REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Computer Engineering Department

TAMIRHANEM: A COMPREHENSIVE AUTO SERVICE MANAGEMENT PLATFORM

Şevval Nur KASAVET 210502087

Yamaç BEZİRGAN 200502023

Bekir ÇAKIR 210502084

CE491 COMPUTER ENGINEERING GRADUATION PROJECT I

Supervisor

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Istanbul, 2026

The Graduation Project titled "**TAMIRHANEM: A COMPREHENSIVE AUTO SERVICE MANAGEMENT PLATFORM WITH AI-POWERED ASSISTANCE**" prepared by **Şevval Nur KASAVET (210502087), Yamaç BEZİRGAN (200502023), and Bekir ÇAKIR (210502084)** and submitted on January 2026 has been accepted unanimously.

	Asst. Prof. Dr. Abdullahi Abdu IBRAHIM Supervisor

DECLARATION

I hereby declare that this Project meets all format and submission requirements for a CE491 COMPUTER ENGINEERING GRADUATION PROJECT I.

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Şevval Nur KASAVET

Signature: _______________

Yamaç BEZİRGAN

Signature: _______________

Bekir ÇAKIR

Signature: _______________

ABSTRACT

TAMIRHANEM: A COMPREHENSIVE AUTO SERVICE MANAGEMENT PLATFORM

WITH AI-POWERED ASSISTANCE

KASAVET, Şevval Nur; BEZİRGAN, Yamaç; ÇAKIR, Bekir

B.Sc., Department of Computer Engineering, Altınbaş University

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

January 2026

Total Page : 54

The automotive service industry in Turkey faces significant challenges in digital transformation, with many businesses still relying on manual paper-based processes for customer management, appointment scheduling, inventory tracking, and financial operations. This inefficiency leads to poor customer communication, inventory management difficulties, and inadequate online presence for service businesses. This graduation project presents Tamirhanem, a comprehensive web-based platform designed to address these challenges through digital transformation of auto service operations.

Tamirhanem is developed as a three-tier architecture platform serving three distinct user roles: customers (vehicle owners), industrialists (auto service business operators), and platform administrators. The system implements a complete solution including Customer Relationship Management (CRM), Enterprise Resource Planning (ERP) functionalities, work order management, digital invoicing with PDF generation, appointment scheduling with calendar integration, maintenance reminder systems, and an innovative AI-powered assistant utilizing Google Gemini API for intelligent customer and business support.

The platform is built using modern web technologies including React 18 with TypeScript for the frontend, Express.js with Node.js for the backend, and PostgreSQL with Prisma ORM for database management. The system implements JWT-based authentication with role-based access control (RBAC), ensuring secure multi-tenant operations. The database schema comprises 21 tables and 17 enumeration types, supporting over 200 RESTful API endpoints.

Key innovations include the business showcase feature providing customizable web presence for service businesses, real-time notification systems, comprehensive review and rating management, and AI-driven assistance for both customers and business operators. The platform successfully demonstrates the potential for digital transformation in the traditional automotive service sector.

Keywords: Auto Service Management, Web Application, CRM, ERP, AI Assistant, React, Node.js, PostgreSQL

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AI	:	Artificial Intelligence
API	:	Application Programming Interface
CORS	:	Cross-Origin Resource Sharing
CRM	:	Customer Relationship Management
CSS	:	Cascading Style Sheets
CRUD	:	Create, Read, Update, Delete
DOM	:	Document Object Model
ERP	:	Enterprise Resource Planning
HTML	:	HyperText Markup Language
HTTP	:	HyperText Transfer Protocol
JSON	:	JavaScript Object Notation
JWT	:	JSON Web Token
MVP	:	Minimum Viable Product
ORM	:	Object-Relational Mapping
PDF	:	Portable Document Format
QA	:	Quality Assurance
RBAC	:	Role-Based Access Control
REST	:	Representational State Transfer
SPA	:	Single Page Application
SQL	:	Structured Query Language
UI	:	User Interface
URL	:	Uniform Resource Locator
UUID	:	Universally Unique Identifier
UX	:	User Experience
XSS	:	Cross-Site Scripting

1. INTRODUCTION

The automotive service industry represents a critical sector within the Turkish economy, serving millions of vehicle owners annually. Despite the technological advancements witnessed across various industries, a significant portion of auto service businesses in Turkey continues to rely on traditional, paper-based methods for managing their daily operations. This digital gap creates numerous inefficiencies, including poor customer communication, inventory management challenges, and limited online visibility for service providers.

This graduation project introduces Tamirhanem, a comprehensive web-based platform designed to facilitate the digital transformation of auto service businesses. The platform provides an integrated solution that addresses the core operational needs of automotive service providers while simultaneously enhancing the customer experience through modern digital interfaces and intelligent assistance systems.

The name "Tamirhanem" combines the Turkish words "tamir" (repair) and "hanem" (my home/place), reflecting the platform's mission to create a digital home for auto service management. The platform serves three distinct user categories: vehicle owners seeking convenient service management, auto service business operators requiring comprehensive business tools, and platform administrators responsible for system oversight.

1.1 PROBLEM STATEMENT

The automotive service sector in Turkey faces several interconnected challenges that impede operational efficiency and customer satisfaction. Through preliminary research and stakeholder interviews, the following key problems have been identified:

Manual Process Management: The majority of auto service businesses continue to utilize paper-based systems for appointment scheduling, customer records, and inventory tracking. This approach results in data loss, scheduling conflicts, and administrative inefficiencies. Service technicians often spend valuable time searching through physical files rather than performing their core repair functions.

Customer Communication Deficiencies: Customers frequently remain uninformed about the status of their vehicle repairs. The absence of automated notification systems leads to unnecessary phone calls, customer anxiety, and reduced trust in service providers. This communication gap particularly affects customer retention and referral rates.

Inventory Control Challenges: Parts and supplies inventory tracking remains predominantly manual, resulting in stockouts of critical components, over-ordering of slow-moving items, and difficulty in identifying optimal reorder points. These inefficiencies directly impact service delivery times and business profitability.

Financial Management Complexity: Invoice generation, payment tracking, and financial reporting often involve time-consuming manual processes. The lack of integrated financial tools makes it difficult for business owners to maintain accurate records and gain insights into their financial performance.

Limited Digital Presence: Many auto service businesses lack any form of online presence, making it difficult for potential customers to discover their services, view their offerings, or schedule appointments. This digital invisibility places traditional businesses at a competitive disadvantage in an increasingly connected marketplace.

1.2 PROJECT OBJECTIVES

The primary objective of the Tamirhanem project is to develop a comprehensive, scalable, and user-friendly web platform that addresses the identified challenges in the automotive service industry. The specific objectives are outlined below:

The first objective is to implement a complete digital appointment management system enabling customers to schedule, modify, and cancel service appointments online while providing businesses with calendar-based visualization and management tools.

The second objective is to develop an integrated Customer Relationship Management (CRM) system for auto service businesses, enabling efficient tracking of customer information, vehicle histories, and service records.

The third objective is to create an Enterprise Resource Planning (ERP) module for inventory management, including stock tracking, supplier management, low-stock alerts, and movement history.

The fourth objective is to implement a work order management system that tracks service jobs from initiation through completion, including parts usage and labor tracking.

The fifth objective is to build a comprehensive invoicing system with automatic calculation, PDF generation, and payment tracking capabilities.

The sixth objective is to provide customizable digital showcase pages for businesses, enabling them to establish an online presence with service listings, contact information, and customer reviews.

The seventh objective is to integrate an AI-powered assistant utilizing the Google Gemini API to provide intelligent support for both customers and business operators, including vehicle diagnostics assistance and business analytics.

The eighth objective is to establish a secure, scalable platform architecture implementing modern security practices including JWT authentication and role-based access control.

1.3 PROJECT SCOPE

The scope of the Tamirhanem project encompasses the development of a fully functional web-based platform with the following inclusions and exclusions:

1.3.1 Included Features

The project scope includes three distinct user role implementations (Administrator, Business Operator, and Customer), each with dedicated panels and functionality. The system encompasses comprehensive appointment management with both list and calendar views, a complete CRM system with customer and vehicle database management, an ERP module for inventory and supplier management, a work order tracking system with parts integration, an invoicing system with PDF generation and payment tracking, a review and rating system with business response capabilities, maintenance reminder functionality, a notification center for all user types, customizable business showcase pages with three template options, AI-powered assistant integration for customer and business support, a support ticket system for platform assistance, and a complete platform administration panel with analytics and reporting.

1.3.2 Excluded Features (Future Development)

The following features are explicitly excluded from the current project scope but are identified for potential future development: native mobile applications for iOS and Android platforms, online payment gateway integration, SMS notification capabilities, multi-language support beyond Turkish and English interfaces, and real-time chat functionality between customers and businesses.

