Team Name: Timeless Innovators

Members:

- Rajdeep Das
- Debasmita Dutta
- Prof. (Dr.) Indrajit De

Problem Statement: Post-Quantum DNSSEC Testbed under a Delegated .IN Domain

TABLE OF CONTENTS

Blog link

Introduction

- **Theme:** Post-Quantum DNSSEC Testbed under a Delegated .IN Domain — Implementation, Measurement, and Open-Source Contributions
- **Focus Areas:** DNSSEC
- **Organized by:** Advanced Internet Operations Research in India (AIORI)
- **Collaborating Institutions:** Institute of Engineering and Management (IEM), Kolkata, India
- **Date:**11/2025
- **Prepared by:**

Name	Designation	Institution
Rajdeep Das	Student	IEM
Debasmita Dutta	Student	IEM
Dr. Indrajit De	Professor	IEM

Contact: (Debasmita.Dutta2022@iem.edu.in)

- **Executive Summary**

Classical public-key algorithms employed by DNSSEC (RSA, ECDSA, EdDSA) are susceptible to future quantum attacks. Emerging PQC signature algorithms standardized or proposed by NIST [5] provide quantum-resistant alternatives, but they produce significantly larger public keys and signatures. DNS, as deployed today, relies heavily on small UDP packets and mature resolver implementations that assume conservative DNSKEY / RRSIG sizes. This project implements and measures a PQC-aware DNSSEC validation path to evaluate practical deployment concerns and to produce code and operational guidance that inform IETF DNSOP [8] and implementers.

- **Overview**

This document describes the design, implementation, test results, and open-source artifacts produced for a Post-Quantum Cryptography (PQC) aware DNSSEC testbed. The work demonstrates DNSSEC validation using PQC signature algorithms (CRYSTALS-Dilithium (Dilithium3), Falcon-512, and SPHINCS+) in a controlled delegated domain (*.iem.lab). The primary deliverables are: (1) a multi-instance authoritative/server architecture using PowerDNS with per-algorithm zones and databases, (2) a PQC verification middleware that enables Unbound recursive resolvers to validate PQC DNSSEC signatures without modifying resolver internals, (3) C boilerplates integrating liboqs / OQS-OpenSSL for key generation, signing and verification, (4) performance measurement scripts and results for concurrent DNS queries, and (5) open-source-ready artifacts and documentation. This report is intended to serve as an implementation reference for the IETF DNSOP PQ-DNSSEC discussion and for practitioners evaluating PQC adoption in DNS infrastructure.

2. Terminology and Conventions

- PQ — Post-Quantum (cryptography)
- PQC — Post-Quantum Cryptography
- DNSSEC — Domain Name System Security Extensions [1][2][3]
- RRSIG — DNSSEC signature resource record [2]
- DNSKEY — DNSSEC public key resource record [2]
- AD flag — Authenticated Data flag in DNS response (set by validating resolvers) [3]
- Unbound — Recursive resolver used in the testbed
- PowerDNS — Authoritative DNS server used in the testbed
- liboqs / OQS-OpenSSL — Open Quantum Safe libraries [6] used for PQC operations
- VM100 / VM102 / VM103 — Virtual machines used in the testbed (authoritative, recursive/middleware, client)
- Conventional key and signature sizes discussed are empirically measured within the testbed and are approximate for the specific parameter sets used (e.g., Dilithium3, Falcon-512, SPHINCS+ variants).

3. Objectives and Scope

3.1 Objectives

- Demonstrate end-to-end DNSSEC validation when zones are signed using PQC signature algorithms (Dilithium3, Falcon-512, SPHINCS+) [5].
- Implement a PQC-capable middleware enabling existing Unbound resolvers to validate PQC signatures without source changes.
- Measure and report signature sizes, query latency, throughput (QPS), CPU/memory overhead, and network effects across concurrency levels.
- Produce open source reference implementations (C boilerplates with liboqs [6], Python middleware) and reproducible deployment instructions.
- Provide operational feedback and empirical data to IETF DNSOP PQ-DNSSEC discussions [8].

