

CSCI485 Topics in Computer Science

MongoDB/NoSQL Project Guidelines

Semester: Fall 2025

There may be changes as needed, but you will be informed through VIU Announcements on VIULearn. Please make sure to check your account regularly.

Total points: 275

With Bonus points: 300

Contribution: 25%

Presenting your work is mandatory for grading of your project

All deliverables you will be asked before the final submission are mandatory for the final evaluation.

Project Overview

This project demonstrates your knowledge of MongoDB and NoSQL database concepts through the design, implementation, and deployment of a real-world application. You will work **individually** to create a complete solution that showcases advanced NoSQL database techniques.

NoSQL MongoDB Project Submission Guidelines & Rubrics

Submission Guidelines

Final Project Package Components

You will be using the MongoDB Community Edition server, which is accessible through campus computers.

Your final submission must include all components organized in a clear directory structure. Submit as a compressed file (ZIP or TAR) with the following organization:

ProjectName_StudentID/

├─ README.md

├─ database/

| └─ schemas/

| └─ sample_data/

| └─ indexes/

├─ queries/

| └─ basic_CRUD/

| └─ aggregations/

| └─ analysis/

├─ documentation/

| └─ project_report.pdf

| └─ database_design.pdf

├─ presentation/

| └─ slides.pdf

└─ reflection/

└─ learning_reflection.pdf

Submission Requirements Checklist

Database Implementation:

- ☐ MongoDB database exported (mongodump format)
- ☐ All 4+ collections with proper schema design
- ☐ 1000+ realistic sample records
- ☐ Schema validation rules implemented
- ☐ All relationship types demonstrated (1:1, 1:many, many:many)

Queries and Operations:

- ☐ Minimum 25 different queries documented
- ☐ CRUD operations for all collections
- ☐ 3+ complex aggregation pipelines
- ☐ Performance optimization with explain plans
- ☐ Text search implementation
- ☐ Database transactions implemented
- ☐ Minimum 5 strategic indexes
- ☐ Geospatial queries (if applicable)

Documentation:

- ☐ Database design document
- ☐ Query documentation
- ☐ Final project report (10-15 pages)

Presentation Materials:

- ☐ Presentation slides

Technical Standards

Code Quality:

- Use consistent naming conventions (camelCase for fields, descriptive collection names)
- Include comprehensive comments for complex aggregation pipelines
- Error handling in all scripts
- Follow MongoDB best practices for schema design

Data Quality:

- Realistic sample data relevant to chosen domain
 - Proper data types and formats
 - No duplicate or inconsistent records
 - Appropriate data distribution for testing queries
-

Assessment Rubrics

Database Implementation (20 points each)

Criteria	Excellent (16-20)	Good (12-15)	Satisfactory (9-11)	Needs Improvement (0-8)
Schema Design	Optimal document structure, proper embedding vs. referencing decisions, excellent field naming and organization	Good schema design with minor areas for improvement, mostly appropriate structural decisions	Adequate schema design, some suboptimal choices but functional	Poor schema design, inefficient structure, inappropriate for NoSQL
Collections & Relationships	4+ well-designed collections, all relationship types expertly implemented, clear data modeling rationale	4+ collections with good relationships, minor implementation issues	4 collections present, relationships implemented but could be optimized	Fewer than 4 collections or poorly implemented relationships
Data Volume & Quality	1000+ high-quality, realistic records, excellent data distribution and variety	1000+ good quality records with minor data quality issues	1000+ records present but some quality or realism issues	Insufficient records or poor data quality
Validation & Constraints	Comprehensive validation rules, excellent error handling, robust constraints	Good validation implementation with minor gaps	Basic validation present, some constraints missing	Minimal or ineffective validation
Indexing Strategy	5+ strategic indexes, excellent performance impact, well-documented rationale	5+ indexes with good performance benefits, mostly well-justified	5+ indexes present but some may be unnecessary or suboptimal	Fewer than 5 indexes or poorly planned indexing

MongoDB Operations & Queries (20 points each)