1.4 DOCUMENT STRUCTURE

This graduation project report is organized into six main chapters. Chapter 1 provides an introduction to the project, including the problem statement, objectives, and scope. Chapter 2 presents a comprehensive literature review covering the automotive service industry, existing solutions, relevant web technologies, and AI applications in customer service. Chapter 3 details the methodology employed in the project, including the development approach, technology stack selection, system architecture, database design, API design, and security considerations. Chapter 4 describes the system design and implementation in detail, covering frontend and backend development, database implementation, and the specific features of each user panel. Chapter 5 presents the testing strategy and evaluation results, including functional testing, user interface testing, and performance metrics. Chapter 6 concludes the report with a project summary, achievements, limitations, and recommendations for future work.

Table 1.1 presents the project team members and their primary areas of responsibility within the development process.

Table 1.1: Project Team Members and Responsibilities

Team Member	Student ID	Primary Responsibilities
Şevval Nur KASAVET	210502087	Customer Panel Development, Frontend Implementation, Quality Assurance and Testing

Yamaç BEZİRGAN	200502023	Admin Panel Development, Backend Infrastructure, Database Architecture, Authentication System
Bekir ÇAKIR	210502084	Business Panel Development, Backend Services, Work Order and Invoice Systems

2. LITERATURE REVIEW

This chapter presents a comprehensive review of the literature relevant to the Tamirhanem project. The review covers the current state of the automotive service industry, existing digital solutions in the market, the web technologies employed in the project, and the application of artificial intelligence in customer service contexts.

2.1 AUTOMOTIVE SERVICE INDUSTRY OVERVIEW

The automotive aftermarket service industry represents a significant economic sector globally, with particular importance in countries with high vehicle ownership rates such as Turkey. According to industry reports, the Turkish automotive aftermarket was valued at approximately USD 12 billion in 2023, with projections indicating continued growth driven by the increasing average age of vehicles on the road and rising vehicle complexity [1].

The sector comprises diverse service providers ranging from authorized dealership service centers to independent repair shops, commonly referred to as "sanayici" in Turkish. Independent service providers constitute the majority of the market, serving price-conscious consumers who prefer more affordable alternatives to dealership services [2]. However, these independent businesses often lack the technological infrastructure and digital tools available to larger dealership networks.

Within this context, the digital maturity gap between large dealership networks and independent service providers creates both challenges and opportunities. Industry observations suggest that businesses adopting integrated management platforms experience notably improved customer loyalty and reduced administrative overhead [3]. For Tamirhanem's target market of independent Turkish service providers, this gap represents the core problem our platform addresses through accessible, affordable digital tooling designed specifically for their operational scale and requirements.

The COVID-19 pandemic accelerated the need for digital transformation in the automotive service industry. Contactless service scheduling, digital payment options, and online communication channels became essential features rather than conveniences [4]. This shift in consumer expectations has created an opportunity for platforms that can bridge the digital gap for traditional service providers.

2.2 EXISTING SOLUTIONS AND COMPETITORS

Several software solutions exist in the automotive service management space, each with varying degrees of functionality and target markets. This section examines the major existing solutions and identifies the gaps that Tamirhanem aims to address.

Shop-Ware is a cloud-based auto repair shop management system widely used in North America. The platform offers comprehensive features including digital vehicle inspections, customer communication tools, and integrated payment processing. However, the solution is primarily designed for the North American market and lacks localization for Turkish business requirements, including Turkish tax regulations and local payment methods [5].

Mitchell 1 provides ProDemand and Manager SE solutions that combine repair information databases with shop management capabilities. These enterprise-level solutions offer extensive vehicle database integration but require significant investment and technical infrastructure, making them unsuitable for small to medium-sized independent service providers in Turkey [6].

In the Turkish market, local solutions offer basic appointment scheduling and customer record management. However, these solutions typically lack modern user interfaces, mobile responsiveness, AI integration, and comprehensive business management features such as inventory control and financial reporting [7].

Table 2.1 presents a comparative analysis of existing solutions against the features offered by Tamirhanem.

Table 2.1: Comparison of Existing Auto Service Management Solutions

Feature	Shop-Ware	Mitchell 1	Local Solutions	Tamirhanem
Turkish Localization	No	No	Yes	Yes
CRM System	Yes	Yes	Basic	Yes
Inventory Management	Yes	Yes	Limited	Yes

Work Order System	Yes	Yes	Basic	Yes
AI Assistant	No	No	No	Yes
Business Showcase	No	No	No	Yes
Customer Portal	Limited	Limited	No	Yes
Modern UI/UX	Yes	Moderate	No	Yes
Affordable for SMBs	Moderate	No	Yes	Yes

2.3 WEB TECHNOLOGIES REVIEW

Modern web application development has evolved significantly, with numerous frameworks and technologies available for building scalable, maintainable applications. This section reviews the key technologies selected for the Tamirhanem platform and their relevance to the project requirements.

2.3.1 Frontend Technologies

For Tamirhanem's frontend requirements, React was selected based on its component-based structure that naturally maps to our three-panel architecture. Each panel (admin, business, customer) shares common UI elements such as tables, forms, and navigation components, making reusability essential for our small team's productivity [8]. The framework's efficient update mechanism proved particularly valuable for our dashboard interfaces where appointment lists, stock levels, and financial summaries require frequent refreshes without full page reloads.

TypeScript adoption was driven by our team's parallel development workflow where three developers simultaneously built different panels. The static type definitions created a contract between frontend services and backend API responses, catching integration mismatches during compilation rather than runtime [9]. This proved especially valuable when coordinating the shared data structures for appointments, invoices, and user profiles across all three panel implementations.

Tailwind CSS was chosen to maintain visual consistency across Tamirhanem's three panels developed by different team members. Rather than creating separate stylesheets that might diverge over time, the utility class approach embedded styling decisions directly in components [10]. Combined with shadcn/ui's pre-built accessible components, this enabled rapid UI development while ensuring the admin, business, and customer panels share a cohesive visual language.

2.3.2 Backend Technologies

Node.js selection unified our development language across frontend and backend, reducing the learning curve for team members and enabling shared utility functions and type definitions. For Tamirhanem's expected usage pattern where multiple business operators simultaneously manage appointments and inventory, the runtime's concurrent request handling aligns well with our scalability requirements [11].

Express.js provided the structural foundation for Tamirhanem's API layer through its middleware pipeline architecture. Our implementation chains authentication verification, role-based authorization, request logging, and input validation as discrete middleware functions, enabling clean separation of cross-cutting concerns from business logic [12].

Prisma ORM bridges our TypeScript codebase with the PostgreSQL database through auto-generated type definitions derived from the schema file. For Tamirhanem's 21-table schema with complex relationships between users, appointments, work orders, and invoices, Prisma's type-safe query builder prevents field name typos and relationship navigation errors at compile time [13].

2.3.3 Database Technologies

PostgreSQL was selected over alternatives based on Tamirhanem's specific data modeling requirements. The platform's core workflow connecting appointments to work orders to invoices demands strict referential integrity through foreign key constraints, which PostgreSQL enforces reliably [14]. Additionally, the native JSON column type enabled flexible storage for business showcase customization data and working hours configurations without requiring schema modifications for each customization option.

2.4 AI IN CUSTOMER SERVICE APPLICATIONS

Conversational AI capabilities have become increasingly accessible for integration into business applications through commercial API offerings. For Tamirhanem, incorporating an intelligent assistant addressed a specific user need: customers frequently have questions about maintenance schedules, warning lights, and service requirements that do not necessitate human staff involvement [15].

Google Gemini was selected for Tamirhanem's AI assistant based on three practical criteria: the multimodal capability enabling customers to upload vehicle photos for preliminary issue identification, the straightforward REST API integration compatible with our Express.js backend, and the free tier quota sufficient for development testing and initial deployment phases [16].

The value proposition for Tamirhanem's AI integration centers on reducing repetitive inquiry handling for business operators while providing customers with immediate responses outside business hours [17]. Common questions regarding oil change intervals, tire rotation schedules, and warning light meanings can be addressed automatically, allowing service staff to concentrate on hands-on repair work rather than phone inquiries.

The application of AI in automotive diagnostics has also shown promising results. Computer vision systems can analyze images of vehicle components to identify potential issues, providing valuable preliminary assessments that can guide service decisions [18]. This capability is integrated into Tamirhanem through the AI assistant's image analysis feature.

3. METHODOLOGY

This chapter describes the methodology employed in the development of the Tamirhanem platform. The discussion covers the development approach, technology selection rationale, system architecture design, database schema design, API design principles, and security architecture implementation.

3.1 DEVELOPMENT METHODOLOGY

Tamirhanem's development followed an iterative approach adapted to our academic timeline and three-person team structure. Rather than rigid sprint ceremonies, we implemented weekly synchronization meetings to coordinate interface contracts between the three panels, daily progress updates via Discord messaging, and bi-weekly advisor consultations to validate alignment with graduation project requirements [19].

The development process was organized into phases spanning 18 weeks. The first phase (Weeks 1-2) focused on project planning, requirements analysis, and technology selection. The second phase (Weeks 3-4) involved detailed system design, including database schema design and user interface mockups. The third phase (Weeks 5-8) concentrated on backend infrastructure development, including the authentication system, API architecture, and database implementation. The fourth phase (Weeks 9-12) involved parallel development of the three user panels by the respective team members. The fifth phase (Weeks 13-14) focused on frontend-backend integration and system testing. The sixth phase (Weeks 15-16) was dedicated to quality assurance and bug fixing. The final phase (Weeks 17-18) involved documentation and project delivery.