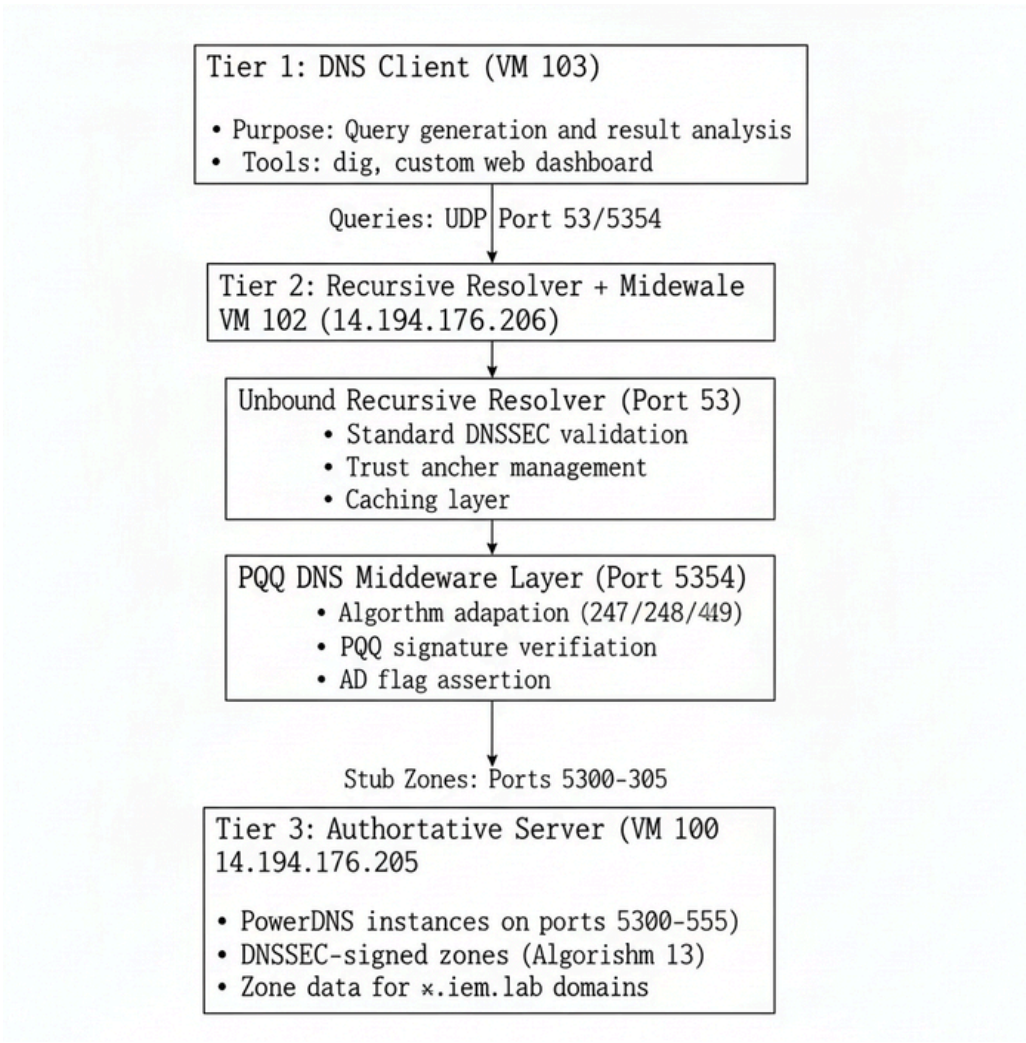
3.2 Scope

- Controlled testbed (Proxmox VMs) — no public global authoritative zone changes.
- Focuses on signature generation/verification path; does not cover registrar/registry rollouts or global delegation mechanics.
- Includes hybrid and transition strategies as future work, but not implemented in this sprint.

4. System Design and Architecture

- **4.1 Testbed Topology**

- The testbed is separated into three logical tiers:
- Tier 1 — DNS Clients (VM103): Test clients issuing queries; dig, custom scripts and dashboard.
- Tier 2 — Recursive Resolver + PQC Middleware (VM102, 14.194.176.206): Unbound (port 53) performs standard validation [3] when possible; PQC middleware (port 5354) performs PQC adaptations and returns validated responses to clients with AD set.
- Tier 3 — Authoritative Servers (VM100, 14.194.176.205): Multiple PowerDNS instances (ports 5300–5305) each serving a zone and signing records with a specific algorithm [4].



