy_llm.py
Backend – Schemas	backend_brt/app/schemas/recovery_strategy_schemas.py
Backend – Models	backend_brt/app/models/recovery_strategy_models.py
Backend – Dependencies	backend_brt/requirements.txt

Key Components

1. Recovery Strategy Service Layer

Handles business logic for recovery strategy data management and retrieval.

Responsibilities:

· Data Seeding: Create sample departments, processes, and recovery strategies for demonstration

· Hierarchy Retrieval: Get departments with nested subdepartments, functions, and strategies

· Strategy Creation: Create new recovery strategies for processes

· Database Integration: Manage recovery strategy records with BIA process linkage

· Data Organization: Structure strategies hierarchically by organization, department, subdepartment, and process

Key Methods:

· create_sample_data(): Generate demonstration data for the system

· get_all_departments_with_strategies(): Retrieve complete organizational hierarchy with strategies

· seed_more_data(): Add additional sample data for diverse departments

· seed_strategies_for_existing_processes(): Auto-generate strategies for processes without them

· create_recovery_strategy(): Create new recovery strategy for a specific process

2. LLM Integration Service

Manages communication with external AI models for intelligent strategy generation.

Responsibilities:

· AI Strategy Generation: Generate recovery strategies using external LLM API

· Context-Aware Generation: Utilize BIA data, impact analysis, and operating requirements

· Strategy Reasoning: Provide detailed justifications for each recovery approach

· Fallback Mechanisms: Provide default strategies when AI generation fails

· Error Handling: Robust error handling with timeout management

· Multi-Strategy Support: Generate strategies for People, Technology, Site, and Vendor scenarios

LLM Endpoint:

https://ey-catalyst-rvce-ey-catalyst.hf.space/business-continuity/api/business-continuity/generate-recovery-strategies

Strategy Types Generated:

- People Unavailability: Cross-training, backup personnel, escalation procedures
- Technology/Data Unavailability: Backups, redundancy, disaster recovery documentation
- Site Unavailability: Alternative locations, remote work, communication protocols
- Third-Party Vendor Unavailability: Alternate vendors, emergency contacts, contingency contracts

3. Recovery Strategy Router

Handles all HTTP API endpoints for recovery strategy operations.

Responsibilities:

- CRUD Operations: Create, read, update, delete recovery strategies
- Strategy Generation: Trigger AI-powered strategy generation for processes
- Status Management: Update implementation status of strategies (Implemented, In Progress, Not Implemented)
- Statistics & Analytics: Provide summary statistics and coverage metrics
- Hierarchy Retrieval: Get organizational hierarchy with strategies
- Bulk Operations: Generate strategies for multiple processes in parallel
- Database Initialization: Set up recovery strategy tables and seed data

Key Endpoints:

- `GET /api/recovery-strategies/` - Get all recovery strategies
- `GET /api/recovery-strategies/process/{process_id}` - Get strategy for specific process
- `POST /api/recovery-strategies/generate/{process_id}` - Generate AI strategy
- `POST /api/recovery-strategies/generate-missing` - Generate strategies for all processes without them
- `PUT /api/recovery-strategies/process/{process_id}/status` - Update implementation status

· GET /api/recovery-strategies/stats/summary - Get summary statistics

· GET /api/recovery-strategies/departments/hierarchy - Get department hierarchy with strategies

· POST /api/recovery-strategies/init-db - Initialize database

4. Enhanced Recovery Strategy Service

Provides advanced recovery strategy features with dynamic configuration.

Responsibilities:

· Dynamic Strategy Configuration: Enable/disable strategy types per department

· AI Content Management: Manage AI-generated content for strategies

· Department-Level Configuration: Configure AI generation settings by department

· Content Type Selection: Allow selective strategy type generation

· Force Regeneration: Support forcing re-generation of existing strategies

· Batch Processing: Generate strategies for multiple processes simultaneously

5. Enhanced Recovery Router

Advanced API endpoints for dynamic recovery strategy management.

Responsibilities:

· Dynamic Strategy Retrieval: Get strategies based on department configuration

· Configuration Management: Update department-level recovery settings

· Targeted AI Generation: Generate AI content for specific strategy types

· Process-Level Generation: Generate strategies for individual processes

· Department-Level Generation: Batch generate strategies for entire departments

· Configuration Retrieval: Get current recovery configuration for departments

6. Frontend Components

RecoveryStrategyDashboard.jsx - Main dashboard component for recovery strategy management.

Features:

· Hierarchical view of departments, subdepartments, and processes

· Strategy status indicators (Implemented, In Progress, Not Implemented)

· Expandable/collapsible tree structure for easy navigation

· Strategy filtering and search capabilities

· Real-time status updates

· Visual indicators for strategy coverage

· Export capabilities for reporting

RecoveryStrategy.jsx - Detailed view and editing component for individual recovery strategies.

Features:

· Display all recovery strategy types for a process

· Edit strategy content and reasoning

· Update implementation status

· Generate AI-powered strategies

· View BIA context and impact analysis

· Track strategy update history

· Save and version control

API Endpoints

1. Get All Recovery Strategies

Endpoint: `GET /api/recovery-strategies/`

Purpose: Retrieve all recovery strategies with pagination support.

Query Parameters:

· `skip` (integer, default: 0) - Number of records to skip

· `limit` (integer, default: 100) - Maximum records to return

Request:

GET /api/recovery-strategies/?skip=0&limit=50

Response includes: Total count, strategies array with all strategy types and implementation statuses.

2. Get Recovery Strategy for Process

Endpoint: `GET /api/recovery-strategies/process/{process_id}`

Purpose: Retrieve detailed recovery strategy information for a specific process.

Request:

GET /api/recovery-strategies/process/a1b2c3d4-e5f6-7890-abcd-ef1234567890

Response includes:

- People unavailability strategy with reasoning and status
- Technology/data unavailability strategy with reasoning and status
- Site unavailability strategy with reasoning and status
- Third-party vendors unavailability strategy with reasoning and status
- Process vulnerability strategy with reasoning and status
- Technology unavailability strategy with reasoning and status
- Third-party unavailability strategy with reasoning and status
- Enabled strategies list
- AI-generated sections list
- AI last updated timestamp

3. Generate Recovery Strategy for Process (AI-Powered)

Endpoint: `POST /api/recovery-strategies/generate/{process_id}`

Purpose: Generate comprehensive recovery strategies using AI based on BIA data and process context.

Request:

POST /api/recovery-strategies/generate/a1b2c3d4-e5f6-7890-abcd-ef1234567890

Response includes:

- Generated strategy text for each type

· Initial implementation statuses (e.g., Not Implemented)

· Success status and confirmation message

4. Generate Missing Strategies (Bulk Generation)

Endpoint: POST /api/recovery-strategies/generate-missing

Purpose: Automatically generate recovery strategies for all processes that don't have them, using parallel processing.

Request:

POST /api/recovery-strategies/generate-missing

Response: Message confirming number of strategies generated and total count.

5. Update Strategy Implementation Status

Endpoint: PUT /api/recovery-strategies/process/{process_id}/status

Purpose: Update the implementation status of recovery strategies for tracking progress.

Request Body:

```
{
"people_status": "Implemented",
"technology_status": "In Progress",
"site_status": "Implemented",
"vendor_status": "Not Implemented",
"process_vulnerability_status": "In Progress",
"technology_unavailability_status": "Implemented",
"third_party_unavailability_status": "In Progress"
}
```

Response: Confirmation message with process_id.

6. Get Recovery Statistics Summary

Endpoint: GET /api/recovery-strategies/stats/summary

Purpose: Retrieve summary statistics for recovery strategy coverage and implementation.

Request:

GET /api/recovery-strategies/stats/summary

Response includes:

· Total processes and strategies count

· Coverage percentage

· Missing strategies count

· Status breakdown by strategy type (implemented vs not implemented)

7. Get Department Hierarchy with Strategies

Endpoint: `GET /api/recovery-strategies/departments/hierarchy`

Purpose: Retrieve complete organizational hierarchy with recovery strategies in nested structure.