Version control through Git with GitHub enabled parallel development of the three panels without merge conflicts disrupting progress. Each team member maintained a dedicated branch for their panel, with integration to the main branch occurring after peer review. This workflow proved essential during weeks 9-12 when all three panels were under simultaneous active development.

3.2 TECHNOLOGY STACK SELECTION

The technology stack for Tamirhanem was selected based on several criteria including developer productivity, ecosystem maturity, performance characteristics, community support, and alignment with project requirements. The selected technologies are presented in Tables 3.1 and 3.2.

Table 3.1: Frontend Technology Stack

Technology	Version	Purpose
React	18.3.1	User interface framework with component-based architecture
TypeScript	5.8.3	Static typing for JavaScript, improving code quality
Vite	5.4.19	Build tool providing fast development server and optimized builds
Tailwind CSS	3.4.17	Utility-first CSS framework for rapid styling
shadcn/ui	-	Accessible, customizable UI component library (48 components)
TanStack Query	5.83.0	Server state management with caching and synchronization
React Router DOM	6.30.1	Client-side routing for single-page application
Axios	1.13.2	HTTP client for API communication
React Hook Form	7.61.1	Form state management with validation support
Zod	3.25.76	Schema validation for form inputs
Recharts	2.15.4	Data visualization and charting library
Framer Motion	12.24.8	Animation library for enhanced user experience

Table 3.2: Backend Technology Stack

Technology	Version	Purpose
Node.js	18+	JavaScript runtime environment for server-side execution
Express.js	4.18.2	Web application framework for API development
TypeScript	5.3.3	Static typing for backend code
Prisma ORM	7.2.0	Object-relational mapping with type-safe database access
PostgreSQL	15+	Relational database management system
jsonwebtoken	9.0.3	JWT token generation and verification
bcrypt	6.0.0	Password hashing for secure credential storage
Zod	4.2.1	Input validation and schema definition

PDFKit Google Generative AI	0.17.2 0.24.1	PDF document generation for invoices Gemini API integration for AI assistant
Pino	8.17.2	High-performance logging
Helmet	7.1.0	Security headers middleware

3.3 SYSTEM ARCHITECTURE DESIGN

Tamirhanem's architecture separates concerns across three distinct layers to enable independent development and testing of each tier. This separation allowed our team to work on frontend components while backend API contracts remained stable, and permitted database schema refinements without requiring immediate frontend changes.

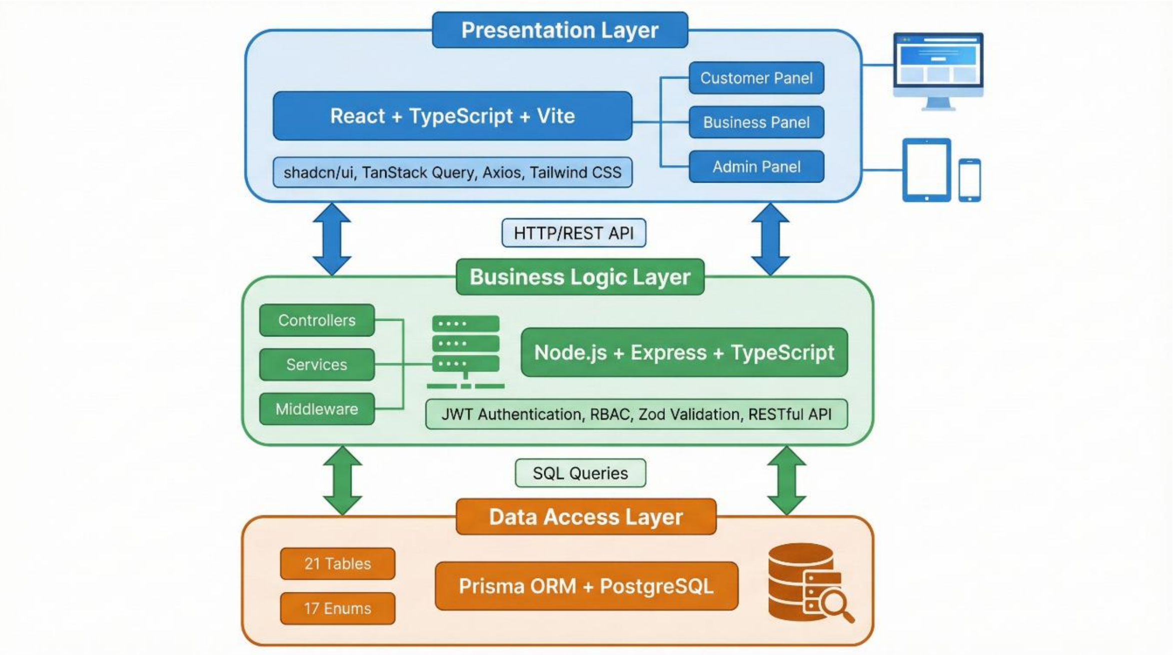


Figure 3.1 Three-Tier System Architecture of Tamirhanem Platform

The Presentation Layer is implemented as a React Single Page Application (SPA) responsible for rendering the user interface, handling user interactions, managing client-side state, and communicating with the backend through RESTful API calls. The SPA approach provides a fluid user experience by avoiding full page reloads during navigation.

The Business Logic Layer is implemented using Express.js and comprises controllers that handle HTTP requests, services that implement business logic, middleware for cross-cutting concerns such as authentication and logging, and validators for input sanitization.

This layer enforces business rules, manages transactions, and coordinates data flow between the presentation and data access layers.

The Data Access Layer utilizes Prisma ORM to interact with the PostgreSQL database. Prisma provides a type-safe query builder that generates optimized SQL queries while preventing SQL injection attacks through parameterized queries. The data layer also handles connection pooling and transaction management.

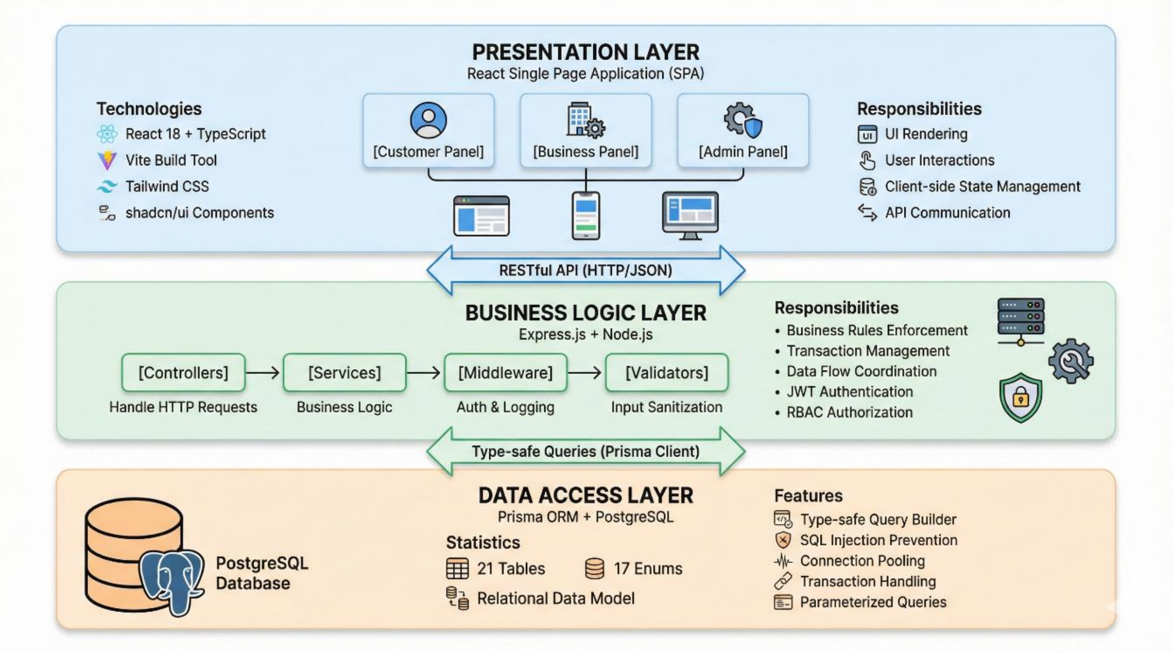


Figure 3.2 System Component Architecture and Layer Responsibilities

3.4 DATABASE DESIGN

The database design for Tamirhanem was developed through an iterative process of requirements analysis, entity identification, relationship mapping, and normalization. The resulting schema comprises 21 tables and 17 enumeration types, designed to support the platform's comprehensive functionality while maintaining data integrity and query performance.