- **4.2 Zone and Algorithm Mapping**

Port	Zone	Algorithm
5300	iem.lab (baseline)	ECDSA P-256 (Algorithm 13) [4]
5305	dilithium.iem.lab	Dilithium3 (PQC) [5]
5304	falcon.iem.lab	Falcon-512 (PQC) [5]
5303	sphincs.iem.lab	SPHINCS+ (PQC) [5]

Each zone has its own MySQL backend to ensure isolation and reproducibility.

- **4.3 Middleware Design**

The PQC middleware is implemented in Python (dnslib + liboqs bindings [6]). Responsibilities:

  - Convert algorithm identifiers from PQC test zones into a form Unbound/clients can understand.
  - Perform PQC signature verification using liboqs [6] where the resolver lacks native support.
  - Assert the AD flag [3] on validated responses and cache results to improve throughput.
  - Handle EDNS0 and UDP fragmentation edge cases to manage large RRSIG payloads [2].

The middleware operates as a validation proxy: it receives responses from authoritative servers, verifies signatures, and provides validated answers to the recursive resolver or directly to clients depending on configuration.

5. Implementation Details

- **5.1 Software Components and Versions**
  - PowerDNS 4.5.3 (authoritative)
  - Unbound 1.19.x (recursive resolver)
  - MySQL 8.0 (backend for PowerDNS instances)
  - liboqs (latest stable built in testbed)
  - OQS-OpenSSL (patched OpenSSL for PQC key ops)
  - Python 3.11 with dnslib and liboqs bindings for middleware
  - Build tools: gcc, cmake, make

- **5.2 PQC Toolchain and Libraries**
  - Cloned and built liboqs and OQS-OpenSSL from Open Quantum Safe repositories [6].
  - Implemented C boilerplate programs for each algorithm (Dilithium3, Falcon-512, SPHINCS+) [5] using the OQS SIG API: key generation, signing, verification, base64 encoding for DNSKEY storage [2] , and sample benchmarking harnesses.

- **5.3 PowerDNS Multi-instance Setup and MySQL Backends**
  - Four PowerDNS instances were configured each with a separate gmysql backend pointing to powerdns, pdns_dilithium, pdns_falcon, pdns_sphincs.
  - Each PowerDNS instance had its own pdnsutil configuration namespace and systemd unit for lifecycle management.

- **5.4 Key Management and Simulated PQC Keys**
  - For reproducibility during development, simulated key files were created in /etc/powerdns/pqc-keys as placeholders where full OQS keypair integration was not yet desired for test runs.
  - For full PQC signing flows, boilerplate code performed OQS keypair generation [6] and inserted the base64-encoded public keys [2] into cryptokeys/records tables where needed.

- **5.5 Middleware Implementation (Python)**
  - The middleware listens on port 5354, accepts DNS queries, forwards to the appropriate authoritative instance, retrieves RRSIG and DNSKEY [2] , and verifies the signature using liboqs [6] .
  - Verified responses are returned with AD set [3] and optionally cached in a local store for repeated queries.
  - EDNS0 buffer sizes and DO/AD/TC handling are implemented; middleware ensures proper response truncation or TCP fallback when necessary.

- **5.6 Build and Execution (C Boilerplates)**
  - Provided Makefile.dilithium, Makefile.falcon, etc. to compile the implementation examples.
  - Example run sequence: make -f Makefile.dilithium && ./dilithium3_dnssec — this exercises key generation, signing, verification, tamper tests, and benchmarks.

6. Sprint Methodology and Timeline

Work was organized into weekly sprints with clear deliverables:

- Week 1 (Infrastructure): Proxmox VM creation, OS hardening, network setup.
- Week 2 (Server Setup): PowerDNS and MySQL multi-instance configuration; baseline ECDSA zone deployment [4] .
- Week 3 (PQC Toolchain): liboqs and OQS-OpenSSL build [6] ; initial C boilerplates implemented.
- Week 4 (Middleware & Integration): Python middleware implemented; Unbound integration tested.
- Week 5 (Testing & Measurement): Performance scripts executed (concurrency tests: 1, 5, 10), packet captures and CPU/memory monitoring.
- Ongoing: Documentation, packaging artifacts and preparing open source releases.

