Cấu trúc cơ sở dữ liệu MongoDB cho hệ thống quản lý công việc như sau:

## 1. Collection users

javascript

{ \_id: ObjectId, username: String, fullName: String, email: String, phoneNumber: String, passwordHash: String, isActive: Boolean, lastLogin: Date, loginAttempts: Number, lockUntil: Date, twoFactorEnabled: Boolean, position: String, directManager: ObjectId, *// Reference to another user* joinDate: Date, roles: [ObjectId], *// Reference to roles collection* specialPermissions: [ObjectId], *// Reference to permissions collection* customSettings: { notificationPreferences: { inApp: Boolean, email: Boolean, sms: Boolean, notificationType: { newTask: Boolean, taskUpdate: Boolean, deadline: Boolean, mentions: Boolean }, quietHours: { enabled: Boolean, start: String, *// Format: "HH:MM"* end: String, *// Format: "HH:MM"* excludeEmergency: Boolean } }, dashboardLayout: Object }, metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 2. Collection roles

javascript

{ \_id: ObjectId, name: String, *// "Ban lãnh đạo khoa", "Trưởng bộ môn", "Giảng viên", etc.* description: String, permissions: [ObjectId], *// Reference to permissions collection* isSystem: Boolean, *// True for system-defined roles* expirationPolicy: { hasExpiration: Boolean, defaultDuration: Number *// Days* }, metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId }}

## 3. Collection permissions

javascript

{ \_id: ObjectId, code: String, *// Unique code for this permission* name: String, description: String, module: String, *// Module this permission belongs to* actions: [String], *// "create", "read", "update", "delete", etc.* isSystem: Boolean, *// True for system-defined permissions* metadata: { createdAt: Date, updatedAt: Date }}

## 4. Collection departments

javascript

{ \_id: ObjectId, name: String, code: String, description: String, parentDepartment: ObjectId, *// Reference to parent department* manager: ObjectId, *// Reference to user collection* foundingDate: Date, displayOrder: Number, isActive: Boolean, path: String, *// Materialized path for hierarchical queries (e.g., "/parent/child/grandchild")* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 5. Collection userDepartments

javascript

{ \_id: ObjectId, userId: ObjectId, *// Reference to users collection* departmentId: ObjectId, *// Reference to departments collection* position: String, startDate: Date, endDate: Date, isActive: Boolean, notes: String, metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId }}

## 6. Collection tasks

javascript

{ \_id: ObjectId, title: String, description: String, objective: String, managingDepartment: ObjectId, *// Reference to departments collection* priority: String, *// "Khẩn cấp", "Cao", "Trung bình", "Thấp", "Lập kế hoạch"* status: String, *// "Nháp", "Chưa bắt đầu", "Đã phân công", "Chờ làm rõ", "Đang thực hiện", "Chờ phê duyệt", etc.* categories: [ObjectId], *// Reference to taskCategories collection* startDate: Date, dueDate: Date, deadlineType: String, *// "Cứng" or "Mềm"* completedDate: Date, progressPercentage: Number, *// 0-100* isRecurring: Boolean, parentTask: ObjectId, *// For subtasks, reference to parent task* assignedTo: { type: String, *// "user", "group", "department"* id: ObjectId *// Reference to relevant collection* }, supervisor: ObjectId, *// Reference to users collection* approver: ObjectId, *// Reference to users collection* requiresApproval: Boolean, attachments: [{ fileName: String, fileType: String, fileSize: Number, uploadDate: Date, uploadedBy: ObjectId, path: String }], relatedLinks: [String], tags: [String], budget: { amount: Number, currency: String }, requiredResources: [ObjectId], *// Reference to resources collection* recurringTaskSettings: { frequency: String, *// "daily", "weekly", "monthly", "quarterly"* interval: Number, *// Repeat every X days/weeks/months* daysOfWeek: [Number], *// 0-6 (Sunday to Saturday)* dayOfMonth: Number, *// 1-31* endDate: Date, occurrences: Number }, metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 7. Collection taskCategories

javascript

{ \_id: ObjectId, name: String, description: String, parentCategory: ObjectId, *// Reference to parent category* color: String, *// Hex code for color* displayOrder: Number, path: String, *// Materialized path for hierarchical queries* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId }}

## 8. Collection subTasks

javascript

{ \_id: ObjectId, parentTaskId: ObjectId, *// Reference to tasks collection* title: String, description: String, assignedTo: ObjectId, *// Reference to users collection* startDate: Date, dueDate: Date, status: String, priority: String, progressPercentage: Number, *// 0-100* dependsOn: [ObjectId], *// References to other subTasks* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId }}