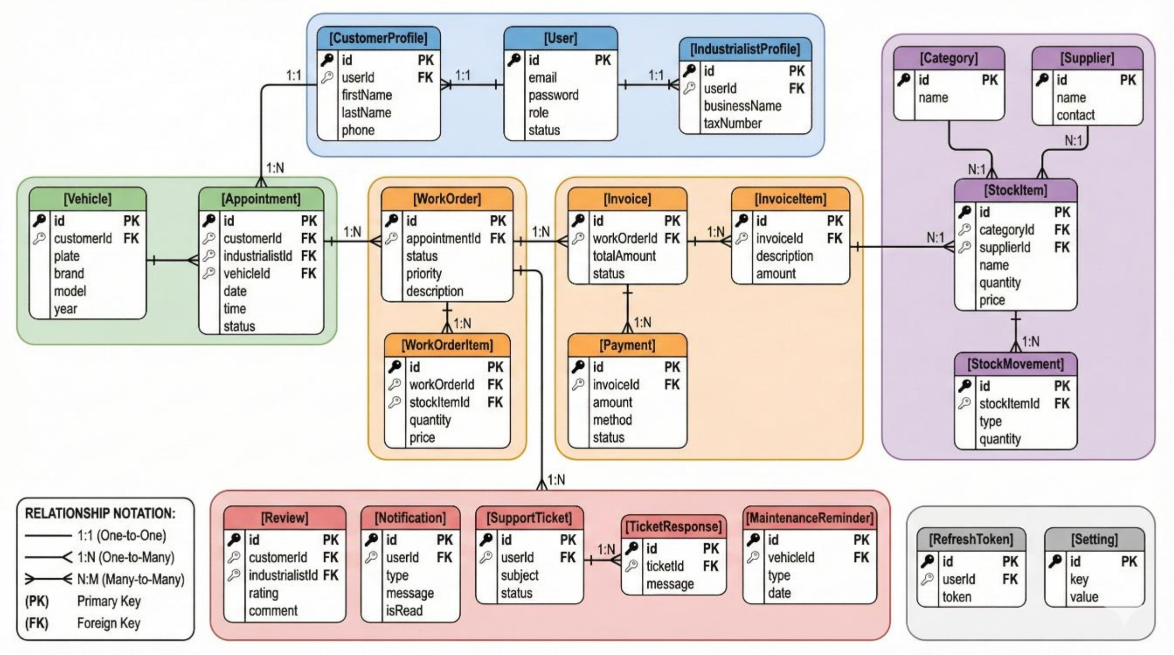


Figure 3.3 presents the Entity-Relationship diagram showing the 21 database tables and their relationships. Tables are color-coded by functional domain: user management (blue), appointments (green), work orders and invoicing (orange), inventory (purple), and support services (red).

Table 3.3 provides an overview of the database tables and their primary functions within the system.

Table 3.3: Database Tables Overview

Table Name	Description	Key Relationships
User	Core user account information and authentication credentials	1:1 with CustomerProfile, IndustrialistProfile
CustomerProfile	Extended profile information for customer users	1:N with Vehicle, Appointment, Invoice
IndustrialistProfile	Business information for service providers	1:N with Appointment, StockItem, WorkOrder
Vehicle	Customer vehicle information	N:1 with CustomerProfile
Appointment	Service appointment records	1:1 with WorkOrder, Review
WorkOrder	Service job tracking	1:N with WorkOrderItem, 1:1 with Invoice
Invoice	Billing documents	1:N with InvoiceItem, Payment
StockItem	Inventory items	N:1 with Category, Supplier

Review	Customer ratings and reviews	N:1 with CustomerProfile, IndustrialistProfile
Notification	System notifications to users	N:1 with User
SupportTicket	Customer support requests	1:N with TicketResponse

3.5 API DESIGN

The Tamirhanem API follows RESTful design principles, providing a consistent and intuitive interface for client-server communication. The API implements standard HTTP methods (GET, POST, PUT, PATCH, DELETE) with appropriate status codes and structured JSON response formats.

The URL structure follows the pattern /api/{resource} for collection operations and /api/{resource}/{id} for individual resource operations. Sub-resources are accessed through nested paths such as /api/work-orders/{id}/items. Statistics and aggregate data are available through dedicated endpoints like /api/{resource}/stats.

All API responses follow a consistent format including a success boolean, a message string, the data payload, and pagination information where applicable. Error responses include appropriate HTTP status codes and descriptive error messages to facilitate debugging and user feedback.

Table 3.5: API Endpoint Categories and Counts

Category	Base Path	Endpoint Count	Description
Authentication	/api/auth	6	User registration, login, token management
Admin	/api/admin	60+	Platform management, analytics, user control
Customer Portal	/api/portal	30	Customer-facing operations
Appointments	/api/appointments	9	Scheduling operations
Work Orders	/api/work-orders	9	Service job management
Invoices	/api/invoices	10	Billing and payment operations
Stock	/api/stock	15	Inventory management
AI Assistant	/api/ai	5	AI chat and suggestions

Total	-	200+	Complete API surface

3.6 SECURITY ARCHITECTURE

Security is a critical aspect of the Tamirhanem platform, given the sensitive nature of customer and business data handled by the system. The security architecture implements multiple layers of protection following industry best practices and security standards.

3.6.1 Authentication System

Tamirhanem's authentication employs a dual-token mechanism calibrated for our multi-role platform. The 7-day access token lifetime balances security with convenience for business operators accessing the platform daily, while the 30-day refresh token stored in the RefreshToken table enables session continuity. The token payload containing userId, email, and role allows each API request to be authorized without database lookup, reducing latency for authenticated operations.

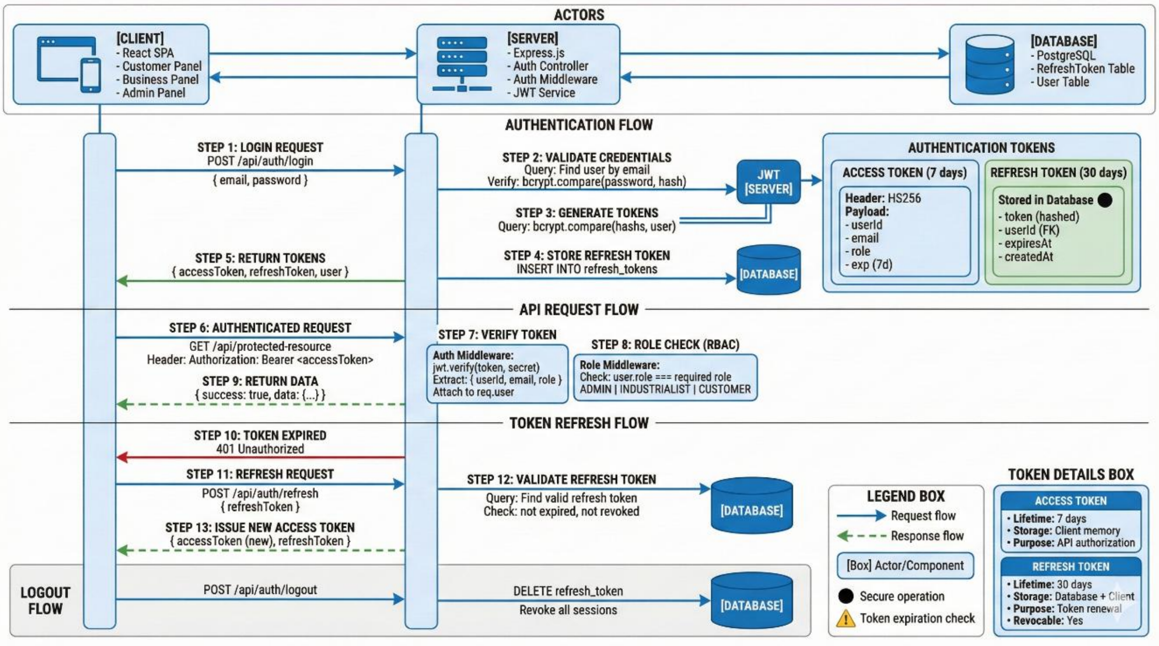


Figure 3.4 depicts the JWT authentication flow with dual-token strategy. The diagram shows the complete lifecycle: user login with bcrypt password verification, access token (7 days) and refresh token (30 days) generation, authenticated API requests with RBAC middleware, token refresh mechanism, and logout with session revocation.

Password storage utilizes bcrypt hashing with cost factor 12, selected to balance computational overhead with security requirements for our expected user base. The registration form enforces an 8-character minimum with mandatory uppercase, lowercase, and numeric characters, validated client-side through Zod schemas before server submission.

3.6.2 Authorization and Access Control

Authorization logic enforces three distinct access levels corresponding to Tamirhanem's user types. ADMIN users access platform-wide management functions including user approval and revenue reporting. INDUSTRIALIST users access their own business data including customers, inventory, and invoices. CUSTOMER users access personal vehicles, appointments, and service history. A roleGuard middleware intercepts protected route requests, comparing the authenticated user's role against endpoint requirements before permitting access.

3.6.3 Data Protection

Every API endpoint validates incoming data through Zod schemas defined in dedicated validator files, rejecting malformed requests before they reach service layer processing. Prisma's query builder constructs parameterized SQL statements automatically, eliminating manual string concatenation that could enable injection attacks. On the frontend, React's JSX rendering escapes dynamic content by default, mitigating script injection through user-submitted text fields.

The Express application applies Helmet.js middleware to inject protective HTTP headers preventing clickjacking, MIME sniffing, and enforcing HTTPS connections. CORS middleware restricts API access to requests originating from our frontend domain, blocking attempts to access Tamirhanem's API from unauthorized external applications or scripts.