Request:

GET /api/recovery-strategies/departments/hierarchy

Response: Hierarchical JSON structure with all departments, subdepartments, functions, and attached strategies for each process.

8. Initialize Recovery Strategy Database

Endpoint: `POST /api/recovery-strategies/init-db`

Purpose: Initialize database tables and seed sample data for recovery strategies.

Request:

POST /api/recovery-strategies/init-db

Response: Success status confirmation.

9. Get Enhanced Dynamic Recovery Strategies

Endpoint: `GET /enhanced-recovery-strategies/`

Purpose: Retrieve recovery strategies with dynamic configuration based on department settings.

Request:

GET /enhanced-recovery-strategies/

Response: Full strategies with department-based dynamic configuration applied.

10. Get Department Recovery Configuration

Endpoint: `GET /enhanced-recovery-strategies/department/{department_id}/config`

Purpose: Retrieve recovery configuration settings for a specific department.

Request:

GET /enhanced-recovery-strategies/department/dept-001/config

Response includes:

- Configuration ID and department reference
- Default enabled strategies list
- AI generation flag and frequency setting
- Creation and update timestamps

11. Update Department Recovery Configuration

Endpoint: PUT /enhanced-recovery-strategies/department/{department_id}/config

Purpose: Update recovery configuration for a department, affecting all processes within it.

Request Body:

```
{
"default_enabled_strategies": "people,technology,site",
"enable_ai_generation": true,
"ai_generation_frequency": "monthly"
}
```

Response: Updated configuration object.

12. Generate AI Content for Process

Endpoint: POST /enhanced-recovery-strategies/process/{process_id}/generate-ai

Purpose: Generate AI-powered recovery strategy content for a specific process.

Query Parameters:

- content_types (array of strings, default: ["all"]) - Types of strategies to generate
- force_regenerate (boolean, default: false) - Force regeneration even if content exists

Request:

POST /enhanced-recovery-strategies/process/proc-001/generate-ai?content_types=["people","technology"]&force_regenerate=true

Response: Success status, generated strategy content for specified types.

13. Generate AI Content for Department

Endpoint: POST /enhanced-recovery-strategies/department/{department_id}/generate-ai

Purpose: Batch generate AI-powered recovery strategies for all processes in a department.

Query Parameters:

- content_types (array of strings, default: ["all"]) - Types of strategies to generate
- force_regenerate (boolean, default: false) - Force regeneration even if content exists

Request:

POST /enhanced-recovery-strategies/department/dept-001/generate-ai?content_types=["all"]&force_regenerate=false

Response: Success status, count of processes updated and strategies generated.

Data Models

RecoveryStrategy

Field specifications:

- process_id (UUID) - Links to BIAProcessInfo
- people_unavailability_strategy (String) - Strategy for personnel unavailability
- people_reasoning (String) - Justification for people strategy
- people_status (Enum) - "Implemented", "In Progress", "Not Implemented"
- technology_data_unavailability_strategy (String) - Strategy for tech/data issues
- technology_reasoning (String) - Justification for technology strategy
- technology_status (Enum) - Implementation status
- site_unavailability_strategy (String) - Strategy for location unavailability
- site_reasoning (String) - Justification for site strategy
- site_status (Enum) - Implementation status
- third_party_vendors_unavailability_strategy (String) - Strategy for vendor issues

· vendor_reasoning (String) - Justification for vendor strategy

· vendor_status (Enum) - Implementation status

· process_vulnerability_strategy (String) - Strategy for vulnerability mitigation

· process_vulnerability_reasoning (String) - Justification for vulnerability strategy

· process_vulnerability_status (Enum) - Implementation status

· technology_unavailability_strategy (String) - Technology failover strategy

· technology_unavailability_reasoning (String) - Justification

· technology_unavailability_status (Enum) - Implementation status

· third_party_unavailability_strategy (String) - Third-party contingency strategy

· third_party_unavailability_reasoning (String) - Justification

· third_party_unavailability_status (Enum) - Implementation status

· enabled_strategies (String) - Comma-separated list of enabled strategy types

· ai_generated_sections (String) - Comma-separated list of AI-generated sections

· ai_last_updated (DateTime) - Last AI generation timestamp

· created_at (DateTime) - Creation timestamp

· updated_at (DateTime) - Update timestamp

DepartmentRecoveryConfig

Field specifications:

· id (UUID) - Unique identifier

· department_id (UUID) - Department reference

· default_enabled_strategies (String) - Comma-separated:
"people,technology,site,vendor,process_vulnerability"

· enable_ai_generation (Boolean) - AI generation flag

· ai_generation_frequency (String) - "daily" | "weekly" | "monthly" | "manual"

· created_at (DateTime) - Creation timestamp

· updated_at (DateTime) - Update timestamp

AIGenerationRequest

Field specifications:

· process_id (UUID, optional) - For single process generation

· department_id (UUID, optional) - For department-level generation

· strategy_types (List[String]) - ["people", "technology", "site", "vendor", "process_vulnerability"]

· force_regenerate (Boolean) - Force regeneration flag

AIGenerationResponse

Field specifications:

· success (Boolean) - Operation success flag

· message (String) - Response message

· generated_strategies (Dict[String, String]) - Strategy type → Generated content mapping

Key Features

1. AI-Powered Strategy Generation

· External LLM integration for intelligent strategy creation

· Context-aware generation using BIA data and impact analysis

· Detailed reasoning for each recovery approach

· Automatic fallback to template-based strategies if AI fails

2. Comprehensive Strategy Types

· People Unavailability: Cross-training, backup staff, succession planning

· Technology/Data: Backups, redundancy, disaster recovery, failover

· Site Unavailability: Alternative locations, remote work, relocation plans

· Vendor/Third-Party: Backup vendors, SLAs, contingency contracts

· Process Vulnerability: Single point of failure analysis, redundancy planning

3. Implementation Status Tracking

- Three-status system: Implemented, In Progress, Not Implemented
- Independent tracking for each strategy type
- Visual dashboard indicators for status overview
- Progress metrics and coverage analytics

4. Hierarchical Organization

- Organizational structure: Organization → Department → Subdepartment → Process → Strategies
- Nested JSON response format for easy frontend consumption
- Department-level configuration cascading to processes

5. Dynamic Configuration

- Enable/disable specific strategy types per department
- Configure AI generation frequency
- Selective strategy generation based on department needs
- Force regeneration capabilities

6. Bulk Operations

- Generate strategies for all processes without them
- Department-level batch AI generation
- Parallel processing for improved performance
- Missing strategy identification and auto-generation

7. Statistics & Analytics

- Coverage metrics (% of processes with strategies)
- Implementation status breakdown by strategy type
- Missing strategy identification
- Department-level analytics

Setup & Configuration

Backend Setup

```
cd backend_brt
pip install -r requirements.txt
python main.py
```

Environment Variables

```
DATABASE_URL=postgresql://user:password@localhost/bcm_db
LLM_ENDPOINT=https://ey-catalyst-rvce-ey-catalyst.hf.space/business-continuity/api/business-continuity/generate-recovery-strategies
SECRET_KEY=your_secret_key
```

Database Initialization

Initialize database

```
curl -X POST http://localhost:8000/api/recovery-strategies/init-db
```

Seed additional data

```
curl -X POST http://localhost:8000/api/recovery-strategies/seed-more-data
```

Generate strategies for existing processes

```
curl -X POST http://localhost:8000/api/recovery-strategies/seed-strategies
```

Frontend Setup

```
cd EY-Catalyst-front-end
npm install
npm run dev
```

Best Practices

Strategy Generation

- Always provide complete BIA data for better AI results
- Include impact analysis and minimum operating requirements
- Review and refine AI-generated content before finalizing
- Use force regeneration sparingly to avoid inconsistency

Status Management

- Update implementation status as strategies are deployed
- Use "In Progress" for strategies under active implementation
- Document implementation dates and responsible parties
- Conduct quarterly reviews of strategy effectiveness

Configuration

- Set department configurations before bulk generation
- Enable only relevant strategy types for each department
- Configure AI generation frequency based on business needs
- Regularly review and update department settings

Performance

- Use pagination for large datasets
- Leverage bulk generation for new departments
- Cache hierarchy data on frontend for better UX
- Monitor AI API response times and implement timeouts

Strategy Type Details

People Unavailability Strategies

Focus: Ensuring operations continue when key personnel are unavailable.

