

Renée Ionescu

Senior Software Engineer
Email: renee.ionescu@example.com
Phone: +40 700 000 002
Location: București, Romania
Link: <https://example.com/renee>
Link: <https://github.com/example-renee>
Fictional sample data. Not a real candidate.

Summary

Software engineer leading reliable API and data-platform work. Builds observable services, documents operational tradeoffs and mentors engineers through delivery and incident review.

Experience

Senior Software Engineer | Example Systems

2023-03 - Present

- Designed a PostgreSQL-backed job queue with idempotent workers and recovery tests for interrupted tasks.
- Introduced OpenTelemetry traces across API and worker boundaries so on-call engineers could isolate slow requests.
- Led design reviews for a staged migration from synchronous processing to background jobs, documenting rollback criteria.
- Mentored engineers through incident reviews and paired on regression tests for duplicate delivery and partial failure.

Software Engineer | Sample Cloud

2020-01 - 2023-02

- Implemented versioned Node.js APIs and contract tests to keep customer integrations working through releases.
- Reworked database queries after profiling production-like workloads; verified query plans before adding indexes.
- Built an accessible React console for reviewing failed jobs and retrying only the affected work.

Junior Software Engineer | Demo Works

2017-07 - 2019-12

- Developed Python import tools that preserved source identifiers and rejected ambiguous records for manual review.
- Added integration tests around CSV exports, Unicode names and date handling to prevent silent data corruption.

Education

MSc | Distributed Systems | Example Technical University
Education date: 2019-06
BSc | Computer Science | Example Technical University
Education date: 2017-06

Skills

TypeScript; React; Node.js; PostgreSQL; Python; Git; Docker; Kubernetes; Redis; OpenTelemetry

Projects

Queue Observatory

2024-05 - Present

Technologies: Python, PostgreSQL, Docker
URL: <https://example.com/queue-observatory>

- Built an independent workload simulator that reproduces worker crashes and compares retry policies.
- Published runnable test cases and design notes explaining when idempotency keys expire and how recovery is verified.

Accessible Booking Lab

2022-04 - 2022-10

Technologies: TypeScript, React, Node.js
URL: <https://example.com/accessible-booking>

- Created a keyboard-accessible booking demo with time-zone conversion and appointment-conflict tests.
- Documented browser support and known limitations so contributors could reproduce failures.