4. SYSTEM DESIGN AND IMPLEMENTATION

This chapter presents the detailed design and implementation of the Tamirhanem platform. The discussion covers the frontend architecture and component design, backend service implementation, database realization, and the specific features developed for each user panel.

4.1 FRONTEND IMPLEMENTATION

The frontend of Tamirhanem is implemented as a React Single Page Application (SPA) with TypeScript, providing a type-safe development experience. The application comprises 141 React components organized into logical categories based on functionality and user role.

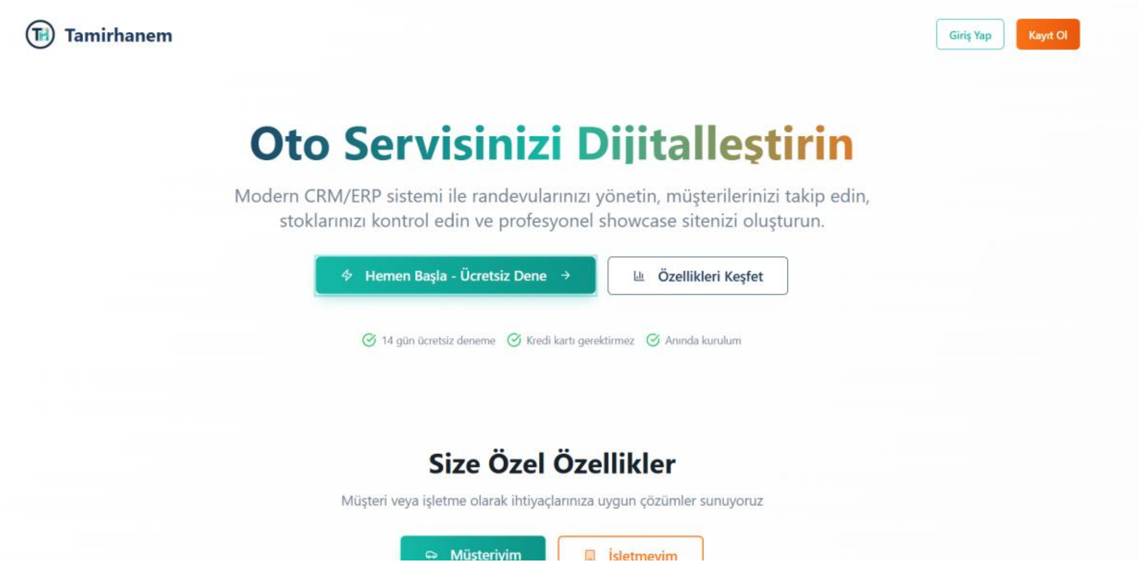
4.1.1 Component Architecture

The component architecture follows a hierarchical structure with shared UI components at the base, feature-specific components in the middle layer, and page components at the top. The shadcn/ui library provides 48 accessible, customizable base components built on Radix UI primitives, ensuring consistent styling and accessibility compliance.

The components are organized into the following directories: admin/ containing 35 components for the administration panel, dashboard/ containing 45 components for the business panel, customer/ containing 6 high-level components for the customer panel, ui/ containing 48 shadcn/ui base components, templates/ containing 3 showcase template components, layout/ containing navigation and structure components, and ai/ containing the AI assistant chat interface components.

4.1.2 Landing Page Design

The landing page serves as the primary entry point for the platform, designed to communicate the value proposition of Tamirhanem to potential users. The page features a hero section with animated elements using Framer Motion, a features section highlighting benefits for both customers and businesses, a pricing section presenting subscription tiers, and a footer with contact information and legal links.



Oto Servisinizi Dijitalleştirin

Modern CRM/ERP sistemi ile randevularınızı yönetin, müşterilerinizi takip edin, stoklarınızı kontrol edin ve profesyonel showcase sitenizi oluşturun.

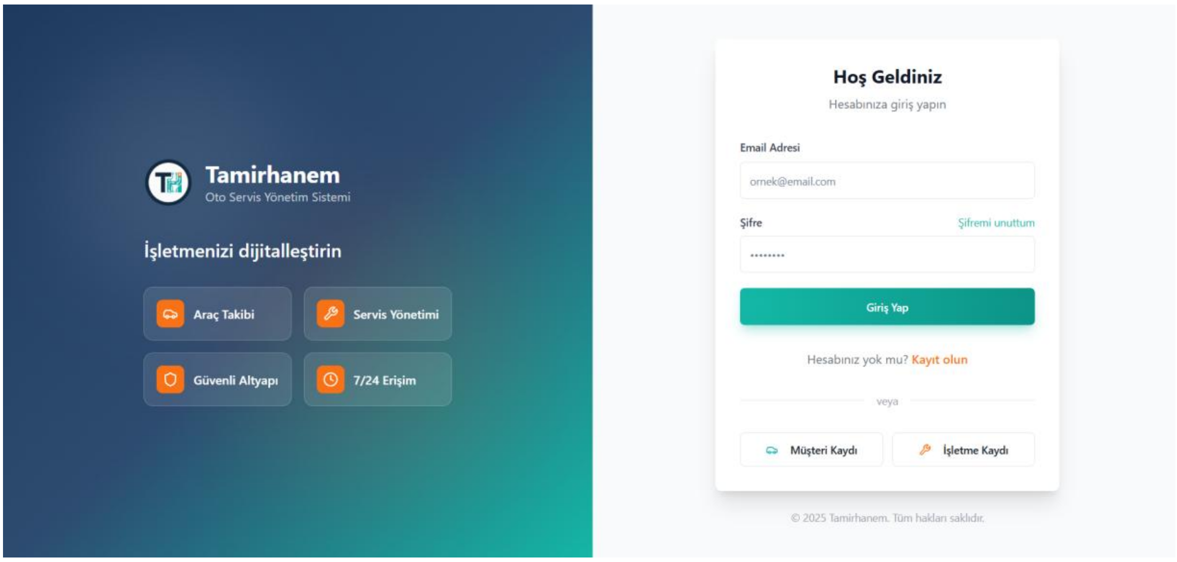
Size Özel Özellikler

Müşteri veya işletme olarak ihtiyaçlarınıza uygun çözümler sunuyoruz

Figure 4.1

4.1.3 Authentication Pages

The login and registration pages implement modern form handling with React Hook Form and Zod validation. The registration flow supports role selection, allowing users to register as either customers or business operators with appropriate profile information collection.



İşletmenizi dijitalleştirin

Hoş Geldiniz

Hesabınıza giriş yapın

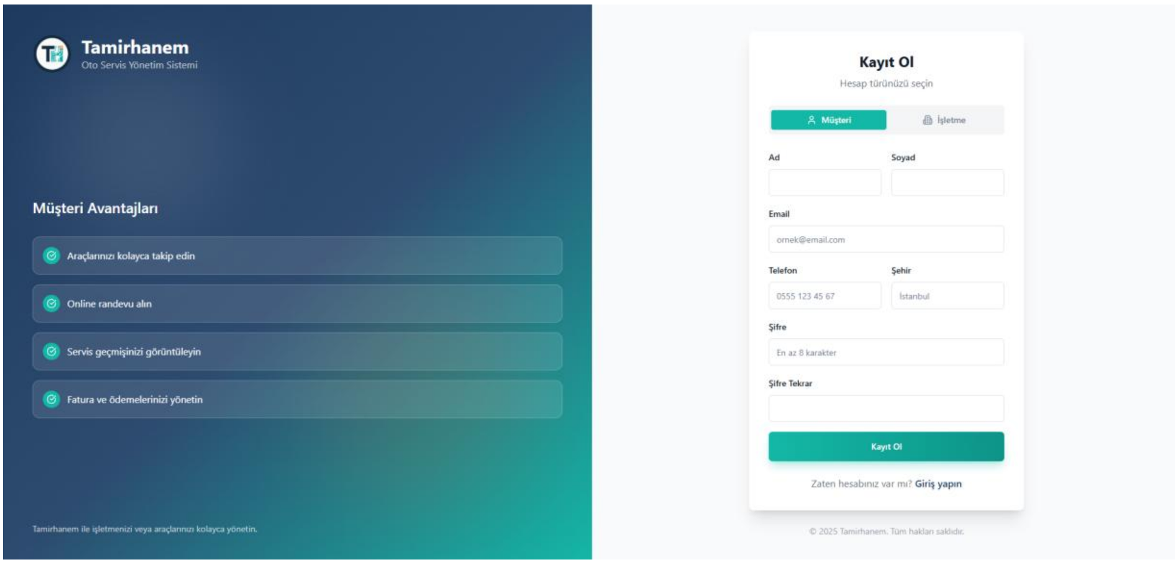
Email Adresi: ornek@email.com

Şifre: ••••••••

Hesabınız yok mu? Kayıt olun

© 2025 Tamirhanem. Tüm hakları saklıdır.

Figure 4.2



Müşteri Avantajları

- Araçlarınızı kolayca takip edin
- Online randevu alın
- Servis geçmişinizi görüntüleyin
- Fatura ve ödemelerinizi yönetin

Tamirhanem ile işletmenizi veya araçlarınızı kolayca yönetin.