Key Elements:

- Cross-training programs for critical roles
- Succession planning and backup personnel identification
- Documentation of critical procedures and knowledge
- Emergency contact lists and escalation procedures
- Skills matrix and capability mapping

Technology/Data Unavailability Strategies

Focus: Protecting against technology failures and data loss.

Key Elements:

- Backup and recovery procedures (RPO/RTO targets)
- Redundant systems and failover capabilities
- Disaster recovery site configuration
- Data replication and synchronization
- Infrastructure-as-code for rapid rebuild

Site Unavailability Strategies

Focus: Maintaining operations when facilities are inaccessible.

Key Elements:

- Alternative work locations identification
- Remote work infrastructure and capabilities
- Emergency communication protocols
- Facility evacuation procedures
- Supply chain relocation planning

Third-Party/Vendor Unavailability Strategies

Focus: Reducing dependency risks on external providers.

Key Elements:

- Backup vendor identification and contracts
- Service level agreements (SLAs) with penalties
- Vendor financial health monitoring
- Escrow agreements for critical software
- Emergency vendor contact procedures

Process Vulnerability Strategies

Focus: Identifying and mitigating single points of failure.

Key Elements:

- Dependency mapping and analysis
- Single point of failure identification
- Redundancy implementation
- Process automation opportunities
- Regular vulnerability assessments

Troubleshooting

AI Generation Fails

- Check LLM endpoint connectivity
- Verify BIA process data exists for the process
- Review timeout settings (default: 60 seconds)
- Check fallback strategy generation working
- Verify impact analysis and operating requirements data

Missing Strategies

- Run `/api/recovery-strategies/generate-missing` endpoint
- Check BIA process info exists for all processes
- Verify process → BIA linkage in database
- Check enabled_strategies configuration

Performance Issues

- Implement pagination for large result sets
- Use parallel generation for bulk operations
- Cache department hierarchy on frontend
- Optimize database queries with indexes
- Consider async processing for large departments

Future Enhancements

Advanced AI Features

· Custom AI model training on organization-specific data

· Multi-language strategy generation

· Industry-specific template libraries

· Risk-based strategy prioritization

Collaboration Features

· Real-time collaborative strategy editing

· Comment and approval workflows

· Version control and change tracking

· Notification system for status updates

Analytics Enhancements

· Strategy effectiveness metrics

· Cost-benefit analysis for strategies

· Implementation timeline tracking

· Trend analysis and predictive insights

Integration Capabilities

· Third-party BCM tool integration

· GRC platform connectivity

· Incident management system integration

· Automated testing and validation

MODULE 7: PROCESS SERVICE MAPPING

Purpose: Parses PDF documents to extract organizational hierarchy and service maps using Groq LLM. Enables IT managers to map external applications to internal business processes.

Code Paths

Component	File Path

Frontend – Mapping UI	src/modules/process-service-mapping/ProcessServiceMaps.jsx
Backend – Mapping Router	backend_brt/app/routers/process_service_mapping.py

Key Components

Process Service Mapping Router (process_service_mapping.py):

Upload and parse PDF documents

Extract organizational structure using LLM

Delete uploaded files

API Endpoints

Method	Path	Purpose
POST	/process-service-mapping/parse-pdf/	Upload & parse PDF document
DELETE	/process-service-mapping/uploaded-files/{filename}	Delete uploaded file

MODULE 7: BCM POLICY

Purpose: ISO 22301 Gap-Analyser that helps organizations evaluate, refine, and renew their Business Continuity Management (BCM) policies. By uploading policy documents, users can automatically extract clauses, detect compliance gaps, and receive AI-generated guidance, recommendations, and renewal questions.

Code Paths

Component	File Path
backend	Policy/app

CLAUSE EXTRACTION ROUTER (clauses.py)

Key Components

- Extract ISO 22301 clauses from uploaded PDF, DOCX, or TXT documents.
- Convert documents into page-wise text using an internal page extraction function.
- Extract text from base64-encoded files for alternative request formats.
- Return the full list of ISO-required clauses with sample text.

API Endpoints

Method	Path	Purpose
POST	/clauses/extract	Extract clauses and detected clause text from a policy document.
GET	/clauses/required	Return list of ISO-required clauses with sample text

Request:

POST /api/clauses/extract


```
{
 "clauses": [
   {
    "clause": "5.3 Organizational Roles",
    "text": "The organization must define responsibilities for BCM..."
   },
   {
    "clause": "6.1 Actions to Address Risk",
    "text": "Risk assessment and treatment procedures must be documented..."
   }
 ]
}
```


GAP ANALYSIS ROUTER (gap.py)

Key Components

- Read an uploaded policy document and extract raw text.
- Compare extracted text against ISO 22301 requirements (ISO_REQ).
- Perform clause-by-clause evaluation using analyse_clause.
- Build a gap analysis report including severity, evidence, and recommendations.

API Endpoints

Method	Path	Purpose
POST	/gap/analyse	Perform full ISO 22301 clause-by-clause gap analysis.

Example: Gap Analysis Response
Request:

POST /api/gap/analyse


```
{
 "filename": "BCP_Policy.pdf",
 "total_clauses": 22,
 "gaps_found": 5,
 "summary": "Document covers 17/22 ISO 22301 clauses. Critical gaps identified in 2 clauses.",
 "details": [
  {
   "clause": "5.1 Leadership",
   "present": true,
   "evidence": "Found reference to leadership commitment...",
   "gap_severity": "none",
   "recommendation": "No action required."
  },
  {
   "clause": "8.4.3 Exercise Programme",
```

```
    "present": false,
    "evidence": "",
    "gap_severity": "high",
    "recommendation": "Define exercise objectives and frequency..."
  }
 ]
}
```


POLICY RENEWAL ROUTER (renewal.py)

Key Components

- Start policy renewal by extracting policy text and generating AI-based renewal questions.
- Refine specific clause content based on user comments.
- Regenerate an entirely new version of a clause using AI.
- Uses ISO_REQ to detect and process clause content in the uploaded or plain-text file.

API Endpoints

Method	Path	Purpose
POST	/renewal/startPolicyRenewal	Extract existing clause text and generate renewal questions.
POST	/renewal/refine	AI-refine a clause based on user comments.
POST	/renewal/regenerate	AI-regenerate a new version

		of a clause.