Sprint workflow: plan → implement → integration test → benchmark → document → open-source push.

7. Test Plan and Measurement Framework

- **7.1 Functional Validation**
  - Verify DNSSEC chain of trust [1] for each algorithm zone: DNSKEY present [2] , RRSIGs valid [2] , AD flag [3] asserted by resolver or middleware.
  - Tamper tests: modify zone content and confirm verification failures.
- **7.2 Performance Benchmarking (Concurrency Tests)**
  - Run concurrent query tests for each domain at concurrency levels 1, 5, and 10 queries in parallel.
  - Measure: average response time (ms), success rate, QPS (queries per second), and efficiency/scaling metrics.
  - Domains tested: falcon.iem.lab, dilithium.iem.lab, sphincs.iem.lab, iem.lab (baseline), and google.com (external compare).
- **7.3 Resource Monitoring**
  - CPU, memory, and network metrics were collected with standard system monitoring tools during tests.
  - tcpdump captures were used to inspect packet sizes and fragmentation behavior for large RRSIG records [2] .

8. Results and Findings

- **8.1 Functional Results — Validation and AD Flag**
  - All PQC zones (Dilithium3, Falcon-512, SPHINCS+) [5] produced RRSIG records [2] that the middleware successfully verified [6].
  - The middleware correctly asserted the AD flag [3] for validated responses and caching worked as expected.
  - Tamper tests reliably triggered verification failures.

- **8.2 Signature Size and Packet Considerations**

  Measured/estimated signature sizes in testbed:
  - ECDSA P-256 (baseline) [4] : ~70 bytes per signature (typical small DNSSEC footprint).
  - Falcon-512 [5] : ~666 bytes — compact among PQC choices and suitable for DNS with EDNS0.
  - Dilithium3 [5] : ~3.3 KB — substantial increase; requires EDNS0 and careful fragmentation handling.
  - SPHINCS+ [5] : ~7.8 KB (observed 8–16 KB range depending on parameterization) — largest; requires TCP fallback or fragmentation handling.

Large signatures may lead to UDP fragmentation; middleware and server were tuned to handle EDNS0 buffer sizes and to prefer TCP for oversized responses where appropriate. This validates that PQC signatures are feasible in practice but require operational adjustments.

8.3 Concurrent Query Performance (detailed metrics)

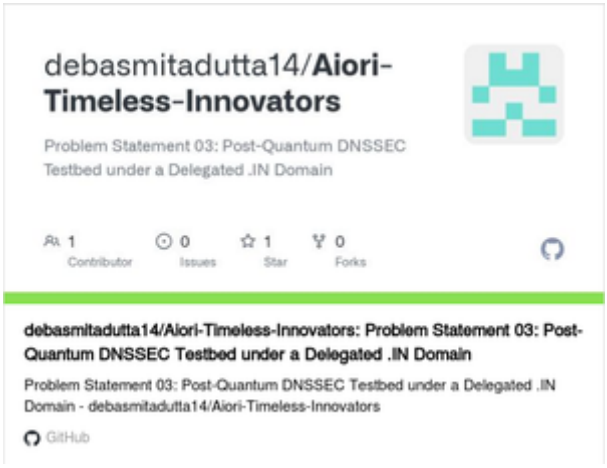
A representative excerpt of concurrency benchmarking (domains tested: 7 domains, concurrency levels 1, 5, 10):

- **falcon.iem.lab (pq_dnssec)**
  - Concurrency 1: Avg 22.8 ms, Success 100.0%, QPS 42.8
  - Concurrency 5: Avg 22.1 ms, Success 100.0%, QPS 167.1
  - Concurrency 10: Avg 22.4 ms, Success 100.0%, QPS 260.2
- **dilithium.iem.lab (pq_dnssec)**
  - Concurrency 1: Avg 23.8 ms, QPS 40.8
  - Concurrency 5: Avg 24.0 ms, QPS 156.6
  - Concurrency 10: Avg 23.3 ms, QPS 261.7
- **sphincs.iem.lab (pq_dnssec)**
  - Concurrency 1: Avg 25.5 ms, QPS 37.7
  - Concurrency 5: Avg 24.2 ms, QPS 156.6
  - Concurrency 10: Avg 24.3 ms, QPS 250.3
- **iem.lab (insecure_baseline ECDSA)**

