PROJECT DOCUMENTATION

DOMAIN: KNOWLEDGE MANAGEMENT PLATFORM FOR PROJECTS

1. INTRODUCTION

1.1 Domain Description and Relevance

The project centers around the development of a "Knowledge Management Platform for projects," a domain critical for modern project management and collaboration. In today's dynamic business environment, effective knowledge management is necessary for successful project execution. This platform addresses the complexities of managing project-related information, fostering collaboration among team members, and ensuring seamless access to critical project documents.

1.2 Background Study

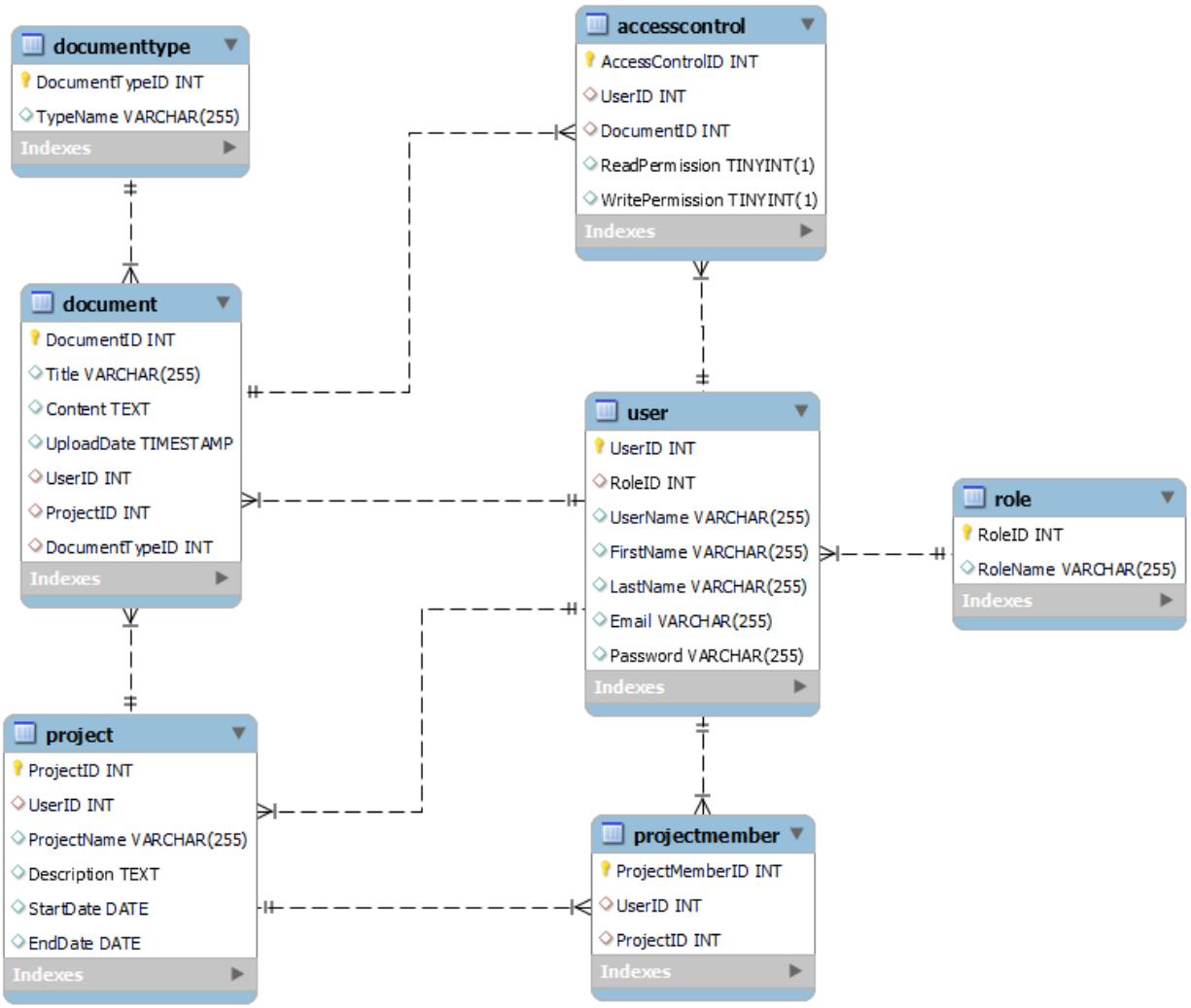
Normally projects knowledge management lack robust integration between project management and document collaboration features. In the project management industry, there is a growing emphasis on integrated solutions that combine document management with project collaboration tools. The findings underscored the need for a robust system that enables efficient document management, user collaboration, and secure access controls.