Example: Policy Renewal Response
Request:
POST /api/renewal/startPolicyRenewal


```
{
 "clauses": [
   {
    "clause": "5.2 Policy",
    "existing_text": "Top management shall establish a business continuity policy...",
    "questions": [
      "Does top management review the policy annually?",
      "Is the policy communicated to relevant stakeholders?"
    ]
   },
   {
    "clause": "6.1 Risk Assessment",
    "existing_text": "The organization shall determine risks that could impact operations...",
    "questions": [
      "Is there a documented methodology for assessing risks?",
      "How frequently is the risk register updated?"
    ]
   }
```

]

}

ENVIRONMENT VARIABLES (.env.example)

```
# DATABASE CONFIGURATION
DATABASE_URL=postgresql+psycopg2://user:password@localhost:54
32/bcm_db USE_SQLITE=False
SQLITE_PATH=./sqlite_db.db

# ACTIVE DIRECTORY (LDAP)
AD_SERVER_URI=ldap://your_ad_server:389
AD_BASE_DN=dc=your,dc=domain
AD_BIND_USER=your_ad_admin@domain.com
AD_BIND_PASSWORD=your_ad_password

# SECURITY
SECRET_KEY=your_secret_key_here_change_in_production
ALGORITHM=HS256
ACCESS_TOKEN_EXPIRE_MINUTES=30

# LLM API (GROQ)
GROQ_API_KEY=your_groq_api_key_here

# ADMIN CREDENTIALS (FOR DEMO/TESTING)
ADMIN_USER=Administrator
ADMIN_PASSWORD=password123

# API CONFIGURATION
API_V1_STR=/api/v1
PROJECT_NAME=Business Resilience Tool

# MONGODB (DOCUMENT STORAGE)
MONGODB_URL=mongodb+srv://user:password@cluster.mongodb.n
et/database MONGODB_DB=business_resilience

# SUPABASE (OPTIONAL)
SUPABASE_URL=https://your_supabase_url.supabase.co
SUPABASE_KEY=your_supabase_service_role_key
```

MODULE 8 - Procedures

Purpose

The BCM Procedures Management System provides AI-powered generation, versioning, and management of enterprise business continuity procedures. The engine produces evidence-based, industry-standard, justification-rich procedure documents for various BCM activities and works independently of database storage. It integrates regulatory frameworks (ISO 22301, ISO 27001), AI content generation, version control, and document lifecycle management to deliver high-quality procedure documentation.

Code Paths

Component	File Path
Frontend – Procedures Dashboard	EY-Catalyst-front-end/src/modules/procedures/
Frontend – Procedure Components	EY-Catalyst-front-end/src/modules/procedures/components/
API Documentation	BCM_SRS.md
Backend – Core App	backend_brt/main.py
Backend – Procedures Router	backend_brt/app/routers/procedures_router.py
Backend – Enhanced Procedures Router	backend_brt/app/routers/enhanced_procedure_router.py
Backend – Procedures Service	backend_brt/app/services/procedures_service.py
Backend – LLM Integration Service	backend_brt/app/services/llm_integration_service.py
Backend – Groq LLM Service	backend_brt/app/services/groq_llm_service.py

Backend – Schemas	backend_brt/app/schemas/procedures.py
Backend – Models	backend_brt/app/models/procedures_models.py
Backend – Dependencies	backend_brt/requirements.txt
Backend – Setup Guide	SETUP_GUIDE.md

Key Components

1. Procedures Service Layer

Handles all business logic for procedure document management and LLM integration.

Responsibilities:

- **Procedure Lifecycle Management**: Create, read, update, and delete procedure documents
- **Version Control**: Maintain version history and allow version rollback
- **AI Content Generation**: Integrate with LLM services to generate procedure content
- **Content Caching**: Cache AI-generated content to optimize performance
- **Template Management**: Provide default templates for different procedure types
- **Export Functionality**: Generate PDF exports of procedure documents
- **Change Log Tracking**: Maintain audit trail of all document changes

Supported Procedure Types:

- Business Impact Analysis (BIA)
- Risk Assessment
- BCM Plan Development
- Crisis Communication
- Recovery Strategy

- Testing & Exercising
- Training & Awareness
- Performance Monitoring
- Nonconformity & Corrective Actions

2. LLM Integration Service

Manages communication with AI models for intelligent content generation.

Responsibilities:

- **Multi-Format Content Generation**: Generate introduction, scope, objectives, methodology, process flows, roles & responsibilities
- **Procedure-Specific Intelligence**: Tailor content based on procedure type
- **Industry Standards Alignment**: Ensure content aligns with ISO 22301, ISO 27001, NIST standards
- **Fallback Mechanisms**: Provide default content when AI generation fails
- **Retry Logic**: Implement exponential backoff for network failures
- **Context-Aware Generation**: Use organization context and custom parameters

Specialized Content Generators:

- `_generate_bia_content()`: Impact parameters, critical processes, peak periods, impact scale matrices
- `_generate_risk_assessment_content()`: Risk parameters, control effectiveness, risk value matrices
- `_generate_bcm_plan_content()`: BCM policy, BCM questions, critical processes
- `_generate_crisis_communication_content()`: Communication channels, stakeholder matrices, message templates
- `_generate_nonconformity_content()`: Nonconformity types, corrective action steps, documentation requirements
- `_generate_performance_monitoring_content()`: Performance indicators, monitoring frequency, reporting structures
- `_generate_training_awareness_content()`: Training programs, awareness activities, competency assessments

- `_generate_testing_exercising_content()`: Exercise types, testing schedules, validation criteria
- `_generate_recovery_strategy_content()`: Recovery strategies, RTO/RPO definitions, continuity options

3. Procedures Router

Handles standard CRUD API endpoints for procedure management.

Responsibilities:

- **Organization-Specific Retrieval**: Get procedures by organization and type
- **CRUD Operations**: Create, read, update, delete procedure documents
- **Version Management**: Handle version history and version selection
- **Content Analysis**: Analyze existing procedures for improvements
- **Document Export**: Export procedures in various formats (PDF, DOCX)
- **Authentication & Authorization**: Ensure RBAC for procedure access

4. Enhanced Procedures Router

Provides advanced AI-powered procedure generation and refinement capabilities.

Responsibilities:

- **Universal Procedure Generation**: Single endpoint for all procedure types
- **AI-Powered Refinement**: Chatbot-style content refinement using natural language
- **Smart Content Editing**: Modify specific sections based on user instructions
- **Database Persistence**: Save and retrieve generated procedures
- **Version Tracking**: Maintain document versions with metadata
- **Real-time Content Validation**: Ensure JSON structure integrity

Key Endpoints:

- `POST /api/enhanced-procedures/generate`: Generate new procedure with AI
- `GET /api/enhanced-procedures/current/{procedure_type}`: Retrieve latest procedure
- `POST /api/enhanced-procedures/refine`: Refine procedure content using AI chatbot

5. Frontend Components

Rich React-based UI for procedure management and editing.

Key Components:

- ProceduresDashboard.jsx: Overview of all procedures with status indicators
- EnhancedProcedureGenerator.jsx: AI-powered procedure generation interface
- BIAProcedure.jsx: Specialized component for BIA procedures
- RiskAssessmentProcedure.jsx: Risk assessment procedure management
- BcmPlanProcedure.jsx: BCM plan development procedures
- CrisisCommunicationProcedure.jsx: Crisis communication procedures
- TestingExercisingProcedure.jsx: Testing and exercising procedures
- TrainingAwarenessProcedure.jsx: Training and awareness procedures
- PerformanceMonitoringProcedure.jsx: Performance monitoring procedures
- NonconformityProcedure.jsx: Nonconformity management procedures
- UniversalProcedureViewer.jsx: Universal procedure viewing component
- FloatingChatbot.jsx: AI assistant for procedure refinement
- ContentEditorModal.jsx: Rich text editor for procedure content
- UnifiedProcedureLayout.jsx: Consistent layout for all procedure types

API Endpoints

1. Get Procedure Templates

Endpoint: GET /procedures/templates

Purpose: Retrieve available procedure templates with default configurations.

2. Generate AI-Powered Procedure

Endpoint: POST /api/enhanced-procedures/generate

Purpose: Generate comprehensive procedure document using AI with organization-specific context.

Request Parameters:

- procedure_type: Type of procedure to generate (bia, risk_assessment, bcm_plan, etc.)
- organization_name: Organization name for context
- organization_id: Unique organization identifier
- content_types: Array of content sections to include
- custom_parameters: Industry, organization size, compliance frameworks

3. Get Current Procedure

Endpoint: GET /api/enhanced-procedures/current/{procedure_type}

Purpose: Retrieve the current active version of a specific procedure type.