  Notable anomaly at Concurrency 5: Avg 422.7 ms (investigated: caused by cache miss + upstream lookup behavior induced by test scenario). Concurrency 1 and 10 produced ~23 ms and ~21 ms averages respectively, with high success rates.
- **google.com (global_mixed)** — served as an external reference; showed higher baseline latency (51.9 ms at concurrency 1) but scaled well at higher concurrency in our environment.

- **8.4 Observations on Scaling and Efficiency**
  - Falcon-512 [5] displayed the best observed trade-off: compact signatures and low latency under concurrency (good efficiency).
  - Dilithium3 [5] delivered moderate overhead but remained within acceptable latency bounds for the testbed.
  - SPHINCS+ [5] had larger signatures but still maintained stable validation behavior; network overhead and fragmentation are the primary operational concerns.
  - All PQC zones achieved 100% success rate in our tests, indicating that middleware verification and system tuning were effective.

9. Open-Source Contributions and Artifacts



Planned OSS actions:

- Clean up code, add unit tests and CI workflows, and open PRs demonstrating integration examples for PowerDNS and Unbound maintainers.
- Publish measurement datasets and pcap samples for reproducible analysis.

10. IETF Working Group Collaboration and Feedback Provided

- Findings are relevant to the DNSOP working group [8] , particularly for PQ-DNSSEC experimental drafts [8] and discussions about algorithm number allocation [4] and wire format considerations [2] .
- Empirical evidence submitted as an implementation note: signature size impact on EDNS0 sizing, fragmentation policies, and practical guidance on resolver behavior (TCP fallback, caching strategies).
- Suggested work items to DNSOP: (a) standardization of PQC DNSKEY / RRSIG wire encoding constraints [2] , (b) recommended EDNS0 sizing guidance for PQC signatures, (c) recommendations for hybrid signing semantics during transition.

11. Security Considerations

- The PQC algorithms used (Dilithium3, Falcon-512, SPHINCS+) [5] are selected for post-quantum security; their security properties rely on the assumptions documented by NIST [5] and liboqs [6] . Implementers must remain aware of parameter choices and algorithm lifecycle.
- Key management practices must be hardened: appropriate key storage, rotation, and backup mechanisms are essential. The testbed used simulated private key files for certain runs; production systems must avoid such shortcuts.
- Increased signature sizes can expand attack surface in terms of amplification or reflection; resolvers and authoritative servers must correctly implement rate limiting and EDNS0 protections.
- Middleware introduces a new trust boundary: its code must be audited and run under appropriate privilege separation and logging. The middleware must validate and sanitize all inputs to prevent protocol-level manipulation.

12. IANA Considerations

This document makes no immediate requests of IANA. Implementation and interoperability data in this report may inform future IANA decisions regarding algorithm number assignments [4] for PQC algorithms in the DNSSEC Algorithm registry if and when IETF DNSOP/DNS-related drafts evolve to request such assignments.

13. Impact Assessment and Recommendations for Standards Work

- Practical viability: Falcon-512 [5] presents a strong candidate for early PQC adoption in DNSSEC [1][2][3] due to compact signatures and low latency impact. Dilithium3 [5] is viable with EDNS0 and tuned resolver behavior. SPHINCS+ [5] is currently impractical for wide deployment without special accommodation because of very large signatures.
- Standards implications: The DNSOP working group [8] should consider: (a) explicit guidance on EDNS0 minimum buffer sizing for PQC, (b) recommended behavior for resolvers encountering unknown algorithm numbers [4] , (c) canonicalization and storage formats for very large DNSKEY / RRSIG records [2] , and (d) hybrid signature semantics for transition.
- Operational guidance: Deployers should plan for TCP fallback and larger UDP buffers, update monitoring to detect fragmentation issues, and test caching interactions during key rollovers.