## 9. Collection taskAssignments

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* assigneeType: String, *// "user", "group", "department"* assigneeId: ObjectId, *// Reference to the relevant collection* assignedBy: ObjectId, *// Reference to users collection* assignedDate: Date, instructions: String, estimatedTime: Number, *// Hours/days* priority: String, attachments: [{ fileName: String, fileType: String, fileSize: Number, uploadDate: Date, path: String }], status: String, *// "Pending", "Accepted", "Rejected", "Reassigned"* responseDate: Date, responseNotes: String, metadata: { createdAt: Date, updatedAt: Date }}

## Collection taskProgress

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* reportedBy: ObjectId, *// Reference to users collection* reportDate: Date, progressPercentage: Number, *// 0-100* description: String, attachments: [{ fileName: String, fileType: String, fileSize: Number, uploadDate: Date, path: String }], issues: String, suggestions: String, metadata: { createdAt: Date }}

## 11. Collection taskFeedback

javascript

{ \_id: ObjectId, taskProgressId: ObjectId, *// Reference to taskProgress collection* providedBy: ObjectId, *// Reference to users collection* feedbackDate: Date, content: String, requestAdjustments: Boolean, adjustmentDetails: String, metadata: { createdAt: Date }}

## 12. Collection extensionRequests

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* requestedBy: ObjectId, *// Reference to users collection* requestDate: Date, currentDueDate: Date, requestedDueDate: Date, reason: String, currentProgress: Number, *// 0-100* implementationPlan: String, status: String, *// "Pending", "Approved", "Rejected"* approvedBy: ObjectId, *// Reference to users collection* approvalDate: Date, approvalNotes: String, metadata: { createdAt: Date, updatedAt: Date }}

## 13. Collection evaluations

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* evaluatedBy: ObjectId, *// Reference to users collection* evaluationDate: Date, rating: String, *// "Không hoàn thành", "Hoàn thành yếu", "Hoàn thành", etc.* comments: String, strengths: String, areasForImprovement: String, suggestions: String, isGroupWork: Boolean, groupEvaluations: [{ memberId: ObjectId, *// Reference to users collection* contribution: Number, *// Percentage* individualRating: String, comments: String }], hasBeenUpdated: Boolean, previousEvaluation: { rating: String, comments: String, updateReason: String }, metadata: { createdAt: Date, updatedAt: Date }}

## 14. Collection evaluationFeedback

javascript

{ \_id: ObjectId, evaluationId: ObjectId, *// Reference to evaluations collection* providedBy: ObjectId, *// Reference to users collection* feedbackDate: Date, content: String, requestReview: Boolean, reviewReason: String, status: String, *// "Pending", "Acknowledged", "Adjusted", "Escalated"* resolvedBy: ObjectId, *// Reference to users collection* resolutionDate: Date, resolutionNotes: String, metadata: { createdAt: Date, updatedAt: Date }}

## 15. Collection resources

javascript

{ \_id: ObjectId, name: String, code: String, type: String, *// "Room", "Equipment", "Software"* location: String, capacity: Number, *// For rooms* specifications: String, *// Technical specifications* manager: ObjectId, *// Reference to users collection* status: String, *// "Available", "In Use", "Maintenance", "Retired"* rules: String, *// Usage rules* acquisitionDate: Date, maintenanceSchedule: [{ lastMaintenance: Date, nextMaintenance: Date, maintenanceType: String, notes: String }], images: [{ fileName: String, path: String, uploadDate: Date }], attachments: [{ fileName: String, fileType: String, fileSize: Number, description: String, path: String }], *// Specific fields based on resource type* roomDetails: { building: String, floor: String, roomNumber: String, equipment: [String] }, equipmentDetails: { model: String, serialNumber: String, manufacturer: String, purchaseDate: Date, warrantyExpiration: Date }, softwareDetails: { version: String, developer: String, licenseType: String, licenseCount: Number, expirationDate: Date, systemRequirements: String }, metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 16. Collection resourceBookings

javascript

{ \_id: ObjectId, resourceId: ObjectId, *// Reference to resources collection* requestedBy: ObjectId, *// Reference to users collection* startTime: Date, endTime: Date, purpose: String, relatedTaskId: ObjectId, *// Reference to tasks collection (optional)* participants: [ObjectId], *// References to users collection* specialRequirements: String, status: String, *// "Pending", "Approved", "Rejected", "Completed", "Cancelled"* approvedBy: ObjectId, *// Reference to users collection* approvalDate: Date, rejectionReason: String, actualStartTime: Date, actualEndTime: Date, resourceCondition: { reported: Boolean, status: String, *// "Normal", "Minor Issues", "Major Issues"* issues: String, maintenanceRequired: Boolean }, metadata: { createdAt: Date, updatedAt: Date }}

