Univerzitet u Sarajevu
Elektrotehnički Fakultet

Elektrotehnički fakultet
Univerziteta u Sarajevu
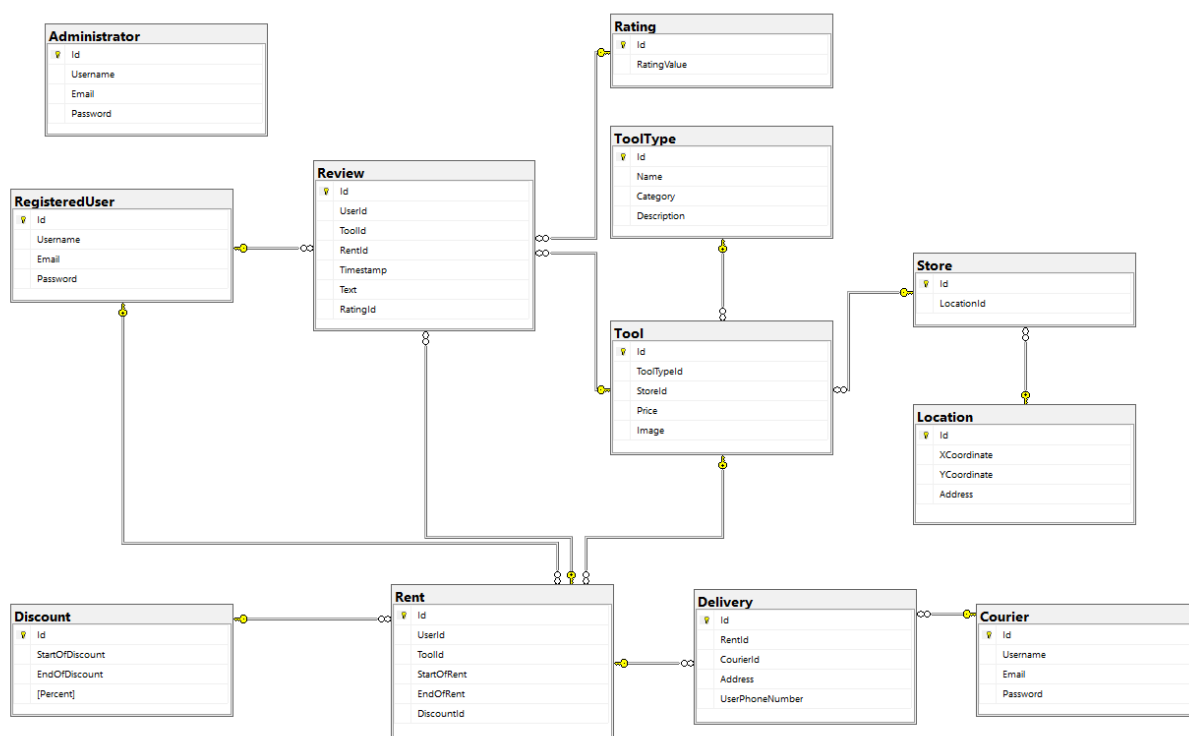
Objektno Orijentisana Analiza i Dizajn

