

Backend/Cloud Developer - Climatise SECR Platform (AI-Enhanced Engineering)

Company Overview

Climatise is a leading carbon emissions tracking and reporting platform designed to help companies monitor, analyze, and report their greenhouse gas emissions. We provide comprehensive solutions for large enterprises with multiple facilities to gain insights into their carbon emissions through advanced data aggregation and conversion methodologies.

Position Overview

We are seeking an experienced Backend/Cloud Developer to join our team and contribute to the development and maintenance of our SECR (Streamlined Energy and Carbon Reporting) platform. You will work on a sophisticated cloud-native architecture built on AWS, handling large-scale data processing, real-time analytics, and regulatory compliance reporting.

What Makes This Role Unique: We embrace state-of-the-art AI-assisted development workflows as a core part of our engineering practice. You'll be expected to leverage cutting-edge AI coding tools to enhance productivity, code quality, and documentation throughout the software development lifecycle.

Key Responsibilities

Infrastructure & DevOps

- Design, implement, and maintain AWS cloud infrastructure using **Terraform** (Infrastructure as Code)
- Manage multi-environment deployments (Development, Staging, Production) with automated CI/CD pipelines
- Configure and optimize **AWS Lambda** functions for serverless computing
- Implement and maintain **VPC**, security groups, and network configurations
- Manage **AWS RDS Aurora PostgreSQL** clusters with backup and disaster recovery strategies
- Configure **DynamoDB** tables for high-performance NoSQL data storage

Backend Development

- Develop Python-based **AWS Lambda functions** for data processing and API resolvers
- Build and maintain **GraphQL APIs** using **AWS AppSync**
- Implement **event-driven architecture** using **SNS/SQS** for asynchronous processing

- Develop data processing pipelines for carbon emissions calculations and aggregations
- Create and maintain **repository pattern** implementations for data access layer
- Work with **Pydantic** models for data validation and serialization

Data Management & Processing

- Design and implement **ETL pipelines** for carbon emissions data processing
- Work with conversion factors from multiple sources (SWC, IEA, DEFRA, AIG)
- Implement **batch processing jobs** for large-scale data aggregation
- Manage **S3** storage for file uploads and processed data
- Develop **utility bill extraction** systems using **AWS Textract** and **Bedrock**
- Build analytics and reporting features for emissions tracking

API Development & Integration

- Develop **GraphQL resolvers** and mutations for frontend integration
- Implement **AWS Cognito** authentication and authorization
- Build **RESTful APIs** and webhooks for third-party integrations
- Create **real-time notification systems** using SNS/SES
- Develop **chatbot services** using **AWS Bedrock**

Monitoring & Optimization

- Implement comprehensive logging using **AWS Lambda Powertools**
- Set up **AWS X-Ray** distributed tracing for performance monitoring
- Configure **CloudWatch** dashboards and alarms
- Optimize Lambda function performance and cost efficiency
- Implement error handling and retry mechanisms

AI-Enhanced Development Workflows

- Utilize **AI coding assistants** (Cursor, Claude Code, GitHub Copilot) for daily development tasks
- Leverage AI tools for **requirements analysis** and **user story generation**
- Employ AI-powered **codebase analysis** tools for understanding large codebases and identifying technical debt
- Use AI assistants for **automated documentation generation** and maintenance
- Create **implementation plans** using AI-assisted planning tools
- Maintain comprehensive **documentation trails** with AI-generated technical specs and ADRs (Architecture Decision Records)
- Apply AI tools for **code review**, **refactoring**, and **optimization suggestions**
- Use AI for **test generation** and **test coverage improvement**

Required Technical Skills

Core Technologies

- **Python 3.12+** with strong expertise in modern Python development
- **AWS Cloud Services** - extensive experience with Lambda, RDS, DynamoDB, S3, SNS, SQS
- **Terraform** for Infrastructure as Code
- **GraphQL** and **AWS AppSync**
- **PostgreSQL** and **Aurora Serverless**
- **DynamoDB** NoSQL database design and optimization