## 17. Collection maintenanceRequests

javascript

{ \_id: ObjectId, resourceId: ObjectId, *// Reference to resources collection* requestedBy: ObjectId, *// Reference to users collection* requestDate: Date, issue: String, priority: String, *// "Low", "Medium", "High", "Critical"* status: String, *// "Pending", "In Progress", "Completed", "Cancelled"* assignedTo: ObjectId, *// Reference to users collection* startDate: Date, completionDate: Date, resolutionNotes: String, metadata: { createdAt: Date, updatedAt: Date }}

## 18. Collection groups

javascript

{ \_id: ObjectId, name: String, code: String, description: String, type: String, *// "Project", "Research", "Teaching", "Administrative"* managingDepartment: ObjectId, *// Reference to departments collection* startDate: Date, endDate: Date, leader: ObjectId, *// Reference to users collection* size: Number, structure: String, *// "Flat", "Hierarchical", "Matrix"* workingRules: String, status: String, *// "Setting Up", "Active", "Completed", "Archived"* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 19. Collection groupMembers

javascript

{ \_id: ObjectId, groupId: ObjectId, *// Reference to groups collection* userId: ObjectId, *// Reference to users collection* role: String, *// "Group Leader", "Secretary", "Member", "Advisor", etc.* responsibilities: String, joinDate: Date, endDate: Date, status: String, *// "Active", "Inactive", "Temporary Leave"* invitedBy: ObjectId, *// Reference to users collection* metadata: { createdAt: Date, updatedAt: Date }}

## 20. Collection documents

javascript

{ \_id: ObjectId, title: String, description: String, type: String, *// "Report", "Guide", "Form", "Data", etc.* tags: [String], owner: ObjectId, *// Reference to users collection* access: { type: String, *// "Public", "Department", "Group", "Private"* entityId: ObjectId *// Reference to departments/groups if applicable* }, accessRights: [{ entityType: String, *// "User", "Role", "Department", "Group"* entityId: ObjectId, permissionLevel: String *// "Read", "Edit", "Full Control"* }], fileName: String, fileType: String, fileSize: Number, path: String, uploadDate: Date, version: Number, versionHistory: [{ version: Number, updatedBy: ObjectId, updateDate: Date, changes: String, path: String }], relatedEntityType: String, *// "Task", "Group", "Resource", etc.* relatedEntityId: ObjectId, *// Reference to the related entity* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 21. Collection notifications

javascript

{ \_id: ObjectId, receiverId: ObjectId, *// Reference to users collection* type: String, *// "Task", "Resource", "Group", "System"* subtype: String, *// "Assignment", "Deadline", "Update", "Approval", etc.* title: String, content: String, priority: String, *// "Low", "Normal", "High", "Critical"* relatedEntityType: String, *// "Task", "Resource", "Group", etc.* relatedEntityId: ObjectId, *// Reference to the related entity* actionUrl: String, *// URL to navigate on click* isRead: Boolean, readAt: Date, expiresAt: Date, metadata: { createdAt: Date }}

## 22. Collection taskAssignmentHistory

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* previousAssignee: { type: String, *// "user", "group", "department"* id: ObjectId }, newAssignee: { type: String, *// "user", "group", "department"* id: ObjectId }, reassignedBy: ObjectId, *// Reference to users collection* reassignDate: Date, reason: String, metadata: { createdAt: Date }}

## 23. Collection taskHistory

javascript

{ \_id: ObjectId, taskId: ObjectId, *// Reference to tasks collection* fieldChanged: String, *// "status", "priority", "dueDate", etc.* oldValue: Mixed, newValue: Mixed, changedBy: ObjectId, *// Reference to users collection* changeDate: Date, reason: String, metadata: { createdAt: Date }}

## 24. Collection recurringTaskTemplates

javascript

{ \_id: ObjectId, title: String, description: String, objective: String, managingDepartment: ObjectId, priority: String, categories: [ObjectId], assigneeType: String, assigneeId: ObjectId, supervisor: ObjectId, approver: ObjectId, requiresApproval: Boolean, estimatedDuration: Number, *// Days* recurringPattern: { frequency: String, *// "daily", "weekly", "monthly", "quarterly"* interval: Number, daysOfWeek: [Number], dayOfMonth: Number, monthsOfYear: [Number], startDate: Date, endDate: Date, occurrences: Number }, nextOccurrence: Date, currentTask: ObjectId, *// Reference to the current active task* metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId }}

