Tables:

- Users(UserID, Gamertag, FirstName, LastName, Password, Email)
    - Primary key UserID
    - Explanation: Table of users (a single entity is a user). Other tables can reference this table when they want to discover what user made a post
- Posts(PostID, UserID, ItemID, PostTime, GoldCost, PostText, Active)
    - Primary Key PostID
    - Foreign Key UserID → Users.UserID
    - Foreign Key ItemID → Items.ItemID
    - Explanation: Users will post what they want to sell and indicate what they'd be willing to trade for (can be multiple types of items)
- BarterOptions(PostID, ItemID, AskingNum, GroupNum)
    - Foreign Key PostID → Posts.PostID
    - Foreign Key ItemID → Items.ItemID
    - Explanation: Associates posts with items and number of items user is willing to trade for. GroupNum is how a user can associate barter options that are linked together. In other words, if a person is willing to sell a sword for 3 potions AND a helmet, the two individual barter options will also contain the same GroupNum to associate them. It is a GroupNum and not a key because there is no limit.
- Items(ItemID, ItemName, TypeID)
    - Primary Key: ItemID
    - Foreign Key: TypeID → references ItemType.TypeID
    - Explanation: These are items that can be bought and sold in the game. Each row represents a class of item, not an instance.
- ItemType(TypeID, TypeName)
    - Primary Key: TypeID
    - Explanation: Each entry in this table is an allowed type that an item can be. (i.e. sword, shield, potion etc). Items will have an item type. This separates the data to ensure items have proper types

Evidence of normalization:

We followed the 4th form of normalization by ensuring that multiple posts created by a user are stored in their own table with references to the UserID.

The 3rd form of normalization is also followed because we ensure that the primary key is the functional determinant of the other attributes in the relation.