4. Create/Update Procedures

Endpoints:

- POST /procedures/bia-procedure/{organization_id} - Business Impact Analysis
- POST /procedures/risk-assessment-procedure/{organization_id} - Risk Assessment
- POST /procedures/bcm-plan-procedure/{organization_id} - BCM Plan Development
- POST /procedures/crisis-communication-procedure/{organization_id} - Crisis Communication

5. Refine Procedure Content

Endpoint: POST /api/enhanced-procedures/refine

Purpose: Intelligently modify procedure content based on natural language instructions using AI.

6. Get Procedure by Type

Endpoint: GET /procedures/{procedure_type}-procedure/{organization_id}

Purpose: Retrieve a specific procedure type for an organization.

Available Procedure Types:

- bia-procedure
- risk-assessment-procedure

- bcm-plan-procedure
- crisis-communication-procedure
- nonconformity-procedure
- performance-monitoring-procedure
- testing-exercising-procedure
- training-awareness-procedure

7. Get All Organization Procedures

Endpoint: `GET /procedures/organization/{organization_id}/procedures`

Purpose: Retrieve all procedure documents for a specific organization.

8. Export Procedure to PDF

Endpoint: `GET /procedures/procedure/{procedure_id}/export?format=pdf`

Purpose: Export a procedure document to PDF format.

9. Delete Procedure

Endpoint: `DELETE /procedures/procedure/{procedure_id}`

Purpose: Delete a procedure document (soft delete with audit trail).

10. Regenerate Procedure

Endpoint: `POST /procedures/regenerate`

Purpose: Regenerate an existing procedure with fresh AI content.

11. Analyze Existing Procedure

Endpoint: `POST /procedures/analyze-existing`

Purpose: Analyze an existing procedure for gaps, improvements, and compliance alignment.

12. Get Procedure Versions

Endpoint: `GET /procedures/versions/{procedure_type}`

Purpose: Get all versions of a specific procedure type with version history.

Data Models

ProcedureDocument

Field	Type	Description
id	Integer	Unique identifier
procedure_type	String	Enum: bia, risk_assessment, bcm_plan, etc.
organization_id	Integer	Organization reference
document_name	String	Document title
document_owner	String	Owner name
document_version_no	String	Version number
document_version_date	String	Version date
prepared_by	String	Preparer name
reviewed_by	String	Reviewer name
approved_by	String	Approver name
use_llm_content	Boolean	AI-generated flag
llm_content	JSON	Generated content
custom_content	JSON	User overrides
is_current	Boolean	Current version flag
created_at	DateTime	Creation timestamp
updated_at	DateTim	Update timestamp

	e	
created_by	Integer	Creator user ID
updated_by	Integer	Updater user ID

Table 1: ProcedureDocument Model

LLMContent

The LLMContent object contains:

- **Standard Sections**: introduction, scope, objective, methodology, process_flow, roles_responsibilities, review_frequency
- **BIA-Specific**: impact_parameters, critical_processes, peak_periods, impact_scale_matrix
- **BCM Plan-Specific**: bcm_policy, bcm_questions
- **Risk Assessment-Specific**: risk_parameters, control_effectiveness, risk_value_matrix

ChangeLogEntry

Field	Type	Description
sr_no	Integer	Serial number
version_no	String	Version identifier
approval_date	String	Approval date
description_of_change	String	Change description
reviewed_by	String	Reviewer name
approved_by	String	Approver name

Table 2: ChangeLogEntry Model

Key Features

1. AI-Powered Content Generation

- Leverages Groq LLM for intelligent procedure content generation
- Context-aware generation based on organization profile and industry
- Supports 9+ procedure types with specialized content
- Automatic alignment with ISO 22301, ISO 27001, NIST frameworks

2. Version Control & Audit Trail

- Complete version history tracking
- Change log for every modification
- Ability to rollback to previous versions
- Approval workflow with multi-level review

3. Intelligent Content Refinement

- AI chatbot for natural language content editing
- Section-specific modifications
- Context preservation during edits
- Fallback mechanisms for AI failures

4. Template System

- Pre-configured templates for all procedure types
- Default document structures
- Industry best practices embedded
- Customizable for organization-specific needs

5. Export & Reporting

- PDF export with professional formatting

· DOCX and HTML export options

· Branded templates with org logos

· Automated report generation

6. Compliance Alignment

· ISO 22301 (Business Continuity) alignment

· ISO 27001 (Information Security) references

· NIST framework integration

· Regulatory compliance tracking

Setup & Configuration

Backend Setup

```
cd backend_brt
pip install -r requirements.txt
python main.py
```

Environment Variables

```
DATABASE_URL=postgresql://user:password@localhost/bcm_db
GROQ_API_KEY=your_groq_api_key
LLM_ENDPOINT=https://api.groq.com/openai/v1
SECRET_KEY=your_secret_key
```

Database Migration

```
alembic upgrade head
```

Frontend Setup

```
cd EY-Catalyst-front-end
npm install
npm run dev
```

Best Practices

Content Generation

· Always provide organization context for better AI results

· Use custom parameters to fine-tune generated content

- Review and refine AI content before finalizing
- Cache frequently used content for performance

Version Management

- Increment version numbers for major changes (1.0 → 2.0)
- Use minor versions for updates (2.0 → 2.1)
- Maintain detailed change logs
- Set current version only after approval

Security

- All endpoints require authentication
- RBAC enforced for procedure access
- Audit logs for all modifications
- Secure storage of sensitive procedure content

Performance

- Use LLM content caching (24-hour TTL)
- Implement retry logic for API failures
- Optimize database queries with indexes
- Lazy load procedure content in UI

Troubleshooting

AI Generation Fails

- Check Groq API key configuration
- Verify LLM endpoint connectivity
- Review API rate limits
- Check fallback content generation

Version Conflicts

- Ensure unique version numbers

- Check current version flags
- Review change log consistency
- Verify approval workflow

Export Issues

- Confirm PDF library installation (ReportLab)
- Check file permissions for export directory
- Verify template files exist
- Review export format configuration

Future Enhancements

- **Multi-language Support**: Generate procedures in multiple languages
- **Collaborative Editing**: Real-time collaborative procedure editing
- **Workflow Automation**: Automated approval routing and notifications
- **Advanced Analytics**: Procedure usage analytics and effectiveness tracking
- **Integration APIs**: Third-party system integrations for procedure sync
- **Mobile Support**: Mobile-optimized procedure viewing and editing
- **AI Training**: Custom AI model training on organization-specific content

MODULE 9 - BCM PLAN DASHBOARD

Purpose

The BCM Plan Management System provides comprehensive business continuity planning capabilities at both organizational and departmental levels. The module delivers AI-powered plan generation, real-time dashboard analytics, multi-level plan management, PDF export functionality, and structured BCM frameworks aligned with ISO 22301 standards. It integrates BIA data, recovery strategies, crisis management, and regulatory compliance requirements to deliver actionable, evidence-based business continuity plans.

Code Paths

Component	File Path
Frontend – BCM Dashboard	EY-Catalyst-front-end/src/modules/bcm/BCMDashboard.jsx
Frontend – BCM Standalone	EY-Catalyst-front-end/src/modules/bcm/BCMStandalone.jsx
Frontend – Organization Plan	EY-Catalyst-front-end/src/modules/bcm/OrganizationBCMPlan.jsx
Frontend – Departmental Plan	EY-Catalyst-front-end/src/modules/bcm/DepartmentalBCMPlan.jsx
Frontend – Department Dashboard	EY-Catalyst-front-end/src/modules/bcm/DepartmentDashboard.jsx
Frontend – Department List	EY-Catalyst-front-end/src/modules/bcm/DepartmentList.jsx
Frontend – API Service	EY-Catalyst-front-end/src/modules/bcm/apiService.js
Frontend – Styles	EY-Catalyst-front-end/src/modules/bcm/BCMDashboard.css
API Documentation	BCM_SRS.md

Backend – Core App	backend_brt/main.py
Backend – BCM Router	backend_brt/app/routers/bcm_router.py
Backend – BCM Plan Service	backend_brt/app/services/bcm_plan_service.py
Backend – Groq LLM Service	backend_brt/app/services/groq_llm_service.py
Backend – BCM RBAC Middleware	backend_brt/app/middleware/bcm_rbac.py
Backend – BCM Debug Utilities	backend_brt/app/utils/bcm_debug.py
Backend – Dependencies	backend_brt/requirements.txt

Key Components

1. BCM Router

Comprehensive API endpoints for BCM plan management, dashboard analytics, and PDF generation.

