**BMÜ329 Veri Tabanı Sistemleri Dersi Dönem Projesi Gereksinimleri ve E-R Diyagramı**

|  |
| --- |
| **Proje Başlığı:** MMO Oyun Sistemleri İçin Veri Tabanı Mimarisi |
| **Proje Ekibindeki kişiler:** Barış Geçer , Ali Baran , Mehmet Karagülle |

1- **Proje Gereksinimleri**

|  |
| --- |
| Bu proje, bir MMO (Massively Multiplayer Online) oyun için geliştirilen veritabanı modelini tanımlamaktadır. Veritabanı, oyuncuların hesap bilgilerini, karakter özelliklerini, görevleri, oyun içi varlıkları ve diğer oyun mekaniklerini organize etmek amacıyla tasarlanmıştır. Varlıklar ve Nitelikleri  1. **Membership\_Account (Üyelik Hesabı)** Oyuncuların hesap bilgilerini içeren bu varlık, her bir oyuncuya ait kullanıcı adı, e-posta adresi, şifre gibi bilgileri saklar. Bu bilgiler, oyuna giriş yapmak ve hesapla ilgili işlemleri gerçekleştirmek için kullanılır. 2. **Player\_Character (Oyuncu Karakteri)** Oyuncuların oyun içinde oluşturduğu karakterlere dair bilgileri barındıran varlıktır. Her bir "Player\_Character" varlığı, karakterin benzersiz bir kimliğini (CharacterID) ve adını (CharacterName) , karakterin seviyesi (Character\_Level) , karakterin seviye atlaması için gereken deneyim puanı miktarı (Character\_Req\_Xp) , karakterin mevcut tecrübe puanı (Character\_Xp) niteliklerini içerir. 3. **Attributes (Özellikler)** Oyuncu karakterinin çeşitli oyun içi özelliklerini (örneğin, sağlık, mana, saldırı gücü gibi) içerir. "Player\_Character" ile 1:n şeklinde   ilişkilendirilmiştir; böylece bir karakterin bir nitelik tablosu olabilir ve aynı değerlere sahip özellik tablosu birden fazla karakterde bulunabilir.   1. **Skills (Yetenekler)** Oyuncuların öğrenebileceği yetenekler burada tanımlanır. Bir "Player\_Character" birden fazla yeteneğe sahip olabilir ve bu yetenekler "Skills" varlığında tutulur. "Player\_Character" ve "Skills" arasında n:m ilişkisi bulunmaktadır. 2. **Proficiency (Uzmanlık)** Oyuncu karakterlerinin belirli alanlardaki ustalık seviyelerini gösterir. Bu varlık, oyuncuların zamanla belirli alanlarda uzmanlaşmasını sağlar ve bu yeteneklerin gelişimini takip eder. Her "Player\_Character" bir veya birden fazla "Proficiency" ile ilişkilendirilmiştir (n:m   ilişkisi).   1. **Inventory (Envanter)** Oyuncuların oyun içinde sahip oldukları eşyaları organize eden "Inventory" varlığı oyuncuların taşıyabileceği eşya sayısına limit koyar (Capacity) Her "Player\_Character" bir envantere ve bu envanterde bir veya birden fazla eşyaya sahip olabilir. "Player\_Character" ve Inventory arasında 1:1 ilişkisi vardır, böylece bir oyuncu karakteri sadece bir envantere sahip olur 2. **Items (Eşyalar)**   **Oyuncuların oyun içerisinde sahip oldukları eşyaları temsil eder. Oyuncuların eşyaları kullabilme yetisine seviye kıstlaması getirilmiştir (Req\_Level) oyun içindeki farklı eşya tipleri birbirinden kendine özgü olan numarası ile ayrılır (Item\_ID) ve eşyalar bu numaraya göre sınıflanır örneğin iki farlı çeşit kılıç biri 1.1 iken diğeri 1.2 olarak numaralandırılır burada 1 kılıçların numarası noktadan sonrası ise eşyanın ayırt edicisidir Envanter ve Eşyalar n:n ilişkisine sahiptir**   1. **Quests (Görevler)** Oyuncuların oyunda alabileceği görevleri temsil eder. Her görev, görev adı (Quest\_Name), açıklaması(Decription), görev ödülü (Reward\_Amount) gibi nitelikleri içerir. "Player\_Character" ve "Quests" arasında n:m   ilişkisi bulunmaktadır; yani bir görev birden fazla karakter tarafından yapılabilir ve bir karakter birden fazla görevi üstlenebilir.   