**Users Collection - Finalized Attributes**

{

"\_id": "ObjectId (Primary Key)",

"name": "String",

"email": "String (Unique, acts as login ID)",

"password": "String (Hashed)",

"type": "String (Enum: 'student', 'teacher', 'admin')",

"gender": "String (Enum: 'male', 'female', 'other')",

"created\_at": "ISODate (Timestamp when the user was created)",

"updated\_at": "ISODate (Timestamp when the user last updated their details)"

}

### ****Users Collection - Explanation & Considerations****

#### ****1️⃣**** \_id ****(Primary Key)****

* **Type**: ObjectId
* **Purpose**: Unique identifier for each user in the Users collection.
* **Why Required?**
  + Auto-generated by MongoDB.
  + Ensures each user has a unique ID for authentication and record-keeping.
  + Used as a **foreign key** in other collections (e.g., Students, Teachers).

#### ****2️⃣**** name ****(Full Name of the User)****

* **Type**: String
* **Purpose**: Stores the user's full name.
* **Why Required?**
  + Essential for personalization and user identification in the system.
  + Used for **displaying names** in reports, dashboards, and user interactions.

#### ****3️⃣**** email ****(Login ID, Unique)****

* **Type**: String (Unique)
* **Purpose**: Stores the **user's email**, which also acts as their **login ID**.
* **Why Required?**
  + Ensures each user has a unique identifier for logging in.
  + Used for **authentication and password recovery**.
  + Email-based authentication is **more secure and scalable**.

#### ****4️⃣**** password ****(Hashed Password)****

* **Type**: String
* **Purpose**: Stores the **hashed password** for authentication.
* **Why Required?**
  + Essential for **secure authentication**.
  + Passwords are stored in a **hashed format** to prevent security risks.
  + Users will authenticate using their email and password.

#### ****5️⃣**** type ****(User Role)****

* **Type**: String (Enum: 'student', 'teacher', 'admin')
* **Purpose**: Defines the **role of the user** in the system.
* **Why Required?**
  + Determines **what permissions and features the user has**.
  + Values:
    - 'student' → Can attempt quizzes, track progress.
    - 'teacher' → Can manage students, assign tasks.
    - 'admin' → Has full system access, can manage users.

#### ****6️⃣**** gender ****(User's Gender)****

* **Type**: String (Enum: 'male', 'female', 'other')
* **Purpose**: Stores the user's gender for profiling.
* **Why Optional?**
  + Some users might **prefer not to disclose** their gender.
  + Helps in generating **personalized reports** and analytics.

#### ****7️⃣**** created\_at ****(Timestamp of Account Creation)****

* **Type**: ISODate
* **Purpose**: Stores the date and time when the user account was first created.
* **Why Required?**
  + Helps in **tracking user registrations**.
  + Useful for **auditing, reporting, and analytics**.

#### ****8️⃣**** updated\_at ****(Timestamp of Last Update)****

* **Type**: ISODate
* **Purpose**: Stores the date and time when the user last modified their details.
* **Why Required?**
  + Useful for tracking **account updates**.
  + Helps in **identifying stale accounts** (e.g., accounts that haven't been updated for long).

**Students Collection - Finalized Attributes**

{

"\_id": "ObjectId (Primary Key - )",

"user\_id": "ObjectId (Reference to Users.\_id)",

"grade": "Integer (Student's grade level)",

"school": "String (School name, optional)",

"teacher\_id": "ObjectId (Reference to Users.\_id, only if assigned to a teacher)",

"created\_at": "ISODate (Timestamp when student record was created)",

"updated\_at": "ISODate (Timestamp when student record was last updated)"

}

### ****Students Collection - Final Explanation & Considerations****

#### ****1️⃣**** \_id ****(Primary Key)****

* **Type**: ObjectId
* **Purpose**: Unique identifier for each student in the Students collection.
* **Why Required?**
  + Each student entry needs a unique identifier to distinguish records in MongoDB.
  + Auto-generated by MongoDB.