Responsibilities:

- Dashboard Analytics: Real-time statistics for processes, BIA completion, criticality
- Department Management: Hierarchical department data with completion rates
- Process Management: Process listing with BIA status and criticality
- Staff Management: Critical staff identification and role mapping
- BCM Plan Generation: Organization and departmental level plan creation
- BCM Plan Updates: Edit and update existing plans

· PDF Export: Generate professional BCM plan PDFs

· Demo Data Seeding: Populate database with sample BCM plans

· Recovery Strategy Stats: Aggregate implementation status tracking

· Audit Trail: Activity logging and review tracking

· RBAC Integration: Role-based access control for plan viewing and editing

API Endpoint Categories:

1. Dashboard Endpoints: Statistics, departments, processes

2. Process Endpoints: All processes, department-specific processes

3. Staff Endpoints: Critical staff listing

4. BCM Plan Endpoints: Get, create, update organization and departmental plans

5. Export Endpoints: PDF generation for BCM and crisis plans

6. Utility Endpoints: Database seeding, health checks, audit trails

2. BCM Plan Service

Service layer for BCM plan generation using LLM integration.

Responsibilities:

· Organization-Level Plan Generation: Executive BCM framework with governance, scope, objectives

· Departmental-Level Plan Generation: Department-specific plans with recovery objectives

· AI-Powered Content Creation: Leverage Groq LLM for intelligent content generation

· Plan Storage & Retrieval: Store plans in JSON format within organization/department descriptions

· Plan Updates: Update existing plans with new content

· Bulk Plan Seeding: Generate plans for all organizations and departments

· Fallback Mechanisms: Provide template-based content when AI fails

· Error Handling: Robust exception handling with detailed logging

Plan Components Generated:

- Organization Plans: Introduction, Purpose, Scope, Governance, Crisis Team, Communications, BIA Summary, Recovery Strategies, Testing & Maintenance
- Departmental Plans: Introduction, Purpose, Scope, Critical Applications, Response Matrix, Recovery Objectives, Roles, Resources, Training, Testing, Review

3. Groq LLM Service Integration

AI-powered content generation using Groq LLM API.

Capabilities:

- Context-Aware Generation: Uses organization/department names and context
- Industry-Standard Content: Produces ISO 22301-aligned BCM content
- Multi-Section Support: Generates introduction, scope, objectives, strategies, etc.
- Async Processing: Non-blocking LLM API calls
- Retry Logic: Handles API failures gracefully
- Caching: Optional caching for improved performance

4. Frontend BCM Dashboard

Interactive React-based dashboard for comprehensive BCM management.

Features:

- Statistics Overview: Total processes, completed BIA, critical processes, departments
- Department Cards: Visual cards showing department stats and completion rates
- Process Listing: Filterable process table with criticality and status
- Critical Staff View: Key personnel identification across critical processes
- Navigation: Quick access to organization and departmental plans
- Real-Time Updates: Live data synchronization
- Responsive Design: Mobile-friendly layout
- Export Actions: PDF generation for plans

5. Organization BCM Plan Component

Detailed view and editor for organization-level BCM plans.

Sections:

- Executive introduction and business context
- Purpose and strategic objectives
- Enterprise-wide scope definition
- BCM governance structure
- Crisis management team composition
- Communication protocols and procedures
- Business impact analysis summary
- Enterprise recovery strategies
- Testing, training, and maintenance schedules

Features:

- Section-based editing with rich text support
- AI-powered content generation per section
- Version control and change tracking
- Export to PDF
- Approval workflow integration
- Compliance mapping (ISO 22301)

6. Departmental BCM Plan Component

Department-specific business continuity plan management.

Sections:

- Department introduction and context
- Department-specific objectives
- Departmental scope and boundaries
- Critical applications and data backup strategies

· Response and escalation matrix (Minor, Moderate, Major, Critical)

· Recovery objectives and prioritized activities (RTO, MTPD)

· Roles and responsibilities matrix

· Critical resource and asset requirements

· Training and awareness programs

· Testing procedures and schedules

· Review and maintenance cadence

Features:

· Department-specific customization

· Multi-level escalation matrix

· RTO/RPO alignment with BIA

· Critical resource mapping

· Drill and exercise planning

· Performance metrics tracking

7. RBAC Middleware

Role-based access control for BCM plan management.

Permissions:

· VIEW_ORG_PLAN: View organization-level BCM plans

· EDIT_ORG_PLAN: Edit organization-level BCM plans

· VIEW_DEPT_PLAN: View departmental BCM plans

· EDIT_DEPT_PLAN: Edit departmental BCM plans

· EXPORT_BCM_PLAN: Export BCM plans to PDF

· ADMIN_BCM: Full administrative access

Access Control:

· Organization-level plan access restricted to authorized users

· Department-level plan access based on department assignment

· Export functionality requires specific permissions

· Audit logging for all plan modifications

API Endpoints

1. Get Dashboard Statistics

Endpoint: GET /bcm/dashboard/stats

Purpose: Retrieve high-level BCM statistics for dashboard display.

Query Parameters:

· organization_id (string, optional): Filter by specific organization

Response:

{
"total_processes": 87,
"completed_bia": 65,
"pending_bia": 22,
"critical_processes": 34,
"total_departments": 12,
"completion_rate": 75
}

2. Get Departments with Statistics

Endpoint: GET /bcm/departments

Purpose: Retrieve all departments with their BCM completion statistics.

Response (Example):

[
{
"id": "dept-001",
"name": "Information Technology",
"organization_id": "org-123",
"total_processes": 25,
"completed_bia": 22,
"critical_processes": 12,
"completion_rate": 88
},
{
"id": "dept-002",
"name": "Finance",

"organization_id": "org-123",
"total_processes": 18,
"completed_bia": 15,
"critical_processes": 8,
"completion_rate": 83
}
]

3. Get All Processes

Endpoint: `GET /bcm/processes/`

Purpose: Retrieve all business processes with their BIA status.

Response (Example):
[
{
"id": "proc-001",
"name": "Payroll Processing",
"department": "Finance",
"criticality": "Critical",
"status": "Completed"
},
{
"id": "proc-002",
"name": "Network Operations",
"department": "Information Technology",
"criticality": "Critical",
"status": "Completed"
}
]

4. Get Department Processes

Endpoint: `GET /bcm/departments/{department_id}/processes`

Purpose: Retrieve processes for a specific department.

5. Get Critical Staff

Endpoint: `GET /bcm/critical-staff`

Purpose: Identify critical staff members across critical business processes.

Response (Example):
[
{
"name": "Sarah Johnson",

"role": "Information Technology Head",
"department": "Information Technology"
},
{
"name": "Michael Chen",
"role": "Network Operations Owner",
"department": "Information Technology"
}
]

6. Get Organization-Level BCM Plan

Endpoint: GET /bcm/organization-plan/{organization_id}

Purpose: Retrieve comprehensive organization-level BCM plan.

