mohbisher.com
github.com/mohbisher
linkedin.com/in/moebisher

Mohammed Bisher

Software Engineer

+966590816125
mohammed@mohbisher.com
MEDINA, SA (ABLE TO RELOCATE)

BRIEF

A passionate Software Engineer with 5 years of experience designing and building well-crafted software solutions, with a strong focus on backend development and hands-on full-stack capabilities. Over the past 2 years, I've also worked as a tech advisor for startups, helping early-stage teams shape their technical strategies and bring products to market. I thrive in dynamic environments, enjoy solving complex problems, and am now seeking a new software engineering role where I can deepen my expertise and contribute to impactful, high-growth projects.

WORK EXPERIENCE

Intern Software Engineer	**Sary, Hybrid**	**Sep 2022 – Feb 2023, 6m**
Junior Software Engineer	**Sary, Hybrid**	**Mar 2023 – June 2024, 1y 4m**
Mid Software Engineer	**Sary, Hybrid**	**May 2024 – June 2025, 1y 3m**

After nearly three years as a Django Backend Engineer at Sary, I've tackled nearly every aspect of the framework, from crafting a numerus amount of new endpoints and models to building heavy service layers and integrating with third-party systems and so on, focusing below on the high-impact areas that showcase significant technical depth and exposure:

- Dealt with structuring applications with DDD aggregates and bounded contexts which helped in breaking down complex business logic into cohesive modules, allowing for simpler development cycles and reduced system complexity.
- Dealt with profiling and rewriting complex queries with custom QuerySets, index optimizations, annotations, and aggregations which helped in speeding up data retrieval by squashing N+1 query issues and improving DB roundtrips, which in turn ensured smooth performance for high-traffic, data-heavy endpoints.
- Dealt with rolling out Redis-backed caching layers with smart key versioning, TTL management, and orchestrated invalidation workflows which helped in easing database strain, allowing for fresh data delivery and sub-second response times even during traffic spikes.
- Dealt with building a lot of high-throughput bulk data services with atomic commits which helped in syncing massive datasets efficiently, setting the stage for consistent data and streamlined operations in high-volume environments.
- Dealt with setting up Celery task queues with prioritized routing, retry policies, and fault-tolerant pipelines using Redis which helped in offloading heavy jobs, which in turn kept APIs responsive and boosted throughput for data-intensive workflows.
- Dealt with implementing feature flags with runtime toggles and DRF versioning strategies which helped in safely rolling out experimental features, cutting down regression risks while keeping legacy clients fully supported.
- Dealt with customizing Django admin views with overridden templates and jQuery-driven dynamic forms which helped in creating intuitive, interactive interfaces, allowing for efficient handling of complex, interlinked data models with real-time updates.
- Dealt with building comprehensive error handling with structured logging frameworks which helped in speeding up issue diagnosis through detailed request contexts, setting the stage for quick resolution of production incidents.
- Dealt with writing extensive Pytest suites with parameterized fixtures, performance profiling, and coverage-driven test frameworks which helped in validating app behavior across varied scenarios, which in turn minimized regressions and optimized runtime efficiency for a solid codebase.
- Dealt with overhauling legacy codebases through modular refactoring, query optimizations, performance-driven migrations, and caching which helped in modernizing the architecture, paving the way for less technical debt, easier maintenance, and faster critical endpoint responses.
- Guided and mentored junior engineers by sharing best practices, like writing clean code and testing thoroughly, helping them grow into confident developers.
- Took initiatives to improve different codebases for better readability and performance, and also recommended adopting modern tools like to simplify and streamline development, collaborating with the team to integrate them effectively.

An impactful project at Sary: Supersonic Products Pipeline

- **Problem**: Internal agents followed a seven-step, labor-intensive process to manually enter product details into traditional forms, receiving hundreds to thousands of products in supplier spreadsheets, with each product taking 3 to 5 minutes to complete.
- **Solution**: Led an initiative to fully automate the pipeline from start to finish, allowing agents to receive supplier spreadsheets, upload them into the tool, and create thousands of products in seconds, including all possible validations, image handling, and production-ready details. The solution used multi-level services with smart validation handling to avoid exhausting the processor and ensure fast data storage.
- **Impact**: Using the tool I built, agents have become able to process 1000 products 99.9% faster, reducing the time from 3000 minutes/1k item to less than 30 seconds/1k items, enabling rapid handling of high-volume inventories.

Another one: Algolia Sync Symphony

- **Problem**: Algolia's (a powerful search provider) integration with our system fell short of its potential, struggling with slow search processes and inaccurate data stored in Algolia's tables.
- **Solution**: Co-developed a robust solution by designing a flexible search service abstraction layer and a sophisticated query builder for complex, multilingual search parameters. Implemented real-time data synchronization using a queue-based architecture with Celery and Redis, ensuring scalability for high-volume requests. Integrated advanced features like personalization, analytics tracking, and business segment-specific search results.
- **Impact**: Reduced search latency by 50%, achieved high data sync accuracy with Algolia, and supported a huge increase in search query volume, paving the way for a boost in user engagement through faster, more relevant results.

One more cz we still have space: Promotions Cache-Crushing

- **Problem**: Promotions, including offers, tiers, and bundles, relied on outdated implementations that failed to align with new algorithms and consumed excessive caching and Redis capacity.
- **Solution**: Led extensive refactorings to align promotions with updated systems, while preserving proper versioning. Pivoted the caching approach from storing a large number of update keys to a daily background job using Celery to optimize storage and prevent loss of task keys during instance failures.
- **Impact**: Transformed promotions into scalable, extensible entities easily integrated across systems, and slashed production Redis instance size from 12GB to 5GB, paving the way for cost-effective operations.

Intern Software Engineer **Solutions by 42, Dammam** **June 2022 – Aug 2022**

- Mainly worked on 2 projects using .NET, you know APIs, models, testing and stuff, not much to mention here honestly.

LANGUAGES AND TECHNOLOGIES

- **Languages**: Python, JavaScript, C#, Java, SQL, HTML & CSS, PHP, Swift
- **Frameworks**: Django Rest Framework, Node.js, .NET, Spring Boot, Next.js, React.js, Vue.js, Flask, FastAPI, Express.js, Laravel, Gatsby
- **Technologies**: Git, Docker, PostgreSQL, MongoDB, Postman, Kubernetes, Redis, RabbitMQ, Elasticsearch, AWS, Azure, Grafana, Nginx, Celery, Ansible, Terraform, Swagger/OpenAPI, Jest, Selenium, Kafka, MySQL, Firebase

EDUCATION

Madinah, SA **University of Prince Mugrin** **Aug 2018 – Dec 2022**

- Bachelor's Degree in Software Engineering. (3.86/4 GPA)
  - Awarded 1st in academic distinction in the College of Computer Science and Information Technology.
  - Worked as a TA teaching Python to new students. Also led the software engineering club (1100+ members).