Codenames Game - Database Schema

1. Overview

This document outlines the structured database schema for the Codenames game, which is implemented using SQL Server.

The database is named `igroup181_test1`, and it stores user information, game data, moves, and real-time chat messages.

2. Database Tables

1. Users Table (Stores registered users)

```
CREATE TABLE Users (
    UserID INT PRIMARY KEY IDENTITY(1,1),
    Username NVARCHAR(50) UNIQUE NOT NULL,
    Email NVARCHAR(100) UNIQUE NOT NULL,
    PasswordHash NVARCHAR(255) NOT NULL,
    RegistrationDate DATETIME DEFAULT GETDATE()
);
```

2. Games Table (Stores game sessions)

```
CREATE TABLE Games (
    GameID INT PRIMARY KEY IDENTITY(1,1),
    CreatedBy INT FOREIGN KEY REFERENCES Users(UserID),
    CreationDate DATETIME DEFAULT GETDATE(),
    Status NVARCHAR(20) CHECK (Status IN ('Waiting', 'In Progress', 'Finished')),
    WinningTeam NVARCHAR(10) NULL
);
```

3. PlayersInGame Table (Links users to specific games)

```
CREATE TABLE PlayersInGame (
    GameID INT FOREIGN KEY REFERENCES Games(GameID),
    UserID INT FOREIGN KEY REFERENCES Users(UserID),
    Team NVARCHAR(10) CHECK (Team IN ('Red', 'Blue')),
    IsSpymaster BIT DEFAULT 0,
    PRIMARY KEY (GameID, UserID)
);
```

4. Words Table (Stores available words for game boards)

```
CREATE TABLE Words (
    WordID INT PRIMARY KEY IDENTITY(1,1),
    Word NVARCHAR(50) UNIQUE NOT NULL
);
```

5. Cards Table (Represents game cards assigned to words)

```
CREATE TABLE Cards (
    CardID INT PRIMARY KEY IDENTITY(1,1),
    GameID INT FOREIGN KEY REFERENCES Games(GameID),
    WordID INT FOREIGN KEY REFERENCES Words(WordID),
    Team NVARCHAR(10) CHECK (Team IN ('Red', 'Blue', 'Neutral', 'Assassin')),
    IsRevealed BIT DEFAULT 0
);
```

6. Moves Table (Tracks moves made by players)

```
CREATE TABLE Moves (
    MoveID INT PRIMARY KEY IDENTITY(1,1),
    GameID INT FOREIGN KEY REFERENCES Games(GameID),
    UserID INT FOREIGN KEY REFERENCES Users(UserID),
    CardID INT FOREIGN KEY REFERENCES Cards(CardID),
    MoveDate DATETIME DEFAULT GETDATE()
);
```

7. ChatMessages Table (Stores in-game chat messages)

```
CREATE TABLE ChatMessages (
    MessageID INT PRIMARY KEY IDENTITY(1,1),
    GameID INT FOREIGN KEY REFERENCES Games(GameID),
    UserID INT FOREIGN KEY REFERENCES Users(UserID),
    MessageText NVARCHAR(255) NOT NULL,
    Timestamp DATETIME DEFAULT GETDATE()
);
```

3. Database Relationships

```
ALTER TABLE Games ADD CONSTRAINT FK_Games_CreatedBy FOREIGN KEY (CreatedBy) REFERENCES Users(UserID);

ALTER TABLE PlayersInGame ADD CONSTRAINT FK_PlayersInGame_Game FOREIGN KEY (GameID) REFERENCES Games(GameID);
ALTER TABLE PlayersInGame ADD CONSTRAINT FK_PlayersInGame_User FOREIGN KEY (UserID) REFERENCES Users(UserID);
```

```sql
ALTER TABLE Cards ADD CONSTRAINT FK_Cards_Game FOREIGN KEY (GameID) REFERENCES Games(GameID);
ALTER TABLE Cards ADD CONSTRAINT FK_Cards_Word FOREIGN KEY (WordID) REFERENCES Words(WordID);

ALTER TABLE Moves ADD CONSTRAINT FK_Moves_Game FOREIGN KEY (GameID) REFERENCES Games(GameID);
ALTER TABLE Moves ADD CONSTRAINT FK_Moves_User FOREIGN KEY (UserID) REFERENCES Users(UserID);
ALTER TABLE Moves ADD CONSTRAINT FK_Moves_Card FOREIGN KEY (CardID) REFERENCES Cards(CardID);

ALTER TABLE ChatMessages ADD CONSTRAINT FK_ChatMessages_Game FOREIGN KEY (GameID) REFERENCES Games(GameID);
ALTER TABLE ChatMessages ADD CONSTRAINT FK_ChatMessages_User FOREIGN KEY (UserID) REFERENCES Users(UserID);
```

4. Sample Data for Testing

```sql
INSERT INTO Words (Word)
VALUES ('Tree'), ('Ocean'), ('Sky'), ('Car'), ('Dog'), ('Moon'), ('Sun'), ('Bridge'),
       ('Computer'), ('Book'), ('Phone'), ('Glasses'), ('Rocket'), ('Planet'),
       ('River'), ('Mountain'), ('House'), ('Lamp'), ('Fire'), ('Window'),
       ('Train'), ('Clock'), ('Castle'), ('Candle'), ('Sword');

SELECT TOP 25 Word FROM Words ORDER BY NEWID();
```

5. Next Steps

- Verify that all tables are correctly created and linked.

- Ensure sample data is inserted and retrieved properly.

- Proceed with implementing ASP.NET API services to interact with the database.