1.3 Requirement Analysis

- **User Roles and Permissions:**
    - User roles identified: Administrator, Project Manager, Team Member.
    - Permissions: Admins have full access, Project Managers can create/edit projects and documents, Team Members have read-only access and can be given write-access by the Manager.
- **Document Management:**
    - Document types include project plans, meeting minutes, design documents, and progress reports.
- **Access Control:**
    - Granular access control with role-based permissions for each document.

    - Specific access settings for confidential project documents.
- **Search and Retrieval:**
    - Advanced search capabilities based on document content, tags, and metadata.
    - Categorization and tagging for easy organization.
- **Project Management Integration:**
    - Integration with popular project management tools to synchronize tasks and milestones with project documents.
- **Usability and User Experience:**
    - User feedback emphasizes the importance of an intuitive interface with minimal learning curve.
    - A user-friendly design that supports both novice and experienced users.
- **Scalability:**
    - The platform should handle the growth of users and documents without performance degradation.

2. ENTITY-RELATIONSHIP DIAGRAM (ERD)



3. TABLE DETAILS

3.1 User Table

- **UserID (INT):** Unique identifier for each user.
- **RoleID (INT):** Foreign key referencing **Role.RoleID**.
- **UserName (VARCHAR):** User's username.
- **FirstName (VARCHAR):** User's first name.
- **LastName (VARCHAR):** User's last name.
- **Email (VARCHAR):** User's email address.
- **Password (VARCHAR):** User's password hash.

3.2 Role Table

- **RoleID (INT):** Unique identifier for each role.
- **RoleName (VARCHAR):** Role name.

3.3 DocumentType Table

- **DocumentTypeID (INT):** Unique identifier for each document type.
- **TypeName (VARCHAR):** Document type name.

3.4 Document Table

- **DocumentID (INT):** Unique identifier for each document.
- **Title (VARCHAR):** Document title.
- **Content (TEXT):** Document content.
- **UploadDate (TIMESTAMP):** Date and time of document upload.
- **UserID (INT):** Foreign key referencing **User.UserID**.
- **ProjectID (INT):** Foreign key referencing **Project.ProjectID**.
- **DocumentTypeID (INT):** Foreign key referencing **DocumentType.DocumentTypeID**.

3.5 Project Table

- **ProjectID (INT):** Unique identifier for each project.
- **UserID (INT):** Foreign key referencing **User.UserID**.
- **ProjectName (VARCHAR):** Project name.
- **Description (TEXT):** Project description.
- **StartDate (DATE):** Project start date.
- **EndDate (DATE):** Project end date.

3.6 ProjectMember Table

- **ProjectMemberID (INT):** Unique identifier for each project member.
- **UserID (INT):** Foreign key referencing **User.UserID**
- **ProjectID (INT):** Foreign key referencing **Project.ProjectID**.

3.7 AccessControl Table

- **AccessControlID (INT):** Unique identifier for each access control entry.
- **UserID (INT):** Foreign key referencing **User.UserID**.
- **DocumentID (INT):** Foreign key referencing **Document.DocumentID**.
- **ReadPermission (BOOLEAN):** Read permission status.
- **WritePermission (BOOLEAN):** Write permission status.

4. NORMALIZATION PROCESS

The normalization process involved organizing the data into well-structured tables to minimize redundancy and dependency issues, adhering to the principles of normalization up to the third normal form (3NF). This included:

- Ensuring atomicity of attributes within each table.
- Eliminating partial dependencies by establishing fully functional dependencies on primary keys.
- Removing transitive dependencies to ensure that non-key attributes depend only on the primary key.

By following these normalization principles, the Knowledge Management Platform for projects database is designed to efficiently manage data, support complex relationships between entities, and ensure data integrity and consistency.

CONCLUSION

The Knowledge Management Platform's database schema serves as a robust foundation for effective project collaboration and documentation. As the project progresses, future considerations may involve continuous refinement based on user feedback, emerging technologies, and evolving project management practices.

This documentation provides a comprehensive overview of the project's database design, laying the groundwork for successful implementation.