Kayıt Ol

Hesap türünüzü seçin

Zaten hesabınız var mı? Giriş yapın

© 2025 Tamirhanem. Tüm hakları saklıdır.

Figure 4.3

4.1.4 State Management

Client-side state management utilizes TanStack Query (formerly React Query) for server state, providing automatic caching, background refetching, and optimistic updates. Local UI state is managed through React's built-in useState and useContext hooks. Custom hooks abstract common data fetching patterns, including useCustomers, useAppointments, useStock, and useInvoices.

4.1.5 Routing and Navigation

React Router DOM handles client-side routing with protected route components that verify authentication status and user roles before rendering protected content. The routing structure separates public routes (landing, login, register), customer routes (/customer/*), business routes (/dashboard/*), and admin routes (/admin/*).

4.2 BACKEND IMPLEMENTATION

The backend is structured following a layered architecture with clear separation between controllers, services, and data access. This structure promotes code reusability, testability, and maintainability.

4.2.1 Controller Layer

The controller layer comprises 14 controller files responsible for handling HTTP requests, extracting request parameters, invoking appropriate services, and formatting responses. Controllers implement input validation using Zod schemas before processing requests. Each controller follows consistent patterns for error handling and response formatting.

4.2.2 Service Layer

The service layer contains 16 service files implementing business logic. Services are responsible for data transformation, business rule enforcement, and coordination of complex operations. The separation of services from controllers allows for easier unit testing and potential reuse across different API endpoints.

4.2.3 Middleware Layer

The middleware layer handles cross-cutting concerns including authentication verification through JWT token validation, role-based authorization through role guard middleware, request logging using Pino logger, and centralized error handling with appropriate HTTP status codes.

4.3 DATABASE IMPLEMENTATION

The database implementation translates the designed schema into a functional PostgreSQL database using Prisma ORM. The Prisma schema file defines all 21 models with their fields, relationships, and constraints.

The User model serves as the central authentication entity, storing email, password hash, role, and status information. One-to-one relationships connect User to either CustomerProfile or IndustrialistProfile based on the user's role. The CustomerProfile model extends User with customer-specific fields including name, contact information, and aggregate statistics such as total visits and total spending.

The IndustrialistProfile model contains comprehensive business information including business name, owner details, contact information, working hours stored as JSON, service offerings, subscription details, and showcase configuration. The showcase feature utilizes JSON fields for flexible data storage of customizable content.

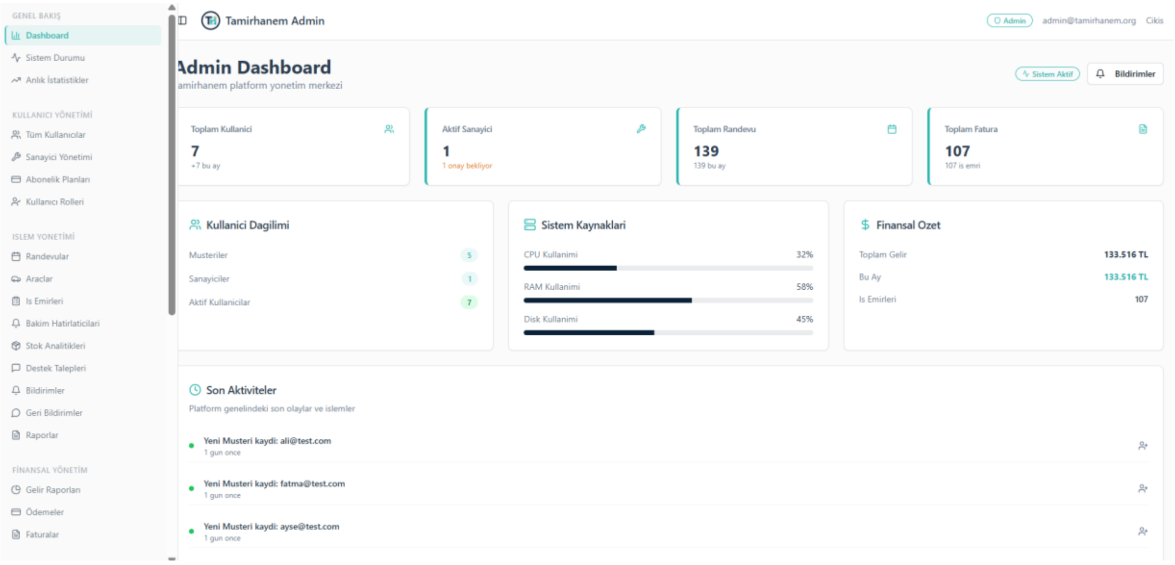
The appointment-to-invoice workflow is supported through carefully designed relationships. An Appointment can generate a WorkOrder, which tracks the service job with associated WorkOrderItems. Upon completion, a WorkOrder can generate an Invoice with InvoiceItems, and Payments are recorded against Invoices.

4.4 ADMIN PANEL FEATURES

The admin panel, developed by Yamaç BEZİRGAN, provides platform administrators with comprehensive tools for managing users, businesses, and platform operations. The panel implements over 60 API endpoints covering all administrative functions.

Table 4.1: Admin Panel Modules

Module	Description	Key Features
Dashboard	Platform overview and statistics	User counts, appointment metrics, revenue summary, activity feed
User Management	User account administration	List, search, filter, status changes, role management
Business Management	Service provider oversight	Verification, approval, suspension, detail editing
Showcase Management	Business showcase approval	Pending approvals, status updates, preview
Appointments	Platform-wide appointment view	Filtering, statistics, status overview
Invoices	Financial document oversight	Status tracking, payment monitoring, statistics
Revenue Reports	Financial analytics	Daily/weekly/monthly charts, top performers
Support Tickets	Customer support management	Ticket queue, responses, assignment, status
Settings	Platform configuration	General settings, contact info, feature toggles



Admin Dashboard

amirhanem platform yonetim merkezi

Toplam Kullanici	Aktif Sanayici	Toplam Randevu	Toplam Fatura
7	1	139	107
+7 bu ay	1 onay bekliyor	139 bu ay	107 is emri

Kullanici Dagilimi

Musteriler	5
Sanayiciler	1
Aktif Kullanicilar	7

Sistem Kaynaklari

CPU Kullanimi	32%
RAM Kullanimi	58%
Disk Kullanimi	45%

Finansal Ozet

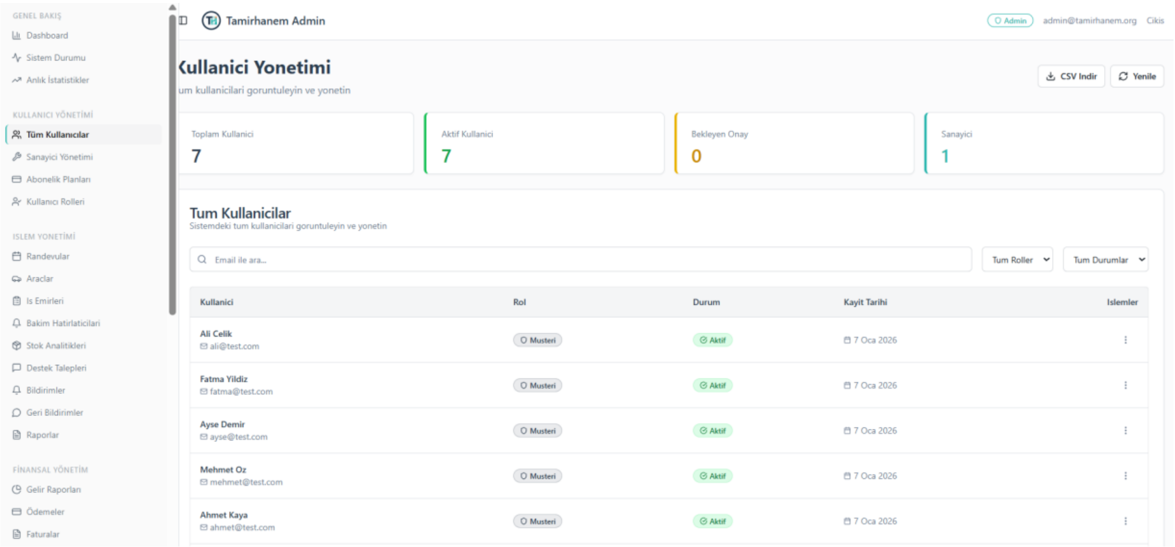
Toplam Gelir	133.516 TL
Bu Ay	133.516 TL
Is Emirleri	107

Son Aktiviteler

Platform genelindeki son olaylar ve islemler

- Yeni Musteri kaydi: ali@test.com
  1 gun once
- Yeni Musteri kaydi: fatma@test.com
  1 gun once
- Yeni Musteri kaydi: ayse@test.com
  1 gun once