AWS Services Expertise

- **AWS Lambda** (Python 3.12 runtime, layers, environment configuration)
- **AWS AppSync** (GraphQL API development and resolver implementation)
- **Amazon RDS Aurora PostgreSQL** (cluster management, backup strategies)
- **Amazon DynamoDB** (table design, streams, global secondary indexes)
- **Amazon S3** (bucket policies, lifecycle management, event notifications)
- **AWS SNS/SQS** (message queuing, event-driven architecture)
- **AWS Cognito** (user pools, identity management, JWT handling)
- **AWS Secrets Manager** (credential rotation, secure access)
- **AWS VPC** (networking, security groups, VPC endpoints)
- **AWS Textract** and **AWS Bedrock** (document processing, AI/ML services)

Development Tools & Practices

- **Git** version control with branching strategies
- **Docker** for containerization and local development
- **pytest** for comprehensive testing strategies
- **pre-commit hooks** and **ruff** for code quality
- **AWS CLI** and **boto3** SDK
- **CI/CD pipelines** with AWS CodePipeline
- **Monitoring** with CloudWatch and X-Ray

Programming & Frameworks

- **Pydantic** for data validation and settings management
- **psycopg** for PostgreSQL connectivity
- **AWS Lambda Powertools** for observability

AI-Assisted Development Tools (Required)

AI Coding Assistants

- **Cursor** for AI-enhanced IDE experience with multi-file awareness and autonomous coding
- **Claude Code** CLI for complex refactoring, codebase analysis, and autonomous task execution

- **GitHub Copilot** for inline suggestions and code completion
- Experience with **AI agent workflows** (Cursor Composer or similar)

AI-Powered Requirements & Planning Tools

- AI-assisted **user story generation** and **acceptance criteria** definition
- Experience using LLMs for **requirements analysis** and **gap identification**

AI-Powered Documentation Tools

- **AI-generated technical documentation** using Claude, GPT-4, or similar
- **Automated ADR (Architecture Decision Records)** generation
- **Inline documentation** and **docstring generation** with AI assistance
- **Diagram generation** from code using AI tools (Mermaid, PlantUML with AI assistance)

AI-Powered Code Analysis

- **Codebase understanding** using tools like Sourcegraph Cody or Claude Code with large context windows
- **AI-powered code review** tools (Qodo, CodeRabbit, or similar)
- **Technical debt analysis** using AI assistants
- **Refactoring suggestions** from AI tools

Best Practices for AI-Assisted Development

- Treat AI as a **pair programmer** requiring direction and oversight
- Provide **maximum context** to AI tools for optimal results
- Break tasks into **manageable chunks** for AI processing
- Implement **rigorous code review** of AI-generated code
- Use **frequent, granular version control** when working with AI-generated code
- **Plan before coding** - always have a clear structure before AI assistance
- Maintain **human oversight** - AI accelerates but doesn't replace judgment

Preferred Qualifications

Experience

- **3+ years** of backend development experience with Python
- **2+ years** of AWS cloud development experience
- **1+ years** with Terraform and Infrastructure as Code
- **1+ years** actively using AI coding assistants in production workflows
- Experience with **serverless architectures** and event-driven systems
- Experience with **GraphQL** API development
- Background in **data processing** and **ETL pipelines**

- Demonstrated proficiency with **AI-assisted development workflows**

Domain Knowledge

- Understanding of **carbon emissions tracking** and environmental data
- Experience with **regulatory compliance** and reporting requirements
- Knowledge of **financial data processing** and multi-currency systems
- Familiarity with **supplier engagement** and supply chain data

AI-Enhanced Engineering Competencies

- **Advanced prompt engineering** skills for AI coding assistants
- Experience with **AI-powered codebase navigation** in large codebases (50k+ LOC)
- Proficiency in **AI-assisted debugging** and **root cause analysis**
- Ability to **critically evaluate** and **refine AI-generated code**
- Understanding of **AI tool limitations** and when human expertise is required
- Experience **training team members** on AI-assisted development practices