Response Includes:

- organization_name, plan_type, plan_version, generated_date
- introduction: Business context and BCM framework overview
- purpose_and_objective: Strategic objectives and goals
- scope: Coverage across locations, units, processes
- governance: BCM governance structure and oversight
- crisis_management_team: Team roles and responsibilities
- communication_protocols: Multi-channel communication strategy
- business_impact_analysis_summary: BIA summary with RTO/RPO targets
- recovery_strategies: Enterprise recovery approaches
- testing_and_maintenance: Testing schedule and plan maintenance

7. Get Departmental BCM Plan

Endpoint: GET /bcm/department-plan/{department_id}

Purpose: Retrieve department-specific BCM plan.

Query Parameters:

- organization_id (string, required): Organization identifier

Response Includes:

· Department-specific introduction and objectives

· Departmental scope and critical applications

· Response and escalation matrix (Minor, Moderate, Major, Critical)

· Recovery objectives with RTO and MTPD targets

· Roles and responsibilities by position

· Critical resource and asset requirements

· Training and awareness program details

· Testing procedures and schedules

· Review and maintenance cadence

8. Update Organization BCM Plan

Endpoint: PUT /bcm/organization-plan/{organization_id}

Purpose: Update organization-level BCM plan with new content.

RBAC Required: EDIT_ORG_PLAN permission

Request Body: Updated plan sections with new content

Response:

```
{
"status": "success",
"message": "Organization BCM plan updated successfully",
"organization_name": "TechCorp Global Inc"
}
```

9. Update Departmental BCM Plan

Endpoint: PUT /bcm/department-plan/{department_id}

Purpose: Update department-level BCM plan.

RBAC Required: EDIT_DEPT_PLAN permission

10. Seed BCM Plans

Endpoint: POST /bcm/seed-plans

Purpose: Generate and populate BCM plans for all organizations and departments.

Response:

{
"status": "success",
"seeded_organizations": 3,
"seeded_departments": 15
}

11. Seed Demo BCM Data

Endpoint: POST /bcm/seed-demo

Purpose: Populate database with comprehensive demo BCM plans for testing.

12. Generate BCM Plan PDF

Endpoint: POST /bcm/generate-pdf

Purpose: Generate professional PDF document for BCM or Crisis Management plans.

Request Body (BCM Plan):

{
"plan_type": "organization",
"organization_id": "org-123"
}

Response:

- Content-Type: application/pdf
- Professional header with organization name and logo
- Table of contents with all plan sections
- Page numbers and footer information
- Generated timestamp and version information

13. Get Recovery Strategy Statistics

Endpoint: GET /bcm/recovery-strategies/stats

Purpose: Aggregate recovery strategy implementation status.

Response:

{
"implemented": 28,
"in_progress": 15,
"not_started": 7,

"total": 50
}

14. Get Audit Trail

Endpoint: GET /bcm/audit-trail

Purpose: Retrieve recent BCM activity and changes.

Query Parameters:

- organization_id (string, optional): Filter by organization
- limit (integer, default: 10): Number of records to return

15. Get Upcoming Reviews

Endpoint: GET /bcm/upcoming-reviews

Purpose: Retrieve scheduled BCM plan reviews and exercises.

16. Test Database Connection

Endpoint: GET /bcm/test-connection

Purpose: Health check for database connectivity.

Response:
{
"status": "ok",
"db": true
}

Data Models

Organization BCM Plan

{
"organization_name": String,
"plan_type": String, # "organization_level"
"plan_version": String, # Version number
"generated_date": String, # ISO date format
"introduction": String, # Executive introduction
"purpose_and_objective": String, # Strategic objectives
"scope": String, # Coverage and boundaries
"governance": String, # BCM governance structure
"crisis_management_team": String, # CMT composition
"communication_protocols": String, # Communication framework

"business_impact_analysis_summary": String, # BIA summary
"recovery_strategies": String, # Enterprise recovery strategies
"testing_and_maintenance": String # Testing schedule
}

Departmental BCM Plan

{
"organization_name": String,
"department_name": String,
"plan_type": String, # "departmental_level"
"plan_version": String,
"generated_date": String,
"introduction": String,
"purpose_and_objective": String,
"scope": String,
"communication_protocols": String,
"business_impact_analysis_summary": String,
"critical_applications_and_data_backup_strategies": String,
"response_and_escalation_matrix": {
"minor": { "description": String, "response_time": String, "escalation_level": String },
"moderate": { "description": String, "response_time": String, "escalation_level": String },
"major": { "description": String, "response_time": String, "escalation_level": String },
"critical": { "description": String, "response_time": String, "escalation_level": String }
},
"recovery_objectives_and_prioritized_activities": {
"rto": String,
"mtpd": String,
"prioritized_activities": String
},
"recovery_strategies": String,
"roles_and_responsibilities": {
"department_head": String,
"bcm_coordinator": String,
"operational_team": String,
"it_support": String
},
"critical_resource_and_asset_requirements": String,
"training_and_awareness": String,
"testing": String,
"review_and_maintenance": String
}

BCM Plan Framework

Organization-Level Plan Structure

1. Introduction

Establish context and authority for the BCM program with organizational background, BCM program history, regulatory drivers, and alignment with ISO 22301 standards.

2. Purpose and Objective

Define strategic intent with objectives including: protect life and safety, stabilize operations, meet regulatory obligations, preserve reputation, maintain customer confidence, and support rapid recovery.

3. Scope

Define coverage across geographic locations, business units, critical processes, IT platforms, facilities, third-party vendors, and stakeholder engagement with clear boundaries and exclusions.

4. Governance

Establish authority and accountability through BCM Steering Committee with executive sponsorship, BCM Manager role, budget allocation, performance metrics, and external audit oversight.

5. Crisis Management Team

Define crisis response organization with Crisis Manager (overall command), Deputy Lead (continuity), Communications Lead (spokesperson), Operations Lead, IT/DR Lead, Facilities Lead, HR Lead, Legal/Compliance, Finance Lead, and Vendor Management roles.

6. Communication Protocols

Ensure timely, accurate communications through multi-channel approach (email, SMS, Teams, hotline, status page), stakeholder segmentation, message templates, spokesperson policy, approval workflow, and communication cadence by severity.

7. Business Impact Analysis Summary

Link BCM plan to BIA findings with critical process identification, RTO/RPO targets, dependency mapping, impact assessment, prioritized restoration sequence, and BIA refresh cycle.

8. Recovery Strategies

Define specific strategies including data protection (backups, geo-redundancy), alternate sites (warm/hot site, cloud failover), manual workarounds, vendor contingencies, staff resilience, and customer support continuity.

9. Testing and Maintenance

Ensure plan currency through testing regimen (monthly contact validation, quarterly tabletops, semi-annual walkthroughs, annual full-scale exercises) and maintenance activities (periodic reviews, change integration, post-exercise updates).

Departmental Plan Structure

Departmental plans follow similar structure with department-specific context, critical applications and backup strategies, four-tier escalation matrix (Minor, Moderate, Major, Critical), RTO/MTPD targets, and department-specific recovery strategies and resources.

