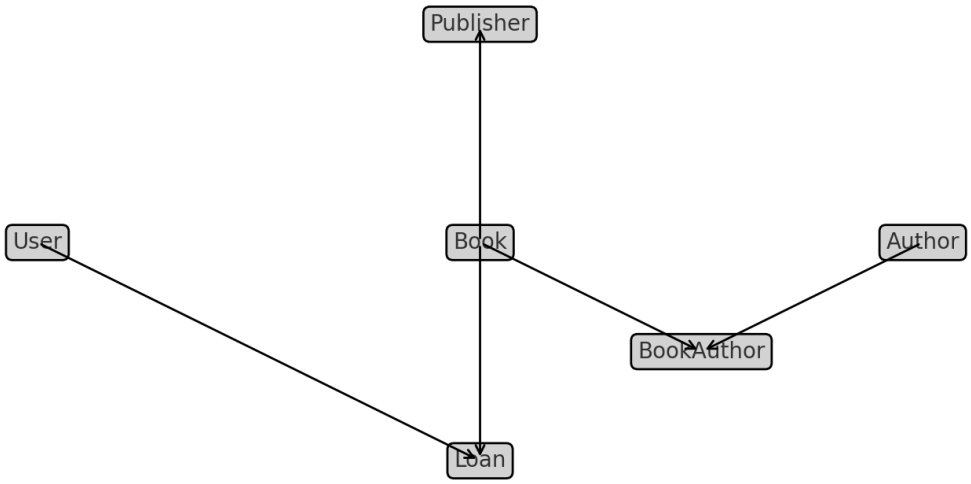
ER Model

The ER model for the Library Management System includes the following entity sets:
- User(userId, email, password_hash, firstName, lastName)
- Book(bookId, title, publicationYear, publisherId)
- Publisher(publisherId, name)
- Author(authorId, firstName, lastName)
- Loan(loanId, loanDate, returnDate, bookId, userId)
- BookAuthor(bookId, authorId)

Relationships:
- Each Book is published by one Publisher (Many-to-One).
- Each Book can have multiple Authors (Many-to-Many).
- Each User can loan multiple Books (One-to-Many).



Relations from ER Model

- User(userId [PK], email [U], password_hash, firstName, lastName)
- Book(bookId [PK], title, publicationYear, publisherId [FK])
- Publisher(publisherId [PK], name)
- Author(authorId [PK], firstName, lastName)
- Loan(loanId [PK], loanDate, returnDate, bookId [FK], userId [FK])
- BookAuthor(bookId [PK, FK], authorId [PK, FK])

Functional Dependencies

- userId determines email, password_hash, firstName, lastName
- email determines userId
- bookId determines title, publicationYear, publisherId
- publisherId determines name
- authorId determines firstName, lastName
- loanId determines loanDate, returnDate, bookId, userId
- bookId and authorId determine book-author associations

Normalization Steps

All relations are in 3NF or BCNF:
- All non-trivial functional dependencies have candidate keys on the left-hand side.
- No transitive or partial dependencies exist due to atomic primary keys.
- The many-to-many relationship BookAuthor is correctly decomposed into a separate relation with a composite key.

Final normalized relations:
- User(userId, email, password_hash, firstName, lastName)
- Book(bookId, title, publicationYear, publisherId)
- Publisher(publisherId, name)
- Author(authorId, firstName, lastName)
- Loan(loanId, loanDate, returnDate, bookId, userId)
- BookAuthor(bookId, authorId)