Figure 4.4



Kullanici Yonetimi

um kullanicilari goruntuleyin ve yonetin

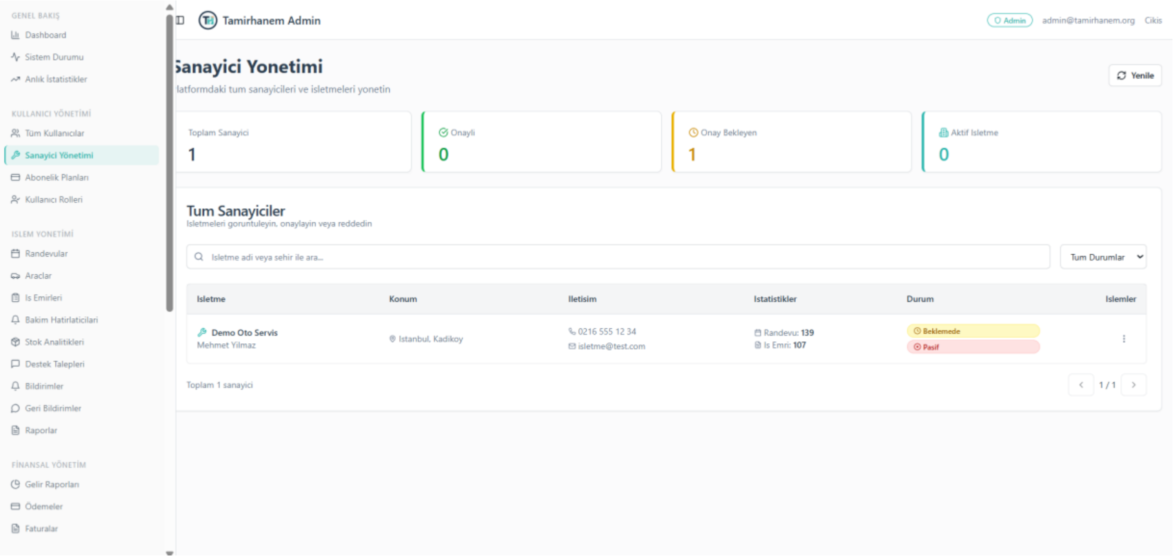
Toplam Kullanici	Aktif Kullanici	Bekleyen Onay	Sanayici
7	7	0	1

Tum Kullanicilar

Sistemdeki tum kullanicilari goruntuleyin ve yonetin

Kullanici	Rol	Durum	Kayit Tarihi	Islemler
Ali Celik ali@test.com	Musteri	Aktif	7 Oca 2026	⋮
Fatma Yildiz fatma@test.com	Musteri	Aktif	7 Oca 2026	⋮
Ayse Demir ayse@test.com	Musteri	Aktif	7 Oca 2026	⋮
Mehmet Oz mehmet@test.com	Musteri	Aktif	7 Oca 2026	⋮
Ahmet Kaya ahmet@test.com	Musteri	Aktif	7 Oca 2026	⋮

Figure 4.5



Sanayici Yonetimi

latformdaki tum sanayicileri ve isletmeleri yonetin

Toplam Sanayici 1

Onayli 0

Onay Bekleyen 1

Aktif Isletme 0

Tum Sanayiciler

Isletmeleri goruntuleyin, onaylayin veya reddedin

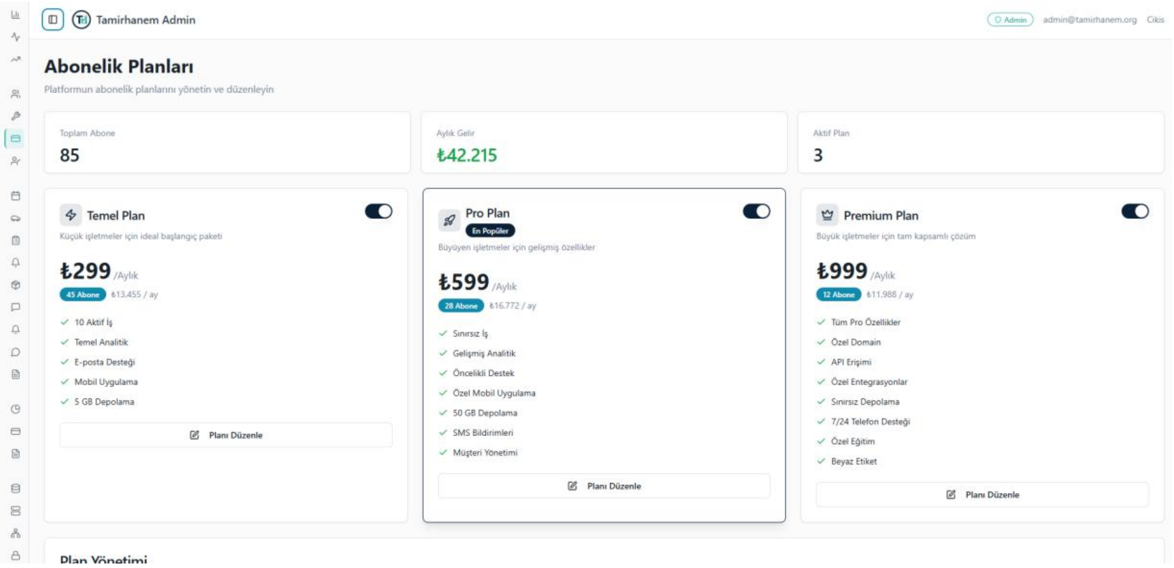
Isletme adi veya sehir ile ara...

Isletme	Konum	Iletisim	Istatistikler	Durum	Islemler
Demo Oto Servis Mehmet Yilmaz	Istanbul, Kadikoy	0216 555 12 34 isletme@test.com	Randevu: 139 Is Emri: 107	Beklemede Pasif	⋮

Toplam 1 sanayici

1 / 1

Figure 4.6



Abonelik Planları

Platformun abonelik planlarını yönetin ve düzenleyin

Toplam Abone 85

Aylık Gelir ₺42.215

Aktif Plan 3

Temel Plan

Küçük işletmeler için ideal başlangıç paketi

₺299 /Aylık

45 Abone ₺13.455 / ay

- ✓ 10 Aktif İş
- ✓ Temel Analitik
- ✓ E-posta Desteği
- ✓ Mobil Uygulama
- ✓ 5 GB Depolama

Pro Plan

En Popüler

Büyüyen işletmeler için gelişmiş özellikler

₺599 /Aylık

28 Abone ₺16.772 / ay

- ✓ Sınırsız İş
- ✓ Gelişmiş Analitik
- ✓ Öncelikli Destek
- ✓ Özel Mobil Uygulama
- ✓ 50 GB Depolama
- ✓ SMS Bildirimleri
- ✓ Müşteri Yönetimi

Premium Plan

Büyük işletmeler için tam kapsamlı çözüm

₺999 /Aylık

12 Abone ₺11.988 / ay

- ✓ Tüm Pro Özellikler
- ✓ Özel Domain
- ✓ API Erişimi
- ✓ Özel Entegrasyonlar
- ✓ Sınırsız Depolama
- ✓ 7/24 Telefon Desteği
- ✓ Özel Eğitim
- ✓ Beyaz Etiket

Plan Yönetimi

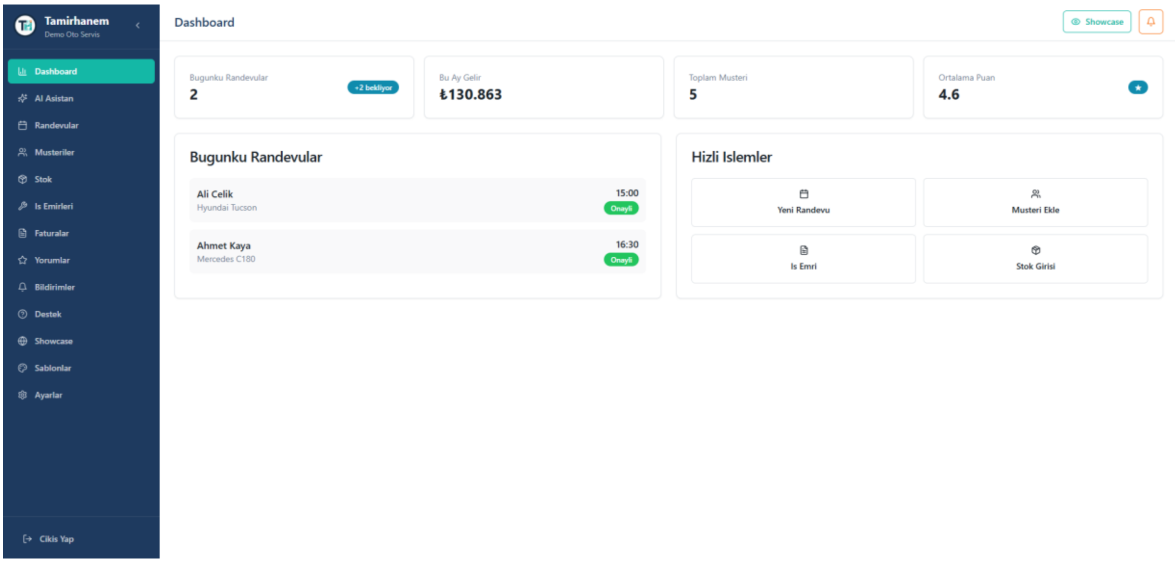
Figure 4.7

4.5 BUSINESS PANEL FEATURES

The business panel, developed by Bekir ÇAKIR, provides auto service operators with comprehensive tools for managing their daily operations. The panel implements CRM, ERP, work order management, and invoicing functionalities.