#### ****2️⃣**** user\_id ****(Reference to**** Users.\_id****)****

* **Type**: ObjectId (Foreign Key)
* **Purpose**: Links each student record to a corresponding **user account** in the Users collection.
* **Why Required?**
  + Every student **must** have an account in the Users collection to log in and access the system.
  + Ensures **data consistency** between users and students.
* **Usage**:
  + When fetching a student's profile, we can retrieve additional details from Users using user\_id.

#### ****3️⃣**** grade ****(Student’s Grade Level)****

* **Type**: Integer
* **Purpose**: Stores the student's grade level.
* **Why Required?**
  + Helps categorize students based on their academic level.
  + Used for **adaptive learning logic**, ensuring students get content tailored to their grade.

#### ****4️⃣**** school ****(School Name)****

* **Type**: String (Optional)
* **Purpose**: Stores the name of the school the student attends.
* **Why Optional?**
  + Not all students may be enrolled in a formal school (e.g., homeschooled students).
  + Some institutions may not require school details.

#### ****5️⃣**** teacher\_id ****(Reference to**** Users.\_id****, Only if Assigned to a Teacher)****

* **Type**: ObjectId (Foreign Key)
* **Purpose**: Links the student to a **specific teacher** in the Users collection.
* **Why Optional?**
  + Not all students are assigned a dedicated teacher (self-learning students).
  + Some students may be assigned to **multiple teachers**, and a different structure might be required in the future.
* **Usage**:
  + Used to fetch the assigned teacher’s details from the Users collection.

#### ****6️⃣**** created\_at ****(Timestamp of Record Creation)****

* **Type**: ISODate
* **Purpose**: Stores the date and time when the student record was first created.
* **Why Required?**
  + Helps track when a student joined the system.
  + Useful for **auditing, reporting, and analytics**.

#### ****7️⃣**** updated\_at ****(Timestamp of Last Update)****

* **Type**: ISODate
* **Purpose**: Stores the date and time when the student record was last modified.
* **Why Required?**
  + Useful for tracking modifications in student data.
  + Helps in managing **recent updates and progress tracking**.

**Questions Collection - Schema & Explanations**

{

"\_id": "ObjectId (Auto-generated MongoDB ID)",

"description": "String (Text of the question)",

"type": "String (Enum: 'SCQ', 'MCQ')",

"options": [

{

"option\_id": "Integer (Unique within the question)",

"text": "String (Option text)",

"is\_correct": "Boolean (true if correct, false otherwise)"

}

],

"solution": "String (Detailed explanation of the solution)",

"difficulty": "String (Enum: 'VERY\_EASY', 'EASY', 'MEDIUM', 'HARD', 'VERY\_HARD', 'UNMARKED')",

"topic": "String (Main topic of the question, optional)",

"sub\_topic": "String (Specific section under the topic)",

"prerequisites": "Array of Strings (topics to know upfront to solve the current question)",

"topic\_id": "String (Unique identifier for the topic, used for categorization)"

}

### ****Explanations & Considerations****

#### ****1️⃣**** \_id ****(Primary Key)****

* **Type**: ObjectId (Auto-generated by MongoDB)
* **Purpose**: Unique identifier for each question.
* **Why Required?**
  + Ensures each question has a unique ID.
  + Used for referencing questions in **assessments, reports, and analytics**.

#### ****2️⃣**** description ****(Question Text)****

* **Type**: String
* **Purpose**: Stores the actual question text.
* **Why Required?**
  + Essential for displaying the question.
  + Used by students to attempt the problem.

