Универзитет „Св. Кирил и Методиј“ во Скопје
ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО

Advanced Project - Documentation

Application Name: AppointMe

Course: Integrated Systems

Student: Blerona Muladauti Mentor: Ph.D Petre Lameski

Student ID:221541

Skopje, January 2026

Contents

Introduction

Owning and managing a business is not easy. Managing appointments creates a lot of tension and stress if not done correctly. Many businesses still rely on manual scheduling, on paper scheduling, phone calls or message applications which are all fine up to some point, but usually time inefficient and not a handy solution. The goal of this project is to design and implement AppointMe, a web-based appointment management system that simplifies scheduling while ensuring reliability and easy use. This application is designed as a multi-tenant system, which allows multiple independent businesses to use the same platform while keeping their data fully isolated. The focus is not only on functionality but also maintainability, clean architecture, real-world applicability and scalability.

About AppointMe

AppointMe is a multi-tenant appointment scheduling web application developed using .NET 8 MVC and structured according to the Onion Architecture pattern. The system allows businesses (tenants) to manage customers, services, appointments, and invoices within a single platform.

Each business operates independently within the system, meaning that data such as customers, appointments, services, prices and invoices are strictly isolated per tenant. Tenants can create appointments, for their customers, that follow business-defined rules such as working hours and working days, appointment duration, and public holidays.

Key Functionalities

The core functionalities provided by AppointMe include:

- Tenant-based customer management
- Appointment creation with availability and overlap checks
- Business-defined working hours and open days
- Service selection with price snapshot at booking time
- Automatic invoice generation per appointment
- Email notifications for appointments and invoices
- Calendar (.ics) file generation for external calendar tools
- Public holiday validation using an external API

Multi-Tenancy and Data Isolation

The system follows a strict tenant-based design, where each business is identified by a unique tenant identifier. All core entities such as customers, appointments, services, and invoices are linked to a specific tenant. Access to data is always filtered by tenant context, ensuring that no business can access or modify data belonging to another tenant. This approach enables safe data isolation while allowing the application to scale as a Software-as-a-Service platform.

Design Decisions and Limitations

The current version of AppointMe focuses on the tenant (business) side of appointment management. End customers interact indirectly through email notifications and calendar invitations, removing the need for a separate client-side portal at this stage. Online payments and advanced client self-service features were intentionally excluded to keep the system focused, maintainable, and aligned with project scope. The architecture allows these features to be added in future iterations without major redesign.

Configurable Services and Invoicing

The system is designed to be flexible and does not enforce the use of all available features. Each tenant can enable or disable optional modules such as services and invoicing through business configuration settings. When services are disabled, appointments can be created without attaching service offerings. Similarly, invoicing can be fully turned off for businesses that do not require invoice generation. These configuration options allow AppointMe to adapt to different business workflows without forcing unnecessary complexity, while still preserving consistent behavior and data integrity when features are enabled.

System Architecture – Onion Architecture Overview

The project architecture consists of multiple layers:

- Domain Layer – Core business entities and rules
- Repository Layer – Data access abstractions and implementations
- Service Layer – Application logic and orchestration
- Web Layer – User interface and HTTP request handling

Service Layer

- This layer coordinates interactions between domain models, repositories and external services. Some of the business rules in this project include:

- Validating appointment dates against business working hours and working days
- Preventing double-booking of time slots
- Handling public holiday restrictions
- Triggering email notifications upon appointment creation
- Generating invoices for appointments

Web Layer

-Implemented with ASP.NET MVC, it represents the interaction point of the application. The controllers in this layer are responsible for:

- Handling HTTP requests
- Validating user input
- Calling appropriate service methods
- Returning views or redirects

The views of course, represent the interaction with the user, and data display.

App Screenshots

The following screenshots include some of the most important views of the app:

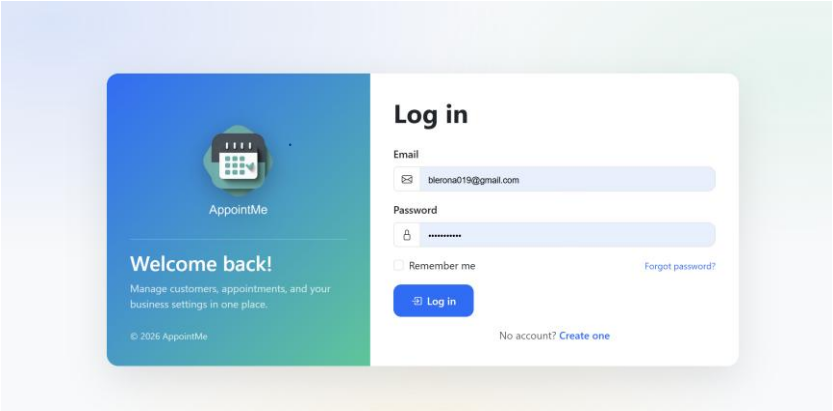

Image1. **Login page**

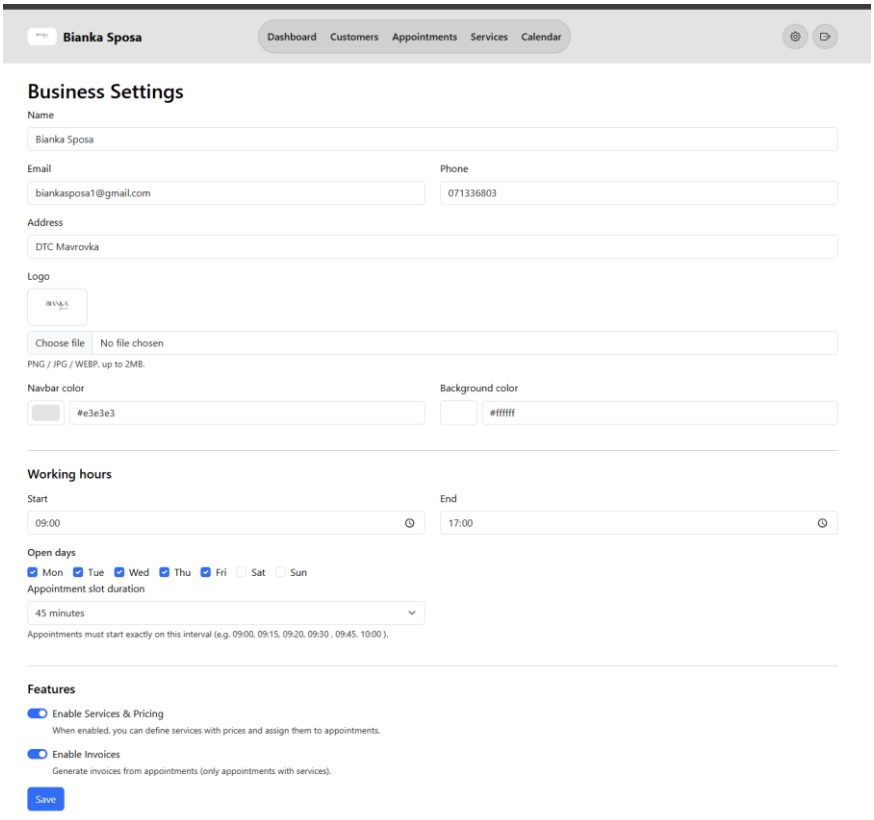

Image2. **Business Settings page**

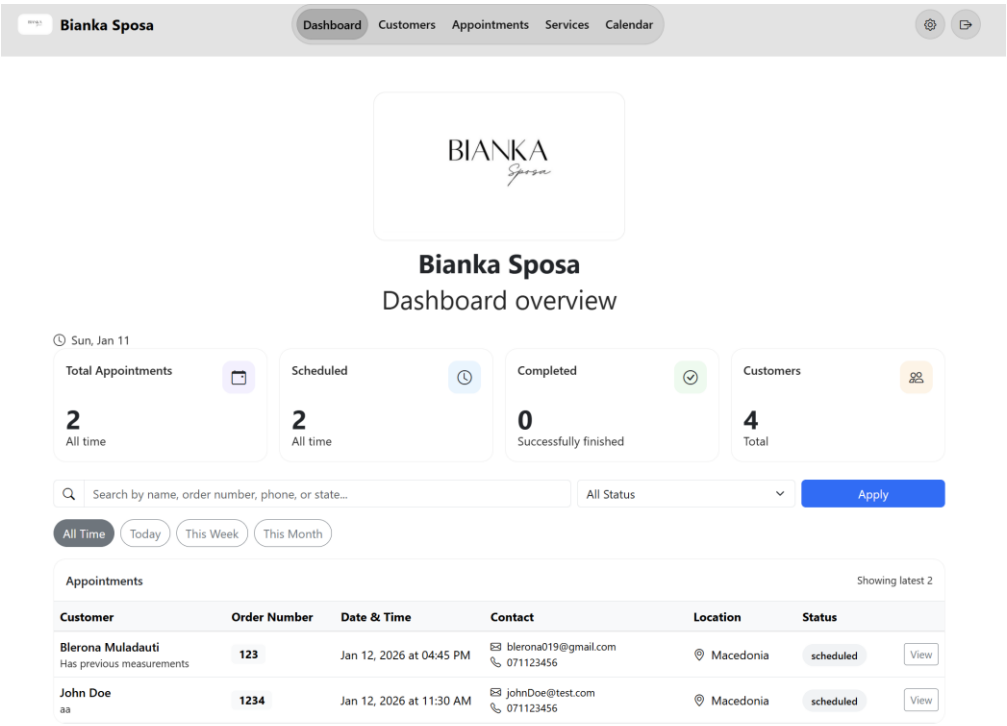

Image3. **Dashboard page**

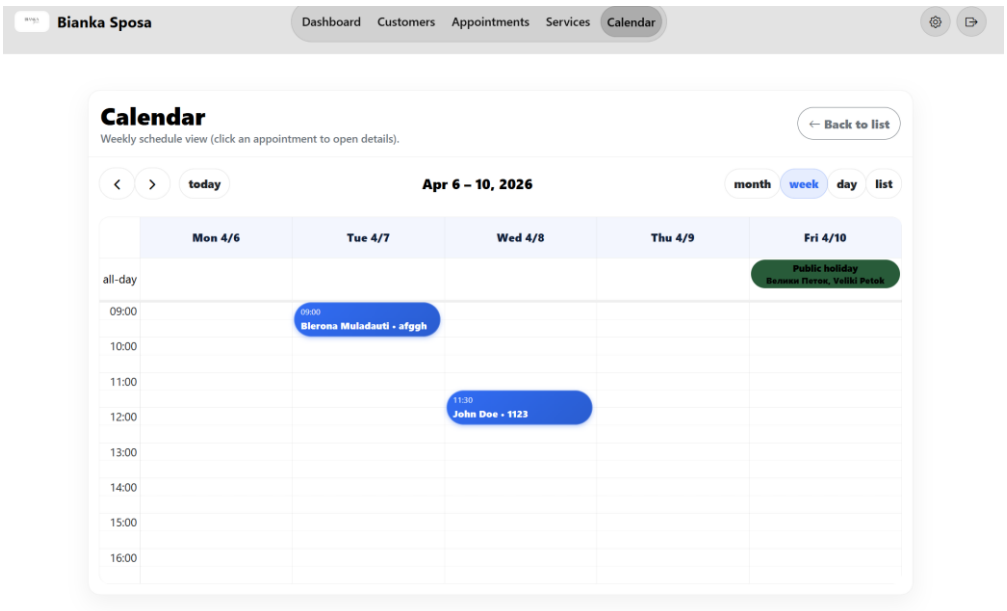

Image4. **Calendar page**

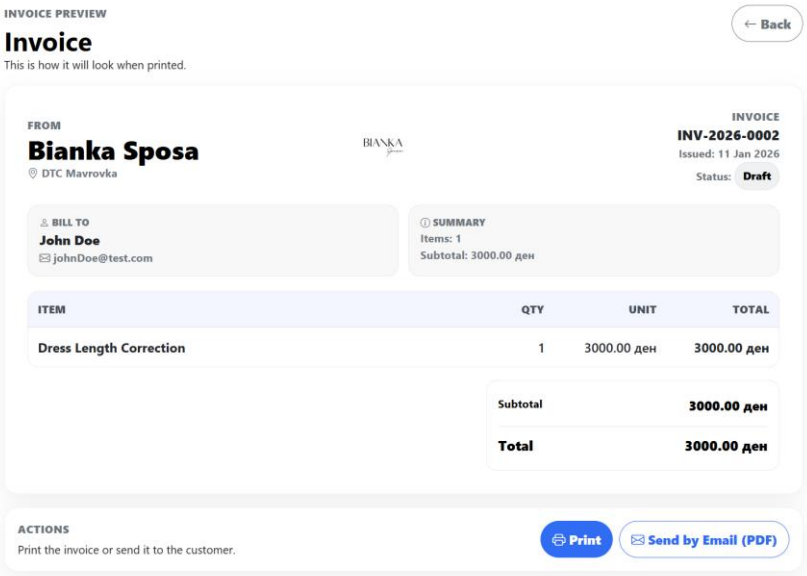

Image5. **Invoice page**

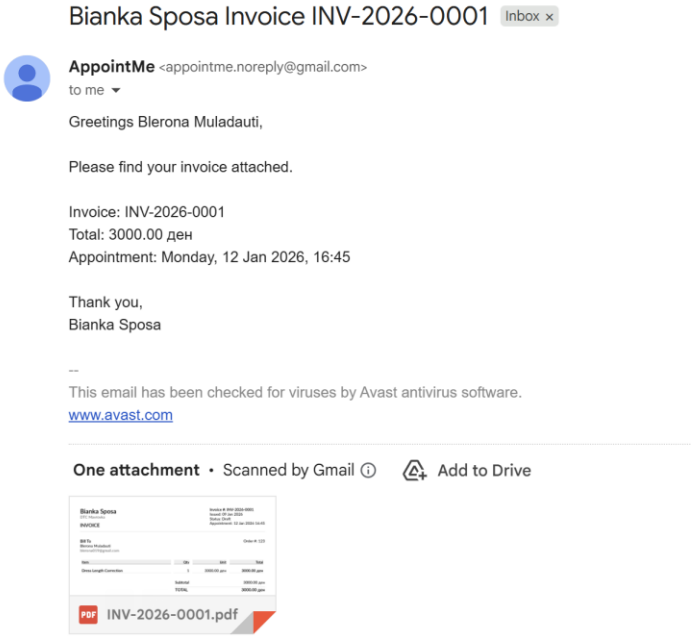

Image6. **Invoice email received**

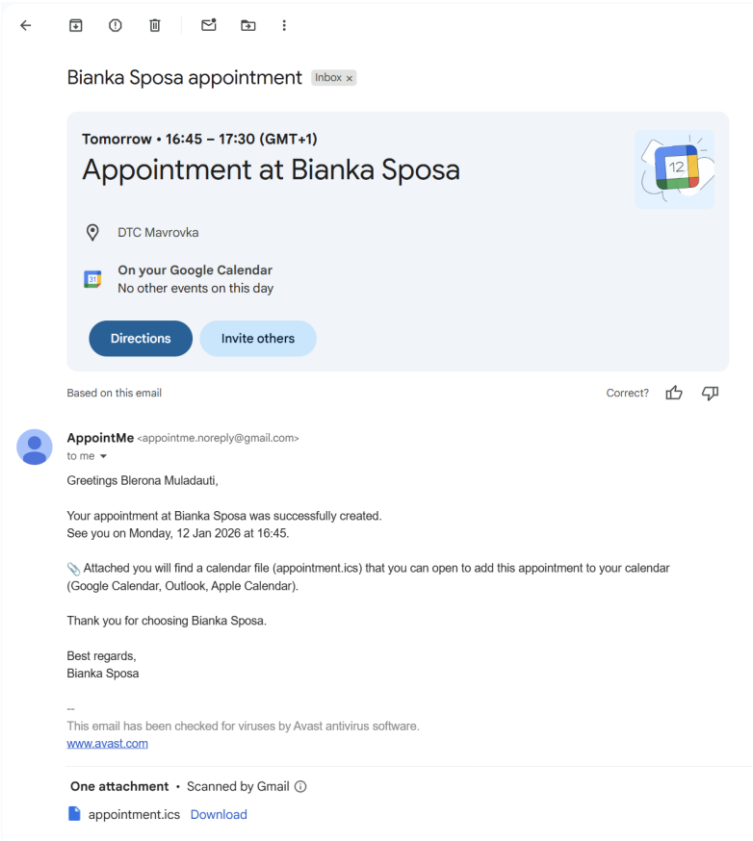

Image7. **Appointment created email received**

User Guidance and Application Flow

When a user registers in the AppointMe system, the registration represents the creation of a tenant, meaning the user becomes the owner and administrator of a business within the platform. Each registered business operates independently, with full data isolation from other tenants.

Initial Business Setup

Immediately after registration and first login, the user is redirected to the Business Settings screen. This step is mandatory, as the core behavior of the application is defined here. On this screen, the business owner can configure:

- Business name and contact email
- Business location and optional logo
- Working days and working hours
- Default appointment duration
- Whether the business will:
    - Use services with pricing, or
    - Operate as a simple appointment management system without services

- Whether invoice generation should be enabled or disabled

These settings allow the application to adapt to different types of businesses and workflows, without enforcing unnecessary features.

Application Usage Flow

After completing the business configuration:

1. Services (Optional)
   If services are enabled, the user can define the services offered by the business, including categories and prices. These services can later be attached to appointments.
2. Customer Management
   The user can add customers, storing their contact information for future appointments.
3. Appointment Scheduling
   Appointments can then be created for customers, respecting:
   - Business working hours and working days
   - Appointment duration
   - Existing appointments (to prevent double-booking)
   - Public holidays (via external API integration)
4. Invoices (Optional)
   If invoice functionality is enabled, invoices can be generated automatically from appointments and optionally sent to customers via email.

Email Integration Notice!!!

For testing email functionality (appointment confirmations and invoice delivery), it is required to enter a valid and existing email address when creating customers. Emails will not be delivered to non-existent or placeholder addresses.