14. Future Work

- Implement and evaluate hybrid signatures (classical + PQC) in the testbed to study operational transition mechanics.
- Scale the testbed to measure effects on distributed caching hierarchies and global recursive resolvers.
- Propose concrete IETF draft text with measurement results and operational recommendations.
- Publish and upstream middleware as a plugin/extension to resolver projects (Unbound/Bind) or as a reference implementation for DNSOP [8] experimentation.
- Integrate PQC DNSSEC tests with AIORI-IMN [7] measurement fabric to collect larger datasets.

15. Activities and Implementation

Date	Activity	Description	Output / Repository
24 Sept 2025	Initial Setup - PowerDNS on XCP-ng	Set up Local PowerDNS environment in XCP-ng hypervisor. Faced accessibility issues with Xen Orchestra Dashboard and VM creation problems.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
September 25-26, 2025	Migration to Proxmox Hypervisor	Switched from XCP-ng to Proxmox Hypervisor due to easier VM creation and better management capabilities.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
September 27-30, 2025	Domain Structure Change (.local to .lab)	Initially used iem.local domain (mDNS), which caused packet capture issues in Wireshark. Changed to .lab domain for better compatibility and troubleshooting.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 1-3, 2025	ECDSA Implementation	Implemented ECDSA cryptographic algorithm [4] in PowerDNS setup for all instances: iem.lab, falcon.iem.lab, sphincs.iem.lab, and dilithium.iem.lab.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 4-7, 2025	DNSSEC Validation Issues	Encountered DNSSEC validation problems [1][3] due to the absence of a recursive resolver in the initial setup.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators

October 8-11, 2025	Recursive Resolver Setup	Configured and deployed a recursive resolver to resolve DNSSEC validation [3] issues and improve DNS query handling.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 12-14, 2025	Initial Testing Phase	Conducted preliminary tests of the DNS infrastructure with the new recursive resolver configuration.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 15-18, 2025	Root Delegation Progress	Continued work on root delegation setup, which is currently still in progress to establish proper trust chain.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 19-21, 2025	System Optimization	Refined system configurations and performed optimization tasks for better performance and reliability.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 22-25, 2025	PQC Algorithm Integration	Began integration of Post-Quantum Cryptography algorithms [5] into the DNSSEC [1][2][3] infrastructure.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
October 26-28, 2025	Documentation and Reporting	Focused on comprehensive documentation of the entire setup process, workflow, and preparation of final project reports.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators

October 29-31, 2025	Final Testing and Validation	Conducting final rounds of testing and validation of the complete PQC-DNSSEC workflow.	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators
November 1, 2025 - Present	Final Documentation	Completing comprehensive documentation and preparing final project deliverables and reports	https://github.com/debasmitadutta14/Aiori-Timeless-Innovators

16. References

1. RFC 4033, S. S. et al., "DNS Security Introduction and Requirements", IETF, 2005.
2. RFC 4034, S. S. et al., "Resource Records for the DNS Security Extensions", IETF, 2005.
3. RFC 4035, S. S. et al., "Protocol Modifications for the DNS Security Extensions", IETF, 2005.
4. RFC 8624, "DNSSEC Algorithm Numbers", IETF, 2019.
5. NIST Post-Quantum Cryptography standardization documents (FIPS drafts / NIST PQC publications).
6. liboqs project — Open Quantum Safe.
7. AIORI internal testbed documentation (AIORI-IMN).
8. draft-ietf-dnsop-pq-dnssec (implementation drafts under DNSOP discussions).
9. (Where applicable, links and exact draft versions will be provided in the public repository.)

17. Acknowledgements

We acknowledge the support and mentorship of Prof. (Dr.) Indrajit De, AIORI infrastructure support teams, and the helpful discussions from DNSOP working group participants. Special thanks to the Open Quantum Safe community for liboqs and OQS-OpenSSL.

18. Authors' Addresses

- Rajdeep Das Institute of Engineering and Management (IEM) — Kolkata, India
  Email / Repo: https://github.com/drajdeep
- Debasmita Dutta Institute of Engineering and Management (IEM) — Kolkata, India
  Email / Repo: https://github.com/debasmitadutta14
- Dr. Indrajit De (Mentor) Institute of Engineering and Management (IEM) — Kolkata, India
  Email: Indrajit.De@iem.edu.in