## 25. Collection reports

javascript

{ \_id: ObjectId, name: String, description: String, type: String, *// "Task", "Performance", "Resource", "Evaluation", "Combined"* parameters: { timeRange: { start: Date, end: Date }, entityType: String, *// "Department", "User", "Group"* entityId: ObjectId, categories: [ObjectId], statuses: [String], metrics: [String] }, displayFormat: String, *// "Table", "BarChart", "LineChart", "PieChart", "Combined", "Dashboard"* displayOptions: { colors: [String], labels: [String], legend: Boolean, sortBy: String, orderBy: String *// "asc", "desc"* }, isTemplate: Boolean, isScheduled: Boolean, schedule: { frequency: String, *// "daily", "weekly", "monthly"* dayOfWeek: Number, dayOfMonth: Number, time: String, recipients: [ObjectId], lastRun: Date, nextRun: Date }, sharedWith: [{ entityType: String, *// "User", "Role", "Department"* entityId: ObjectId, permissionLevel: String *// "View", "Edit"* }], metadata: { createdAt: Date, updatedAt: Date, createdBy: ObjectId, updatedBy: ObjectId }}

## 26. Collection reportData

javascript

{ \_id: ObjectId, reportId: ObjectId, *// Reference to reports collection* generatedAt: Date, generatedBy: ObjectId, *// Reference to users collection* parameters: Object, *// Copy of the parameters used* data: Object, *// The actual report data* metadata: { createdAt: Date }}

## 27. Collection groupEvaluations

javascript

{ \_id: ObjectId, groupId: ObjectId, *// Reference to groups collection* evaluationType: String, *// "Periodic", "Midterm", "Final"* evaluationPeriod: { start: Date, end: Date }, evaluatedBy: ObjectId, *// Reference to users collection* evaluationDate: Date, criteria: [{ name: String, description: String, weight: Number, score: Number, comments: String }], overallComments: String, strengths: String, improvementAreas: String, recommendations: String, individualEvaluations: [{ memberId: ObjectId, *// Reference to users collection* scores: [{ criterionName: String, score: Number, comments: String }], overallScore: Number, comments: String }], metadata: { createdAt: Date, updatedAt: Date }}

## Indexes cần thiết

javascript

*// users collection*db.users.createIndex({ "username": 1 }, { unique: true })db.users.createIndex({ "email": 1 }, { unique: true })db.users.createIndex({ "roles": 1 })*// departments collection*db.departments.createIndex({ "parentDepartment": 1 })db.departments.createIndex({ "path": 1 })db.departments.createIndex({ "code": 1 }, { unique: true })*// userDepartments collection*db.userDepartments.createIndex({ "userId": 1, "departmentId": 1 }, { unique: true })db.userDepartments.createIndex({ "departmentId": 1 })*// tasks collection*db.tasks.createIndex({ "title": "text", "description": "text" })db.tasks.createIndex({ "status": 1 })db.tasks.createIndex({ "dueDate": 1 })db.tasks.createIndex({ "assignedTo.id": 1, "assignedTo.type": 1 })db.tasks.createIndex({ "supervisor": 1 })db.tasks.createIndex({ "managingDepartment": 1 })db.tasks.createIndex({ "categories": 1 })*// taskCategories collection*db.taskCategories.createIndex({ "parentCategory": 1 })db.taskCategories.createIndex({ "path": 1 })*// taskAssignments collection*db.taskAssignments.createIndex({ "taskId": 1 })db.taskAssignments.createIndex({ "assigneeId": 1, "assigneeType": 1 })*// taskProgress collection*db.taskProgress.createIndex({ "taskId": 1 })*// resources collection*db.resources.createIndex({ "name": "text", "code": "text", "type": 1 })db.resources.createIndex({ "status": 1 })*// resourceBookings collection*db.resourceBookings.createIndex({ "resourceId": 1 })db.resourceBookings.createIndex({ "startTime": 1, "endTime": 1 })db.resourceBookings.createIndex({ "requestedBy": 1 })*// groups collection*db.groups.createIndex({ "name": "text", "description": "text" })db.groups.createIndex({ "leader": 1 })db.groups.createIndex({ "managingDepartment": 1 })*// groupMembers collection*db.groupMembers.createIndex({ "groupId": 1 })db.groupMembers.createIndex({ "userId": 1 })db.groupMembers.createIndex({ "groupId": 1, "userId": 1 }, { unique: true })*// documents collection*db.documents.createIndex({ "title": "text", "description": "text", "tags": "text" })db.documents.createIndex({ "relatedEntityType": 1, "relatedEntityId": 1 })db.documents.createIndex({ "owner": 1 })*// notifications collection*db.notifications.createIndex({ "receiverId": 1 })db.notifications.createIndex({ "receiverId": 1, "isRead": 1 })db.notifications.createIndex({ "receiverId": 1, "createdAt": -1 })