Criteria	Excellent (18-20)	Good (16-17)	Satisfactory (14-15)	Needs Improvement (0-13)
Query Variety & Complexity	25+ diverse, complex queries covering all CRUD operations and advanced scenarios	25+ good queries with solid variety and moderate complexity	25+ queries present but limited complexity or variety	Fewer than 25 queries or overly simple implementations
Performance Optimization	Excellent query optimization with detailed explain plans, significant performance improvements documented	Good optimization efforts with explain plans, some performance improvements	Basic optimization attempts with some explain plan analysis	Minimal optimization efforts, no performance analysis

Advanced MongoDB Features (25 points each)

Criteria	Excellent (23-25)	Good (20-22)	Satisfactory (17-19)	Needs Improvement (0-16)
Aggregation Pipelines	3+ complex, sophisticated pipelines solving real business problems, excellent use of operators	3+ good pipelines with solid complexity and business value	3+ pipelines present but relatively simple or limited business impact	Fewer than 3 pipelines or overly basic implementations
Text Search	Advanced text search with indexes, scoring, and complex queries	Good text search implementation with proper indexing	Basic text search functionality working	Minimal or non-functional text search
Advanced Operations	Excellent implementation of transactions and other advanced features	Good implementation of most advanced features with minor issues	Basic implementation of some advanced features	Limited or poor implementation of advanced features
Innovation & Complexity	Goes beyond requirements with creative solutions and advanced techniques	Demonstrates good understanding with some innovative approaches	Meets requirements with standard approaches	Minimal effort, basic implementations only

Documentation & Scripts (5 points)

Criteria	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (0-2)
Code Documentation	Comprehensive comments, excellent README files, clear instructions	Good documentation with minor gaps, mostly clear instructions	Adequate documentation, some areas unclear	Poor documentation, difficult to understand or follow

Presentation & Communication (10 points each)

Criteria	Excellent (9-10)	Good (8)	Satisfactory (7)	Needs Improvement (0-6)
Demonstration	Professional, comprehensive demo clearly showcasing all features and technical complexity	Good demo covering most features with clear explanations	Basic demo showing main functionality	Poor quality demo or missing key features
Q&A	Expert-level technical communication, excellent handling of Q&A, demonstrates deep understanding	Good technical presentation with solid Q&A responses	Adequate presentation with basic Q&A handling	Weak presentation, poor technical communication
Project Report Quality	Outstanding written communication, comprehensive analysis, excellent technical depth	Good report with solid technical content and clear writing	Adequate report meeting basic requirements	Poor report quality, unclear communication

Bonus Opportunities (+25 points)

- **Integration:** Develop working application interface (web app, API, etc.)
-

Common Pitfalls to Avoid

1. Anti-patterns in Schema Design:

- Treating MongoDB like a relational database
- Excessive normalization inappropriate for document databases
- Poor embedding vs. referencing decisions

2. Performance Issues:

- Missing indexes on frequently queried fields
- Inefficient aggregation pipelines
- Large result sets without proper pagination

3. Documentation Gaps:

- Unclear setup instructions
- Missing rationale for design decisions
- Inadequate query explanations

4. Data Quality Problems:

- Unrealistic or inconsistent sample data
- Insufficient data volume for meaningful analysis
- Missing edge cases in test data

Submission Deadline & Late Policy

Final Submission Deadline: November 22, 2025, 11:59 PM (EST)

Late Submission Policy:

- Up to 24 hours late: 10% deduction
- 24-48 hours late: 20% deduction
- 48-72 hours late: 30% deduction
- Beyond 72 hours: Contact instructor immediately

Technical Issues: Contact IT support and instructor immediately if experiencing submission difficulties. Screenshots and error messages must be provided.

Academic Integrity Reminder

- All work must be original and individually completed
- Properly cite any external resources, tutorials, or code snippets
- Collaboration on ideas is encouraged, but implementations must be individual
- Plagiarism detection software will be used on all submissions
- Suspected academic integrity violations will be reported according to university policy

Getting Help

Office Hours: Wednesday, 1:00-2:00 PM, Room 315/210C

Final Reminder: Start early, test thoroughly, and don't hesitate to ask questions. This project is designed to demonstrate your mastery of NoSQL concepts and MongoDB implementation skills.