

Full-Stack Blazor & ASP.NET Engineer • Custom Web Applications • Scalable Architecture

I design and build full-stack applications using Blazor WebAssembly, ASP.NET, EF Core, and SQL. My work prioritizes structural clarity, predictable performance, and maintainability. I specialize in transforming loosely defined ideas into robust, production-ready systems with clean APIs, solid authentication flows, and modular architecture.

Core Technical Capabilities

- Blazor WebAssembly (UI, components, routing, state)
- ASP.NET backend with layered architecture
- REST API design + integration
- Authentication & Authorization (Identity, JWT, roles)
- Entity Framework Core + relational DB modeling
- Admin tooling (dashboards, logs, moderation systems)
- Data analytics pages (charts, aggregations, statistics)
- UI/UX logic for complex workflows
- Deployment to Render, Azure, GitHub Pages

Case Study: Personal Diary Application

The screenshot displays two views of a web application. The left view is the 'Statistics Page', which includes a sidebar with navigation links (Home, Manage account, Logout, Statistics, Admin, Export cre..., Developing..., PIN-Start, How are th..., Create new diary group) and a main content area with a table of statistics and a line chart.

Date	Entries count	Symbols count	Average symbols per entry	Total mood of day
10/30/2025	10	2023	202.3	-1
10/31/2025	1	227	227	1
11/8/2025	3	375	125	1
11/10/2025	1	505	505	
11/12/2025	1	53	53	
11/20/2025	1	347	347	
11/21/2025	2	904	452	
Totals:	19	4434	233	

The right view shows a diary entry for 11/21/2025. The entry text is: 'How are this is working, if i am...'. Below the text is a list of items: '1. With list consisting', '2. organization', '3. and unorganization', '4. lots of unorganization'. The entry is marked as 'added'. The bottom of the page shows a text input field with the placeholder 'Please, enter what you think currently' and a 'Send' button.

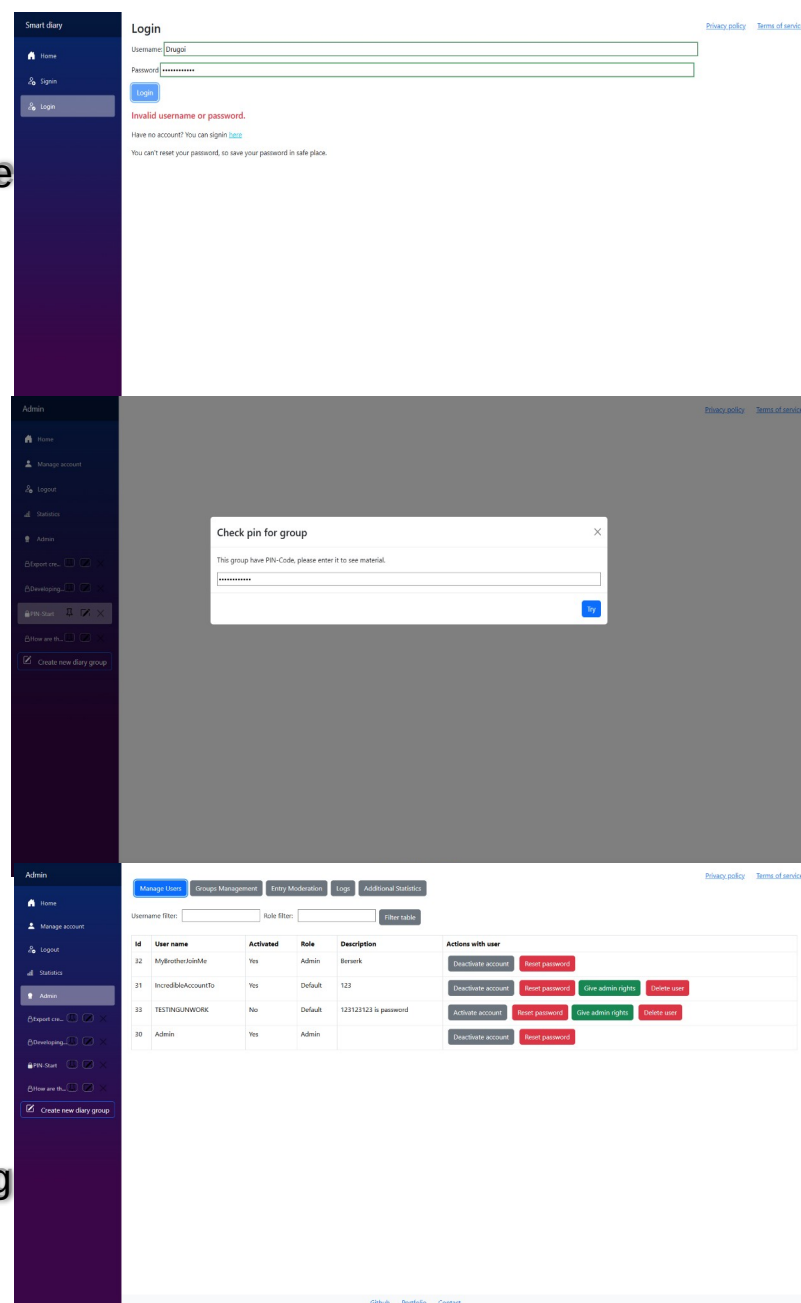
Problem

Users needed a centralized system for structured journaling with privacy controls, mood tracking, analytics, and administrative oversight.

Solution

Designed and implemented a full Blazor SaaS-style system:

- Authentication via ASP.NET Identity
- Rich text editor with format persistence
- Mood tracking integrated into entry metadata
- PDF export for entries and groups
- PIN protection for sensitive areas
- Admin panel with:
 - User management
 - Group controls
 - Moderation actions
- Statistics dashboard with charts, aggregates, and time-based trends
- Modular architecture prepared for scaling (services, repositories)



Technical Notes

- C# 12 / .NET 9
- EF Core with migrations
- Clean separation between UI, Domain, and Infrastructure
- Custom middleware for activity logging
- Responsive UI via Bootstrap + custom components
- CI automation for testing and deployment

Outcome

A maintainable and extensible web application demonstrating full lifecycle engineering: UX, backend logic, data modeling, admin workflows, and analytics.

Links: [Github Repo](#), [Live demo](#), [Portfolio page](#)