#### ****3️⃣**** type ****(Question Type)****

* **Type**: String (Enum: 'SCQ', 'MCQ')
* **Purpose**: Defines the **format of the question**.
* **Why Required?**
  + Helps in structuring how students interact with the question.
  + Enum values:
    - 'SCQ' → **Single Correct Choice**
    - 'MCQ' → **Multiple Correct Choices**

#### ****4️⃣**** options ****(Answer Choices)****

* **Type**: Array of Objects
* **Purpose**: Stores the possible answer choices for the question.
* **Why Required?**
  + Each question needs **multiple answer choices**.
  + Structure:
    - option\_id → Unique within the question.
    - text → Stores the answer choice.
    - is\_correct → Boolean (true if correct, false otherwise)

"options": [

{"option\_id": 1, "text": "Option A", "is\_correct": false},

{"option\_id": 2, "text": "Option B", "is\_correct": true},

{"option\_id": 3, "text": "Option C", "is\_correct": false},

{"option\_id": 4, "text": "Option D", "is\_correct": false}

]

#### ****5️⃣**** solution ****(Explanation of the Answer)****

* **Type**: String
* **Purpose**: Stores the detailed explanation of the correct answer.
* **Why Required?**
  + Helps students understand **why the correct answer is correct**.
  + Useful for **hints, explanations, and learning analytics**.

#### ****6️⃣**** difficulty ****(Difficulty Level)****

* **Type**: String (Enum: 'VERY\_EASY', 'EASY', 'MEDIUM', 'HARD', 'VERY\_HARD', 'UNMARKED')
* **Purpose**: Classifies the question’s difficulty level.
* **Why Required?**
  + Used for **adaptive learning**, ensuring students get questions at the right level.
  + Enum values:
    - 'VERY\_EASY' → Beginner-level.
    - 'EASY' → Basic understanding required.
    - 'MEDIUM' → Standard difficulty.
    - 'HARD' → Advanced problem-solving needed.
    - 'VERY\_HARD' → High-level competition-level problems.
    - 'UNMARKED' → Not yet classified.

#### ****7️⃣**** topic ****(Main Topic of the Question)****

* **Type**: String (Optional)
* **Purpose**: Represents the **main subject area**.
* **Why Optional?**
  + Some questions may not be linked to a **specific topic**.
  + Example values: "Algebra", "Geometry", "Trigonometry".

#### ****8️⃣**** sub\_topic ****(Specific Section Under the Topic)****

* **Type**: String
* **Purpose**: Represents the **specific concept or section** under the main topic.
* **Why Required?**
  + Helps in **fine-grained classification** of questions.
  + Example: If the topic is "Algebra", sub\_topic could be "Quadratic Equations".

#### ****9️⃣**** prerequisites ****(Question Labels for Filtering)****

* **Type**: Array of Strings
* **Purpose**: prerequisites topics for the current question.

#### ****🔟**** topic\_id ****(Unique Topic Identifier)****

* **Type**: String
* **Purpose**: Stores the unique **ID** for the associated topic.
* **Why Required?**
  + Helps in efficiently **linking questions to topics** in other collections.
  + Can be used for **database queries and indexing**.

**AssignmentInstance Collection - Schema & Explanations**

### ****Schema:****

{

"\_id": "ObjectId (Primary Key - Auto-generated)",

"teacher\_id": "ObjectId (Reference to Users.\_id - Teacher)",

"assignment\_id": "ObjectId (Reference to Assignments.\_id)",

"classes\_assigned\_to": ["ObjectId (References to Class.\_id)"],

"students\_assigned\_to": ["ObjectId (References to Users.\_id - Student)"],

"start\_date": "ISODate (Start date for the assignment)",

"end\_date": "ISODate (Due date for the assignment)"

}

### ****Explanations & Considerations****

#### ****1️⃣**** \_id ****(Primary Key)****

* **Type**: ObjectId (Auto-generated by MongoDB)
* **Purpose**: Unique identifier for each **assignment instance**.
* **Why Required?**
  + Each **assignment instance** needs a unique identifier for tracking.
  + Helps in managing **multiple versions** of the same assignment for different classes/students.

#### ****2️⃣**** teacher\_id ****(Reference to the Teacher Who Assigned the Homework)****

* **Type**: ObjectId (Reference to Users.\_id)
* **Purpose**: Links the assignment to the teacher who created it.
* **Why Required?**
  + Ensures the **correct teacher** is responsible for the assignment.
  + Used for **reporting and tracking assignments by teacher**.

