Life-In-Notes

# Controller (folder)

# Models (folder)

## Account.cs

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Threading.Tasks;

namespace Life\_In\_Notes.Models

{

public class Account

{

// Account Id

public int Id { get; set; }

[Required(ErrorMessage = "User Name Field is required")]

[MaxLength(50, ErrorMessage ="Entry Name cannot exceed 50 characters")]

// User Name

public string Name { get; set; }

[Required]

[RegularExpression(@"^[a-zA-Z0-9\_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-.]+$",

ErrorMessage = "Invalid Email Format")]

// UserEmail

public string Email { get; set; }

// UserPassword

public string Password { get; set; }

}

}

## AppDB.Context.cs

## Entry.cs

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Threading.Tasks;

namespace Life\_In\_Notes.Models

{

public class Entry

{

// represents the entryId

public int Id { get; set; }

[Required(ErrorMessage = "Name of Entry field is required")]

[MaxLength(50, ErrorMessage = "Name of Entry cannot exceed 50 characters")]

// Name of entry

public string Name { get; set; }

[Required(ErrorMessage ="The Theme field is required")]

// Type of entry

public TType? Type { get; set; }

[Required(ErrorMessage ="The Type field is required")]

// Theme of Entry

public Theme? Theme { get; set; }

[Required(ErrorMessage = "Date of Entry field required")]

// Date Entry was entered

public string Date { get; set; }

// A reference date for entries needing additional dates

public string RefDate { get; set; }

[Required(ErrorMessage = "The Content field is required")]

// Content of the entry

public string Content { get; set; }

// Represents the UserId related to the entry

public int UserId { get; set; }

}

}

## Notes.cs

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Threading.Tasks;

namespace Life\_In\_Notes.Models

{

public class Note

{

// represents the noteId

public int Id { get; set; }

[Required(ErrorMessage = "The Theme field is required")]

// Type of entry

public TType? Type { get; set; }

[Required(ErrorMessage = "The Type field is required")]

// Theme of Entry

public Theme? Theme { get; set; }

[Required(ErrorMessage = "Date of Entry field required")]

// Date Entry was entered

public string Date { get; set; }

// A reference date for entries needing additional dates

public string RefDate { get; set; }

[Required(ErrorMessage = "The Content field is required")]

// Content of the entry

public string Content { get; set; }

// Represents the EntryId related to the entry

public int EntryId { get; set; }

}

}

# ViewModels (folder)

# Views (folder)