Dokumentacija refactoringa rješenja - Burgija

Tim:
1. Emir Bronja (18253)
2. Zlatan Jakić (18855)
3. Edis Jašarević (19247)
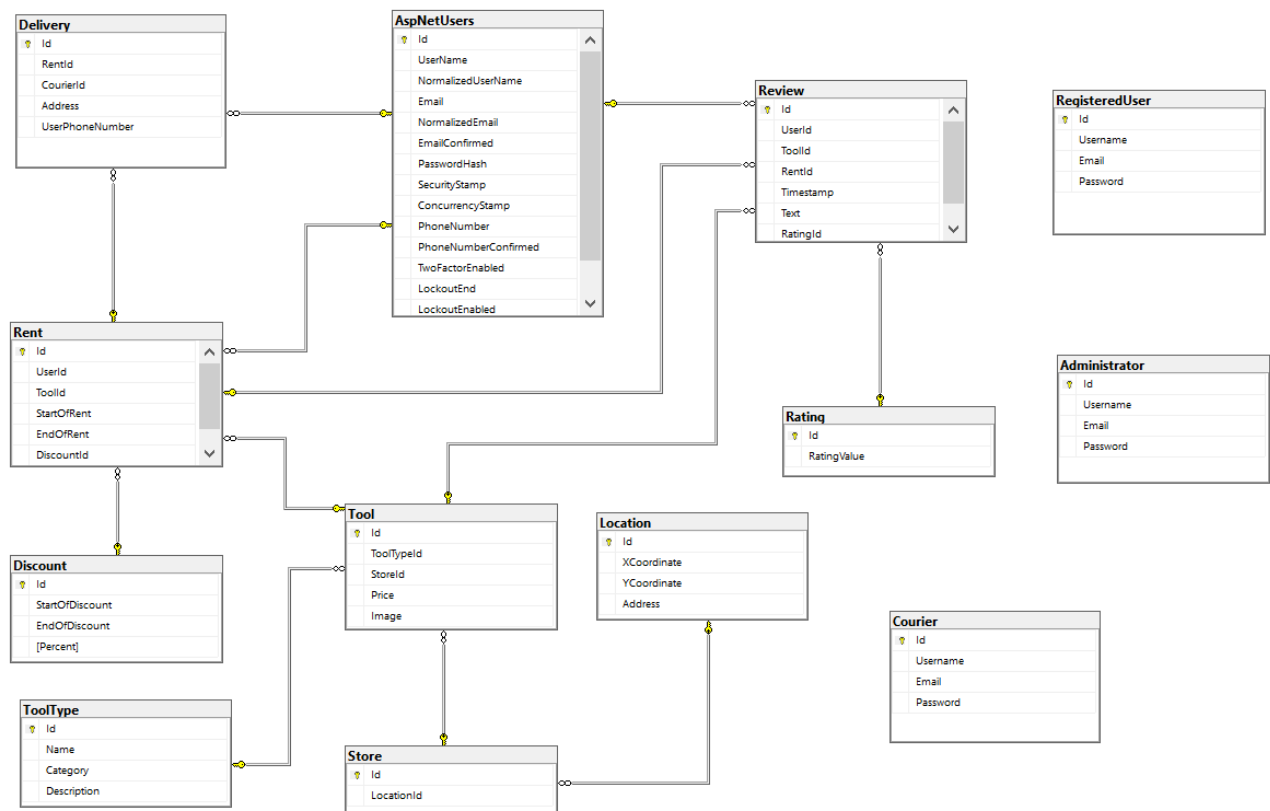4. Ahmed Ljubunčić (19023)
5. Mersid Pilipović (19070)

Prvobitni izgled ER-Dijagrama:

Modifikacija foreign key-eva gdje god je figurisao dostavljac, admin i registered user, tako da je sad foreign key na ASPNetUsers tabeli. Razdvojili smo veze tih tabela RegisteredUser, Admin i Courier tako da one nisu povezane ni sa jednom drugom. Također, promijenili smo tip Primary Key-a u ASPNetUsers tabelli u int, da bi se mogle povezati sa drugim tabelama jer i one imaju ključeve tipa int.
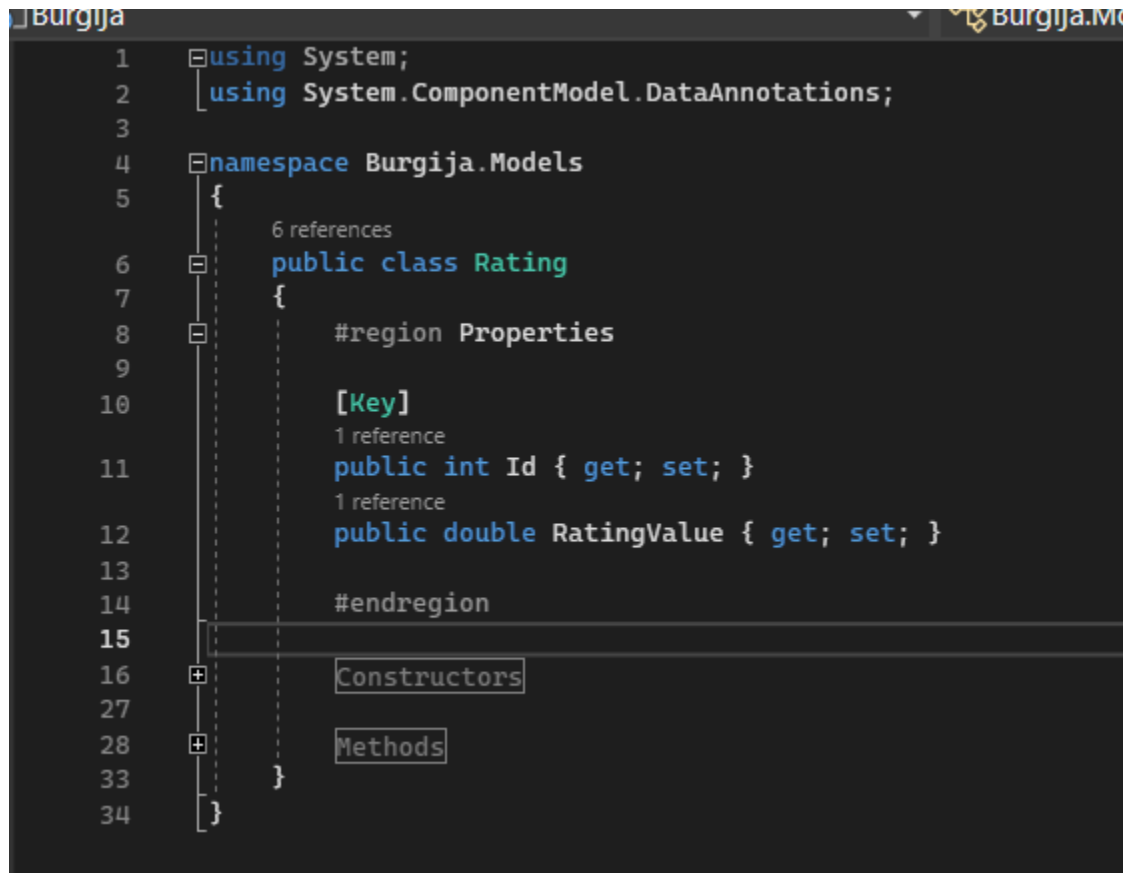
Novi izgled ER-Dijagrama:

```csharp
namespace Burgija.Models
{
    29 references
    public class Rent : IComparable<Rent>
    {
        #region Properties

        [Key]
        8 references
        public int Id { get; set; }
        9 references
        public IdentityUser<int> User { get; set; }

        [ForeignKey("AspNetUsers")]
        6 references
        public int UserId { get; set; }
        12 references
        public Tool Tool { get; set; }

        [ForeignKey("Tool")]
        9 references
        public int ToolId { get; set; }
        23 references
        public DateTime StartOfRent { get; set; }
        20 references
        public DateTime EndOfRent { get; set; }
        9 references
        public Discount Discount { get; set; }

        [ForeignKey("Discount")]
        6 references
        public int? DiscountId { get; set; }
        5 references
        public double RentPrice { get; set; }

        #endregion
```

```csharp
namespace Burgija.Models
{
    18 references
    public class Delivery : IComparable<Delivery>
    {
        #region Properties

        [Key]
        10 references
        public int Id { get; set; }
        7 references
        public Rent Rent { get; set; }

        [ForeignKey("Rent")]
        0 references
        public int RentId { get; set; }
        5 references
        public IdentityUser<int> Courier { get; set; }

        [ForeignKey("AspNetUsers")]
        3 references
        public int? CourierId { get; set; }
        [Required]
        9 references
        public string Address { get; set; }
        9 references
        public string UserPhoneNumber { get; set; }

        #endregion

        Constructors
```

```csharp
namespace Burgija.Models
{
    4 references
    public class Review
    {
        #region Properties

        [Key]
        1 reference
        public int Id { get; set; }
        1 reference
        public IdentityUser<int> User { get; set; }

        [ForeignKey("AspNetUsers")]
        1 reference
        public int UserId { get; set; }
        1 reference
        public Tool Tool { get; set; }

        [ForeignKey("Tool")]
        1 reference
        public int ToolId { get; set; }
        1 reference
        public Rent Rent { get; set; }

        [ForeignKey("Rent")]
        0 references
        public int RentId { get; set; }
        1 reference
        public long Timestamp { get; set; }
        2 references
        public string Text { get; set; }
        1 reference
        public Rating Rating { get; set; }

        [ForeignKey("Rating")]
        1 reference
        public int RatingId { get; set; }

        #endregion
```

Prvobitno smo imali atribut rating u klasi Review, međutim na preporuku asistenta smo uvidjeli da je bolje da imamo zasebnu klasu Rating.

```csharp
using System;
using System.ComponentModel.DataAnnotations;

namespace Burgija.Models
{
    6 references
    public class Rating
    {
        #region Properties

        [Key]
        1 reference
        public int Id { get; set; }
        1 reference
        public double RatingValue { get; set; }

        #endregion

        Constructors

        Methods
    }
}
```