#### ****3️⃣**** assignment\_id ****(Reference to the Assignment Template)****

* **Type**: ObjectId (Reference to Assignments.\_id)
* **Purpose**: Links this instance to the **base assignment**.
* **Why Required?**
  + The **actual assignment** (questions, description, etc.) is stored in a **separate Assignments collection**.
  + Allows **reusability** of assignments across multiple students/classes.

#### ****4️⃣**** classes\_assigned\_to ****(List of Classes This Assignment is Assigned To)****

* **Type**: Array of ObjectId (References to Class.\_id)
* **Purpose**: Specifies which **entire classes** have been assigned this assignment.
* **Why Required?**
  + Allows **bulk assignment** to entire classes.
  + If a student is part of a class listed here, they **automatically receive the assignment**.

#### ****5️⃣**** students\_assigned\_to ****(List of Individual Students Assigned This Assignment)****

* **Type**: Array of ObjectId (References to Users.\_id - Student)
* **Purpose**: Specifies **individual students** who were manually assigned this assignment.
* **Why Required?**
  + This field **overrides** classes\_assigned\_to for students who need **individual assignments**.
  + Useful for:
    - **Special cases** (extra practice, remedial work).
    - **Custom assignments** outside class-based distribution.

#### ****6️⃣**** start\_date ****(Assignment Start Date)****

* **Type**: ISODate
* **Purpose**: Specifies when students can start working on the assignment.
* **Why Required?**
  + Controls **when** students gain access to the assignment.
  + Ensures **timed release** of homework.

#### ****7️⃣**** end\_date ****(Assignment Due Date)****

* **Type**: ISODate
* **Purpose**: Specifies the **deadline** for completing the assignment.
* **Why Required?**
  + Used for **submission tracking**.
  + Helps in **automatically locking** access after the deadline.

**StudentProgress Collection - Schema & Explanations**

{

"\_id": "ObjectId (Primary Key - Auto-generated)",

"student\_id": "ObjectId (Reference to Users.\_id - Student)",

"assignment\_instance\_id": "ObjectId (Reference to AssignmentInstance.\_id)",

"start\_timestamp": "ISODate (Timestamp when student started the assignment)",

"answers": [

{

"question\_id": "ObjectId (Reference to Questions.\_id)",

"selected\_answer": "String (A/B/C/D or 'SKIPPED')",

"submission\_timestamp": "ISODate (Timestamp when the answer was submitted)"

}

]

}

### ****Explanations & Considerations****

#### ****1️⃣**** \_id ****(Primary Key)****

* **Type**: ObjectId (Auto-generated by MongoDB)
* **Purpose**: Unique identifier for each student's progress record.
* **Why Required?**
  + Each record must have a unique ID to track student progress separately.

#### ****2️⃣**** student\_id ****(Reference to the Student)****

* **Type**: ObjectId (Reference to Users.\_id)
* **Purpose**: Links the progress record to a **specific student**.
* **Why Required?**
  + Identifies **who** is attempting the assignment.
  + Used for tracking **individual student performance**.

#### ****3️⃣**** assignment\_instance\_id ****(Reference to the Assigned Homework)****

* **Type**: ObjectId (Reference to AssignmentInstance.\_id)
* **Purpose**: Links the progress to a **specific assignment instance**.
* **Why Required?**
  + Ensures that **student progress is tied to the correct assignment**.
  + Allows teachers to monitor **which students are working on which assignments**.

#### ****4️⃣**** start\_timestamp ****(Timestamp When Student Started the Assignment)****

* **Type**: ISODate
* **Purpose**: Records the **exact time the student starts working** on the assignment.
* **Why Required?**
  + Helps track **engagement time**.
  + Useful for enforcing **time-based restrictions**.

#### ****5️⃣**** answers ****(Array of Student’s Responses)****

* **Type**: Array of Objects
* **Purpose**: Stores **all question attempts** within the assignment.
* **Why Required?**
  + Essential for **evaluating student performance**.
  + Helps track **which questions were answered/skipped**.