Soft Skills & Work Practices

- **Daily reliability** - consistent presence and dependable execution of commitments
- **Meeting attendance** - punctual participation in scheduled meetings and standups
- **Overcommunication** - proactively share updates, blockers, and progress without being prompted
- **Documentation trail** - leave clear documentation for all work, decisions, and code changes
- **Proactive collaboration** - reach out to team members before issues escalate, don't wait to be asked
- **Product engineering mindset** - think beyond tickets; understand the "why" and contribute to product direction
- **Active ownership** - take initiative on improvements, not just passive task execution
- Strong problem-solving and analytical skills
- Experience with **agile development** methodologies
- Ability to work in a fast-paced, collaborative environment
- Understanding of **security best practices** in cloud environments
- **Growth mindset** toward adopting new AI tools and workflows

We need active product engineers, not passive executors. You should be comfortable challenging assumptions, proposing improvements, and driving initiatives forward.

Technical Environment

Architecture

- **Serverless-first** approach with AWS Lambda
- **Event-driven** microservices architecture

- **Multi-tenant** SaaS platform with company isolation
- **GraphQL** API layer with real-time subscriptions
- **Hybrid database** strategy (PostgreSQL + DynamoDB)

Development Workflow

- **Git Flow** branching strategy with ultra-granular commits for AI-assisted work
- **Automated testing** with pytest and mocking frameworks
- **Pre-commit hooks** for code quality enforcement
- **Infrastructure as Code** with Terraform
- **Multi-environment** deployment pipeline (Dev → Staging → Production)
- **AI-enhanced code review** process with human oversight
- **Integrated AI workspace** with Cursor + Claude Code + GitHub Copilot

Code Quality Standards

- **Type hints** throughout Python codebase
- **Repository pattern** for data access abstraction
- **Comprehensive logging** and error handling
- **Unit and integration testing** with high coverage
- **Security-first** development practices
- **AI-generated documentation** with human review and validation
- **Automated code quality checks** enhanced by AI analysis

AI Integration Standards

- **Context-aware AI usage** - provide full context for complex tasks
- **Incremental AI assistance** - small, focused prompts for specific tasks
- **Mandatory code review** - all AI-generated code reviewed line-by-line
- **Documentation trail** - AI-assisted technical decisions documented in ADRs
- **Version control discipline** - frequent commits when working with AI tools
- **Human-in-the-loop** - critical decisions reviewed by senior engineers

What We Offer

- Competitive salary and equity package
- **Premium subscriptions** to AI coding tools (Cursor Pro, Claude, GitHub Copilot)
- Opportunity to work on cutting-edge climate technology with **cutting-edge AI workflows**
- Flexible remote/hybrid working arrangements
- Professional development budget for:
 - AWS certifications
 - AI/ML training and certifications
 - AI coding tools and platforms

- Health and wellness benefits
- Collaborative, mission-driven team environment
- Direct impact on global carbon emissions reduction efforts
- **Innovation time** for exploring new AI tools and improving team workflows

Evaluation Criteria

During the interview process, we will assess:

- Technical proficiency in **Python, AWS, and Terraform**
- Ability to **leverage AI tools effectively** for requirements, coding, and documentation
- **Prompt engineering** skills and AI tool selection
- **Critical thinking** when reviewing AI-generated code
- Understanding of **when to use AI** vs. when human expertise is essential
- **Code quality** and **security awareness** in AI-assisted workflows

Join us in building the future of carbon emissions tracking and help organizations worldwide achieve their sustainability goals through technology. Be part of a team that combines cutting-edge AI-assisted development practices with meaningful climate impact.

Why AI-Enhanced Development at Climatise?

At Climatise, we believe that **AI coding assistants are transforming software engineering**, moving developers from code writers to code editors and architects. By joining our team, you'll:

- Work in an environment where **AI tools are celebrated**, not feared
- Gain expertise in **state-of-the-art AI development workflows** (2026 standards)
- Help shape our **AI-enhanced engineering culture**
- Reduce manual coding effort by up to **50%** while maintaining high quality
- Focus on **architecture and problem-solving** while AI handles boilerplate
- Be part of a team where **~90% of code leverages AI assistance** (industry-leading adoption)

We're not just building climate tech - we're building it faster, smarter, and more efficiently with AI.