

# Smart Diary — Blazor + ASP.NET Core

This project is a full-stack diary and productivity system built with Blazor Server and ASP.NET. It showcases practical engineering decisions, clean C# architecture, and several complex features working together in a coherent UI.

## Key Features

- Rich Text Editor – Create structured entries with formatting support.
- Mood Tracking – Each entry logs emotional state for long-term trend analysis.
- Statistics Dashboard – Aggregated charts for entries, word counts, and mood changes.
- Admin Panel – Complete user and group management, moderation actions, activation, deactivation, and password resets.
- PIN-Protected Sections – Privacy controls for sensitive groups or entries.
- PDF Export – Export entries or groups for offline backup.
- Authentication & Roles – Secure login, registration, and role-based access.
- Light/Dark Mode – as User selects theme.

The image displays several screenshots of the Smart Diary application:

- Admin Panel:** Shows user management, group management, entry moderation, and additional statistics. It includes a table of users with columns for ID, Username, Activated, Role, and Description, along with various moderation actions like deactivate account, reset password, and give admin rights.
- Entry Moderation:** A detailed view of a specific entry with a timestamp (11/21/2025), content ("I can write There some diaries, and then read them... Rich text formating is not included"), and a list of 4 items under "And headings too".
- Rich Text Editor:** A modal showing a rich text editor interface with a toolbar and placeholder text ("Please enter what you think currently").
- Statistics Dashboard:** A table showing aggregated data over time, including Date, Entries count, Symbols count, Average symbols per entry, and Total mood of day. Below it is a line chart showing trends for Entries count in day, Symbols count in day, Average symbols per Entry in day, and Total mood point in day from Nov 2 to Nov 16, 2025.
- Mobile View:** A screenshot of the application's mobile responsive design.



## Engineering Highlights

- C# + .NET 9 across both client and server
- EF Core with migrations and relational design
  - Clean and scalable folder structure
  - Reusable UI components
- Asynchronous operations for responsiveness
- Deployed using Render & GitHub Pages

## Purpose

This application demonstrates my ability to design and implement:

- secure multi-user systems
- admin and moderation tooling
  - analytics dashboards
- structured, maintainable Blazor applications

The project is actively improved as I refine architecture and expand features