1. **Monsters (Canavarlar)** ve **NPCs (Oyuncu Olmayan Karakterler)** Oyun dünyasında bulunan canavarlar ve NPC'ler (Non-Player Characters) bu varlıklarla temsil edilir. Her bir NPC, oyunun hikayesi veya oyuncuların ilerlemesi için belirli işlevlere sahiptir. "Quests" ile "NPC’s" arasında 1:n ilişkisi vardır .Bu sayede bir görevde bir NPC yer alırken bir görev de birden fazla NPC yer alabilir.Canavarlar ise oyuncunun görev sırasında karşılaşacığı zorluklardır eşyalarda olan sınıflandırma sistemi canavarlar için de geçerlidir 2. **Factions (Gruplar)**  Oyun içindeki grupları ve oyuncuların bu gruplara katılımlarını tanımlayan varlıkdır. “Factions” ile "Player\_Character" arasında n:1 ilişkisi vardır; yani her oyuncu yalnızca bir gruba katılabilir ve her grup birden fazla karaktere sahip olabilir. 3. **Achievements (Başarımlar)** Oyuncuların oyun boyunca kazandıkları ödülleri ve başarıları içerir. "Membership\_Account" ve "Achievements" arasında n:m   ilişki vardır; her üyelik hesabında birden fazla başarıma sahip olabilir.Bir başarım birden fazla üyelik hesabında olabilir.   1. **Authority(Yetki)**   Oyunlardaki oyuncuları denetleyen duruma göre oyuna müdahale eden yetkili görevli.”Authority” yetkisine sahip kişilier geçici sürelik yasaklama olsun,oyundan atma olsun o tarz yetkilerin bulunduğu bir veritabanı tablosudur.”Authority” ile “Membership\_Account” arasında (1:n) ilişki vardır bir yetkilinin birden fazla hesabı olabilir ama bir hesabın tek yetkisi olabilir  13.**Server(Sunucu)**  Oynanan oyundaki üye hesaplarının yer aldığı çok fazla kişinin aynı ortamda bulunup oynadığı sistematik yapıya sunucu deriz.Sunucuların kendisi ”server\_id” leri “server\_ip “leri “server\_location “ gibi nitelikleri bulunur .Server ile Membership\_Account arasında da (1:n) ilişki vardır bir sunucuda birden fazla üye hesabı olurken bir üye hesabı tek server da yer alır .  14.**Appearance(Dış Görünüş)**  Bir karakterin dıştan görünümü sağlayan özellikleri dış görünüş tablosundaki nitelikler ile belirliyoruz.Bir karakterin saç rengi,göz rengi,kas tipi gibi nitelikleri bulunur.Player\_Character ile Appearance arasında ( n:1 )ilişki vardır. Bir karakterin bir dış görünüş olabilir bir dış görünüş birden fazla karakterde olabilir   İlişkiler ve Sayısal Kısıtlamalar  * **Membership\_Account - Player\_Character**: 1:n   Bir "Player\_Account" (oyuncu hesabı) birden fazla "Player\_Character" (oyuncu karakteri) oluşturabilir, ancak her karakter yalnızca bir hesap ile ilişkilidir.   * **Attributes – Player\_Character**: 1:n   Bir "Player\_Character" bir "Attributes" (özellik tablosu) değerine sahip olabilir, aynı zamanda bir özellik tablosu birden fazla karakterde olabilir.   * **Player\_Character - Skills**: n:n   Bir "Player\_Character" birden fazla "Skills" (yetenek) öğrenebilir ve aynı yetenek birden fazla karakter tarafından kullanılabilir.   * **Player\_Character - Proficiency**: n:n   Her "Player\_Character" birden fazla "Proficiency" (uzmanlık) alanında beceri sahibi olabilir.Bir uzmanlık alanında da birden fazla karakter sahip olabilir.   * **Player\_Character - Inventory - Items**: 1:1 | 1:n   Her "Player\_Character" bir envantere sahiptir ve bu envanterde birden fazla "Items" (eşya) bulunabilir   * **Player\_Character - Quests**: n:m   Bir "Player\_Character" birden fazla "Quests" (görev) alabilir ve aynı görev birden fazla karakter tarafından yapılabilir.   * **Quests - NPCs**: n:1   Her görevde bir "NPCs" (oyuncu olmayan karakter) olurken her NPC ise birden fazla görev verebilir   * **Player\_Character – Factions: n:1**   Bir "Player\_Character" birden fazla "Factions" (grup) üyesi olamaz ve aynı grup birden fazla karaktere sahip olabilir.   * **Membership\_Account - Membership\_Account**: n:m   Bir "Player\_Account" (oyuncu hesabı) bir veya birden fazla "Friends" (arkadaş) ilişkisine sahip olabilir.   * **Membership\_Account - Achievements**: m:n   Her "Player\_Account" birden fazla "Achievements" (başarım) kazanabilir, ancak her başarı bir hesaba aittir |

2 – E-R Diyagramı  
  
