

MS2 Relational schema

- **Members**(email, user_password, preferredGenre)
- **Normal_Users** (email, first_name, last_name, birth_date, age)

Where Normal_Users.Age = CurrentDate - NormalUsers.birth_date

Normal_Users.Email references Members

- **Verified_Reviewers**(email, years_of_experience, first_name, last_name, isAccepted)

Verified_Reviewers.email references Members

- **Development_Teams**(email , team_name, company_name, formation_date, isAccepted)

Where Development_Teams.email references Members

- **Gaming_Conferences**(game_conference_id , conference_end_date, conference_start_date, duration, venue, conferene_name)

Where duration = Gaming_Conferences.conference_end_date -

Gaming_Conferences.conference_start_date

- **Games**(game_id, game_name, release_date, age_limit, interactivity_rating, uniqueness_rating, graphics_rating, level_design_rating, total_average_rating, conference_releasesd_id)

game_conference_released_id references Gaming_Conferences

- **Ratings**(rating_id, interactivity, uniqueness, level_design, graphics, game_id, member_email)

----- -----

where Ratings.game_id references Games

- **Game_Videos**(game_id, video_path)

Game_Videos.game_id references Games

- **Game_Screenshots(game_id, video_path)**

GameScreenshots.game_id references Games

- **Strategy_Games(game_id, is_real_time)**

Where Strategy.game_id references Games

- **Action_Games(game_id, sub_genre)**

Where Action_Games.game_id references Games

- **Sport_Games(game_id, sports_type)**

Where Sport_Games.game_id references Games

- **RPG_Games(game_id, is_pvp, has_story_line)**

Where RPG_Games.game_id references Games

- **Communities(community_id, theme, community_description, community_name, is_approved, owner_email)**

Owner_email references Normal_Users

- **Topics(topic_id, title, descriptive_text, community_id, member_email)**

member_email references Members

community_id references Communities

- **Gaming_Conference_Reviews(game_conference_review_id, game_conference_review_text, member_email, conference_id)**

Gaming_Conference_Reviews.member_email references Members

Gaming_Conference_Reviews.conference_id references Gaming_Conferences

- **Comments(comment_id, comment_text, member_email)**

Comments.member_email references Members

- **Reviews(review_id, review_text, verified_reviewer_email, game_id)**

Reviews.game_id references Games

Reviews.verified_reviewer_email references Verified_Reviewers

- **Development_Teams_develops_Games(game_id, development_team_email)**

Development_Teams_develops_Games.game_id references Games

GameDevelopmentTeam.DevelopmentTeamEmail references DevelopmentTeam

- **Comments_on_Review(comment_id, review_id)**

Comments_on_Review.comment_id references Comments

Comments_on_Review.review_id references Reviews

- **Comments_on_Topic(comment_id, topic_id)**

Comments_on_Topic.comment_id references Comments

Where Comments_on_Topic.topic_id references Review

- **Comments_on_Gaming_Conference_Reviews(comment_id, gaming_conference_review_id)**

Comments_on_Gaming_Conference_Reviews . comment_id references Comments

Comments_on_Gaming_Conference_Reviews.game_conferences_review_id references Reviews

- **Game_presented_in_Conference(game_id, conference_id)**

Game_presented_in_Conference . game_id references Games

Game_presented_in_Conference.conference_id references Conference

- **Members_add_Members(requester_email, receiver_email, accepts).**

Members_add_Members. requester_email references Normal_Users

Member_add_Members. receiver_email references Normal_Users

- **Normal_User_sends_Thread(sender_email, reciever_email, thread_text)**

Normal_User_sends_Thread(sender_email, receiver_email) references Normal_Users

- **Game_Conference_attended_by_Member(game_conference_id, member_email)**

Game_Conference_attended_by_Member.game_conference_id references Gaming_Conferences

Game_Conference_attended_by_Member.member_email references Members

- **Community_has_Members(community_id, member_email)**

Community_has_Members.community_id references Communities

Community_has_Members.member_email references Members

- **Normal_User_recommends_Game(game_id, recommender_email, reciever_email)**

Normal_User_recommends_Game.game_id references Games

Normal_User_recommends_Game(recommenderEmail, recieverEmail) references Normal_Users