Key Features

1. **Multi-Level Planning**
    - o Organization-level enterprise BCM framework
    - o Department-level detailed operational plans
    - o Hierarchical plan structure with inheritance
    - o Plan version control and change tracking
2. **AI-Powered Content Generation**
    - o Groq LLM integration for intelligent content
    - o Context-aware generation using org/dept names
    - o Industry-standard BCM terminology
    - o ISO 22301-aligned structure
    - o Template-based fallbacks
3. **Comprehensive Dashboard**
    - o Real-time statistics and KPIs
    - o Department completion tracking
    - o Process criticality visualization
    - o Critical staff identification
    - o Drill-down navigation
4. **RBAC Integration**
    - o Organization-level plan permissions

- Department-level plan permissions
- Export and audit permissions
- Fine-grained access control
- Audit logging for all changes

5. **PDF Export**
   - Professional BCM plan documents
   - Crisis management plan export
   - Custom branding and styling
   - Section-based formatting
   - Automated generation timestamps
6. **Response Matrix Framework**
   - Four-tier severity model
   - Time-based response targets
   - Clear escalation paths
   - Authority definition by severity
   - Pre-defined notification lists
7. **Recovery Objectives Tracking**
   - RTO/RPO definition and monitoring
   - MTPD calculation and validation
   - Prioritized activity sequencing
   - Dependency management
   - Testing and validation

Setup & Configuration

Backend Setup

cd backend_brt

pip install -r requirements.txt
python main.py

Environment Variables

DATABASE_URL=postgresql://user:password@localhost/bcm_db
GROQ_API_KEY=your_groq_api_key
SECRET_KEY=your_secret_key
DEBUG=false

Database Initialization

alembic upgrade head
curl -X POST http://localhost:8000/bcm/seed-plans
curl -X POST http://localhost:8000/bcm/seed-demo

Frontend Setup

cd EY-Catalyst-front-end
npm install
npm run dev

Best Practices

Plan Development

- Involve stakeholders from all departments
- Align with organizational strategy and risk appetite
- Ensure executive sponsorship and budget allocation
- Reference BIA findings for all recovery priorities
- Comply with ISO 22301 and regulatory requirements

Testing & Validation

- Test plans at least annually
- Include external stakeholders in exercises
- Document all test results and gaps
- Track improvement actions to completion
- Update plans based on test findings

Maintenance

- Review plans quarterly
- Update immediately after organizational changes
- Integrate with change management processes
- Validate contact information monthly
- Ensure version control and approval workflow

Governance

- Establish BCM Steering Committee with executive sponsorship
- Define clear roles and responsibilities
- Set performance metrics and KPIs
- Conduct annual maturity assessments
- Report to board at least annually

Regulatory Compliance

ISO 22301:2019 Alignment

Clause 8.4 - Business Continuity Procedures

- Organization and departmental plans established
- Procedures documented and maintained
- Recovery strategies defined
- Testing and exercise program in place

Clause 9 - Performance Evaluation

- Dashboard statistics and KPIs
- Recovery strategy tracking
- Testing results documentation
- Management review processes

Clause 10 - Improvement

· Post-incident review process

· Continuous improvement framework

· Action tracking and completion

· Plan update procedures

Troubleshooting

AI Generation Issues

· **Problem**: AI-generated content is generic

· **Solution**: Provide more organizational context

· **Fallback**: Use template-based content and customize manually

Plan Storage Issues

· **Problem**: Plans not persisting in database

· **Solution**: Verify description field in organization/department tables

· **Check**: JSON format validation

PDF Generation Failures

· **Problem**: PDF export returns 500 error

· **Solution**: Verify ReportLab installation

· **Check**: Plan data completeness and format

RBAC Permission Denied

· **Problem**: Users cannot access plans

· **Solution**: Verify user roles and permissions

· **Check**: Organization/department assignment

Future Enhancements

Advanced AI Capabilities

· Real-time plan quality assessment

· Gap analysis against ISO 22301

- Automated compliance mapping
- Intelligent recommendation engine

Collaboration Features

- Real-time collaborative editing
- Comment and review workflow
- Approval routing and notifications
- Version comparison and diff view

Analytics & Reporting

- BCM maturity dashboard
- Drill success rate tracking
- Plan coverage heatmaps
- Trend analysis and predictions

Integration Ecosystem

- GRC platform integration
- Incident management systems
- HR systems for staff data
- Asset management for inventory

Mobile Support

- Native mobile apps (iOS/Android)
- Offline plan access
- Push notifications for updates
- Mobile-optimized PDF viewer

Workflow Automation

- Automated review reminders
- Schedule management for drills

· Action item tracking and alerts

· Approval workflow automation

DATABASE TABLES BY MODULE

Module	Primary Tables	Purpose
Auth & RBAC	users, roles, permissions, organizations, departments, processes	User management and access control
Admin & Dashboard	organizations, departments	Organization hierarchy and stats
BCM/Risk Assessment	bcm_plans, procedure_documents	Plan content and SOP versioning
BIA	bia_process_info, process_impact_analysis, global_process	Process impact data
Crisis Management	crisis_plans, crisis_sections	Crisis plan structure and content
Recovery Strategy	recovery_strategy	4-pillar recovery data

TESTING & HEALTH CHECKS

Backend Health Verification

```
# 1. Swagger UI (Test all endpoints)
curl http://localhost:8000/docs

# 2. Test Authentication
curl -X POST http://localhost:8000/auth/token \
    -H "Content-Type: application/x-www-form-urlencoded" \
    -d "username=Administrator&amp;password=password123"

# 3. Store token for subsequent requests
TOKEN="&lt;access_token_from_response&gt;"
```

```
# 4. Test Dashboard Stats
curl -H "Authorization: Bearer $TOKEN" \
  http://localhost:8000/bcm/dashboard/stats
# 5. Test BIA Endpoints
curl -X POST http://localhost:8000/bia/processes \
  -H "Authorization: Bearer $TOKEN" \
  -H "Content-Type: application/json" \
  -d '{"organization_id":"org_1"}'
```

Frontend Verification

Open http://localhost:5173 in browser

Navigate to Login page

Enter credentials from `.env` (ADMIN_USER, ADMIN_PASSWORD)

Verify all modules accessible in sidebar

Test dashboard loads statistics correctly

COMMON ISSUES & SOLUTIONS

Issue	Root Cause	Solution
AD connection failed	LDAP server unreachable or incorrect credentials	Verify AD_SERVER_URI, AD_BASE_DN in .env; test connectivity with ldapsearch command; ensure VPN/network access to AD server
JWT token invalid or expired	Token signature mismatch or token lifetime exceeded	Check SECRET_KEY and ALGORITHM match between frontend and backend; verify ACCESS_TOKEN_EXPIRE_MINUTES setting
LLM API calls fail	Invalid or expired Groq API key	Verify GROQ_API_KEY in .env; check Groq account for API access; test API key directly with Groq docs
Database connection error	Database not running or incorrect URL	Ensure PostgreSQL/SQLite running; verify DATABASE_URL syntax; check database exists and credentials correct

Module endpoint returns 404	Router not registered or wrong path	Verify router file exists in app/routers/; check router is imported and included in FastAPI app setup in main.py
CORS errors on frontend	Frontend and backend have different origins	Verify FRONTEND_BASE_URL in backend; check FastAPI CORS middleware configured correctly
PDF export fails	ReportLab missing or PDF generation error	Verify ReportLab installed in requirements.txt; check PDF template syntax in bcm_plan_service.py

DEPLOYMENT CHECKLIST

[ ] **Code Repository:** Final commit tagged as `v1.0-deliverable` and pushed to canonical URL

[ ] **Environment Variables:** `.env.example` created with all required variables; actual `.env` configured with production values

[ ] **Database:** PostgreSQL initialized with schema; migrations run; sample data seeded (if applicable)

[ ] **Backend:** FastAPI server starts without errors; Swagger UI accessible at `/docs` [ ] **Frontend:** React app builds successfully; npm run dev works; login page loads [ ] **Active Directory:** AD connection verified; test user can login

[ ] **LLM API:** Groq API key validated; test generation call succeeds

[ ] **API Testing:** All 30+ endpoints tested; sample responses verified [ ] **Security:** JWT tokens valid; RBAC enforced; sensitive data not exposed in logs

DOCUMENT INFORMATION

Field	Value
Document	1.0

Version	
Date	November 28, 2025
Modules Documented	7 (Auth, Admin, BCM, BIA, Crisis, Recovery, Mapping)
API Endpoints	30+ verified endpoints
Code Verified Against	Current codebase as of Nov 28, 2025
Status	Ready for Development & Deployment
Repository	[PLACEHOLDER: Insert canonical GitHub URL]
Commit Hash	[PLACEHOLDER: v1.0-deliverable or commit hash]

Document prepared by: Development Team
Last updated: November 28, 2025
Next review: December 5, 2025