1.1 Class Diagram:

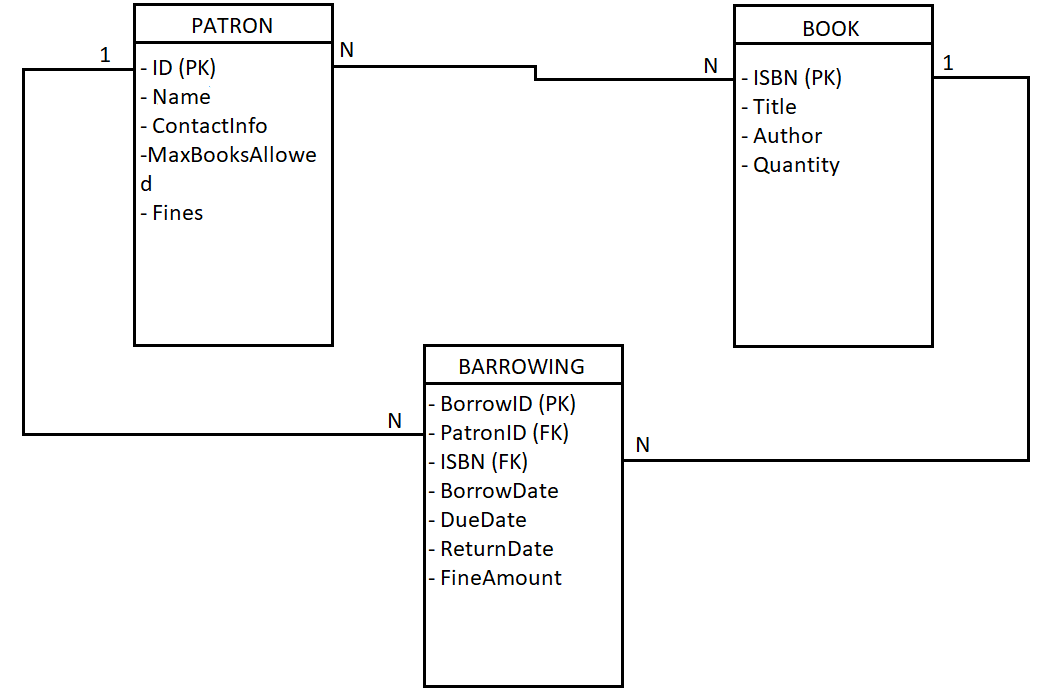
Create a class diagram showing the relationship between Book, Patron, and Borrowing. Include relevant attributes and relationships between classes.

O imagine care conține text, diagramă, captură de ecran, Paralel

Descriere generată automat

1.2 Database Diagram:

Design a database schema representing the entities Book, Patron, and Borrowing. Define the relationships between tables and attributes.



2.1 Logical Design:

• Devise a plan for organizing the questions, their structure, and the user's progress.

For organizing the questions I use an array of objects, each containing two properties. The first property, named question, is a string that represents the displayed question. The second property, named answers, is an array of objects that contains a string representing the text displayed and a boolean used to determine if the answer is correct or not.

For tracking the user progress we use a variable called score witch will be incremented by 2 every time a question will be answered correctly.

• Write a logical outline or pseudocode detailing how the system will generate and present questions while maintaining scores and ensuring no repetition.

For ensuring no repetition and presenting the questions for the user we will shuffle the array of questions when starting the quiz using a random number generator. For maintaining the score we will use a variable that will live in state.

2.3 Class and Database Representation (Explanation Only):

• Explain how you would represent the structure of questions, choices, and user progress in a class diagram and database schema.

Question

- id: int

- questionText: string

- answers: List<Answer>

Answer

- id: int

- answerText: string

- correct: bool

User

- id: int

- username: string

- progress: List<UserProgress>

UserProgress

- id: int

- user: User

- question: Question

- selectedAnswer: Answer

- correct: bool

Questions Table:

Columns:

- id (Primary Key)

- questionText

Answers Table:

Columns:

- id (Primary Key)

- answerText

- correct

- questionId (Foreign Key referencing Questions.id)

Users Table:

Columns:

- id (Primary Key)

- username

UserProgress Table:

Columns:

- id (Primary Key)

- userId (Foreign Key referencing Users.id)

- questionId (Foreign Key referencing Questions.id)

- selectedAnswerId (Foreign Key referencing Answers.id)

- correct

• Describe the relationships between entities, emphasizing the flow of data during the quiz.

1. User-Question Interaction:

Data Flow:

- When a user starts a quiz, the application selects a set of questions.

- The questions are presented to the user one at a time.

- The user interacts with the question by selecting an answer.

Relationships:

- A User can attempt multiple Questions.

- A Question can be attempted by multiple Users.

1. User-Answer Interaction:

Data Flow:

The user selects an answer for a given question.

Relationships:

A User can select multiple Answers.

An Answer can be selected by multiple Users (if the answer is correct for different users).

1. UserProgress Tracking:

Data Flow:

As the user progresses through the quiz, their choices and correctness are tracked.

The application records whether the selected answer is correct or not.

Relationships:

A UserProgress entry represents a user's interaction with a specific Question and the chosen Answer.

A User can have multiple UserProgress entries, each corresponding to a different question attempted.

1. Quiz Results and Scoring:

Data Flow:

At the end of the quiz, the application calculates and displays the user's score based on correct and incorrect answers.

Relationships:

The User's score is determined by calculating the correct and incorrect answers recorded in their UserProgress entries.

1. Question-Answer Relationship:

Data Flow:

The application presents a question along with its answer choices.

Relationships:

A Question has multiple Answers.

An Answer belongs to a specific Question.

Flow of Data during a Quiz Session:

The flow typically begins with selecting questions for a quiz session.

Questions are presented one at a time to the user.

The user selects an answer, and their choice is recorded in the UserProgress table.

The correctness of the answer is determined and stored in the same UserProgress entry.

This process repeats until all questions are answered.

At the end of the quiz, the application calculates the user's score based on their progress.