Table 4.2: Business Panel Modules

Module	Description	Key Features
Dashboard	Business overview	Today's appointments, monthly revenue, customer count, ratings
Appointments - List	Appointment management	Filtering, status updates, creation, customer details
Appointments - Calendar	Visual scheduling	Weekly calendar view, drag-drop, color coding
Customer Management	CRM functionality	Customer database, vehicle history, contact management
Inventory Management	ERP functionality	Stock items, categories, suppliers, movements, alerts
Work Orders	Service job tracking	Creation, parts addition, status tracking, technician assignment
Invoices	Billing management	Invoice creation, PDF generation, payment recording
Reviews	Reputation management	Customer reviews, response capability, visibility control
Showcase	Online presence	Template selection, content editing, preview, publishing



Bugunku Randevular
2
+2 bekliyor

Bu Ay Gelir
₺130.863

Toplam Musteri
5

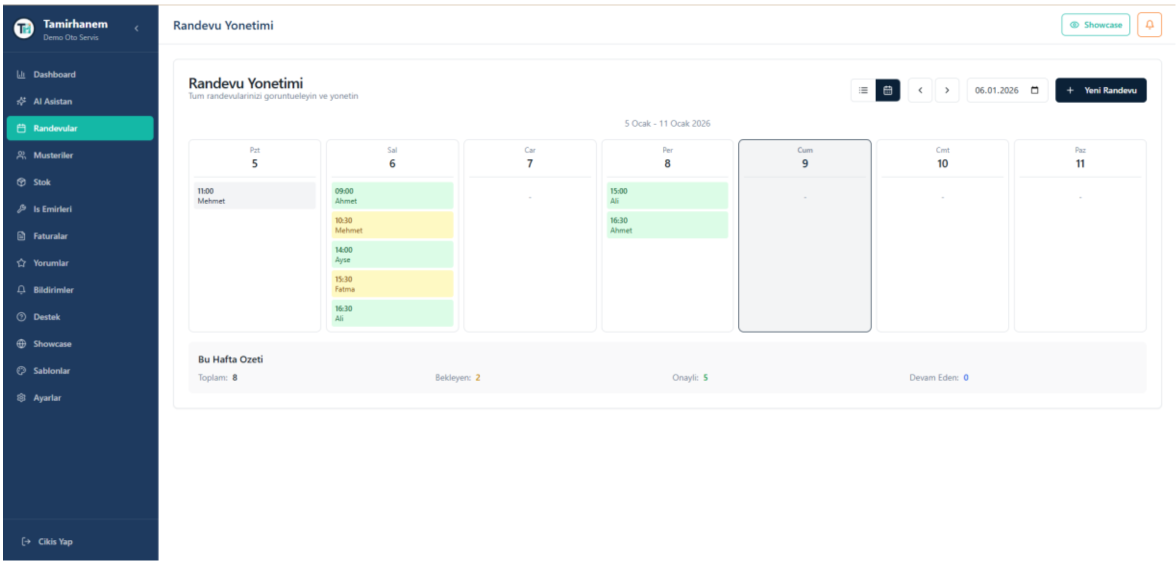
Ortalama Puan
4.6

Bugunku Randevular

Ali Celik
Hyundai Tucson
15:00
Onayli

Ahmet Kaya
Mercedes C180
16:30
Onayli

Figure 4.8



Randevu Yonetimi

Randevu Yonetimi
Tum randevularinizi goruntueleyin ve yonetin

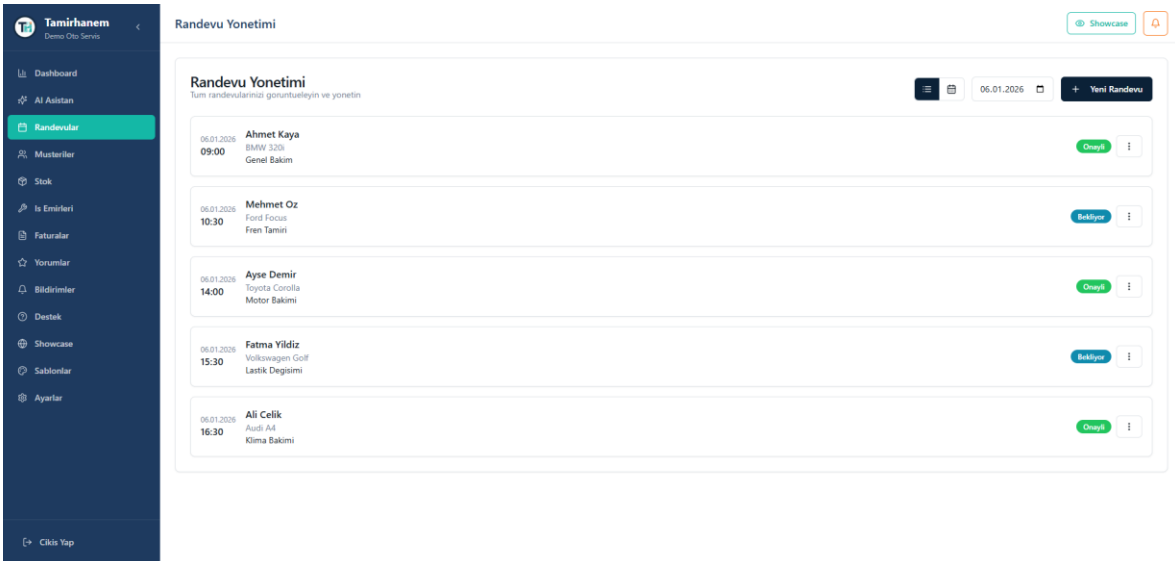
5 Ocak - 11 Ocak 2026

Pzt 5	Sal 6	Car 7	Per 8	Cum 9	Cmt 10	Paz 11
11:00 Mehmet	09:00 Ahmet 10:30 Mehmet 14:00 Ayse 15:30 Fatma 16:30 Ali	-	15:00 Ali 16:30 Ahmet	-	-	-

Bu Hafta Ozeti

Toplam: 8 Bekleyen: 2 Onayli: 5 Devam Eden: 0

Figure 4.9

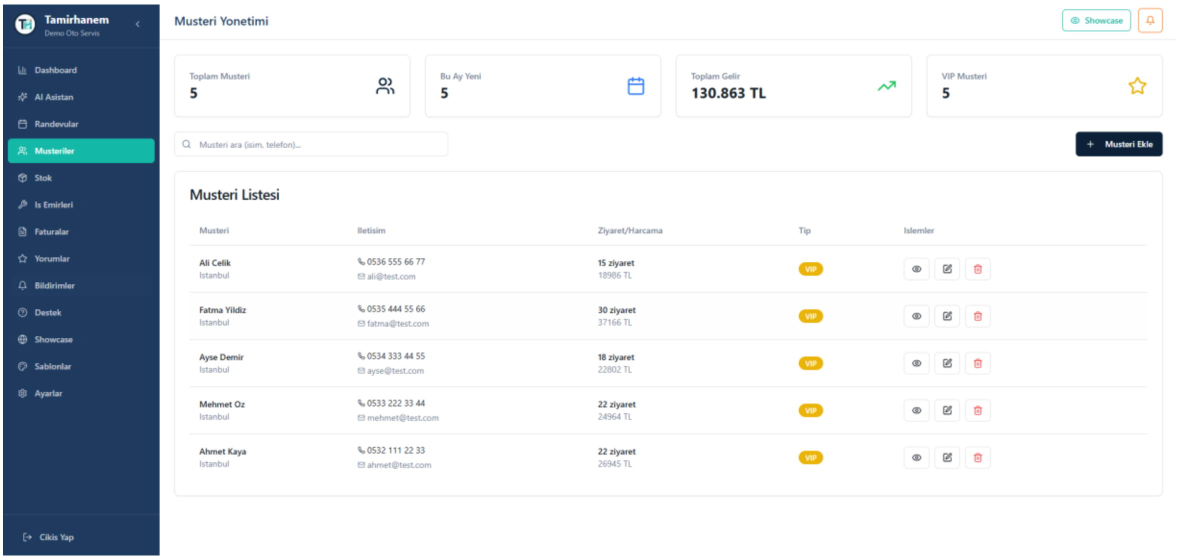


Randevu Yonetimi

Randevu Yonetimi
Tum randevularinizi goruntueleyin ve yonetin

06.01.2026 09:00	Ahmet Kaya BMW 320i Genel Bakim	Onayli
06.01.2026 10:30	Mehmet Oz Ford Focus Fren Tamiri	Bekliyor
06.01.2026 14:00	Ayse Demir Toyota Corolla Motor Bakimi	Onayli
06.01.2026 15:30	Fatma Yildiz Volkswagen Golf Lastik Degisimi	Bekliyor
06.01.2026 16:30	Ali Celik Audi A4 Klima Bakimi	Onayli

Figure 4.10



Musteri Yonetimi

Toplam Musteri 5

Bu Ay Yeni 5

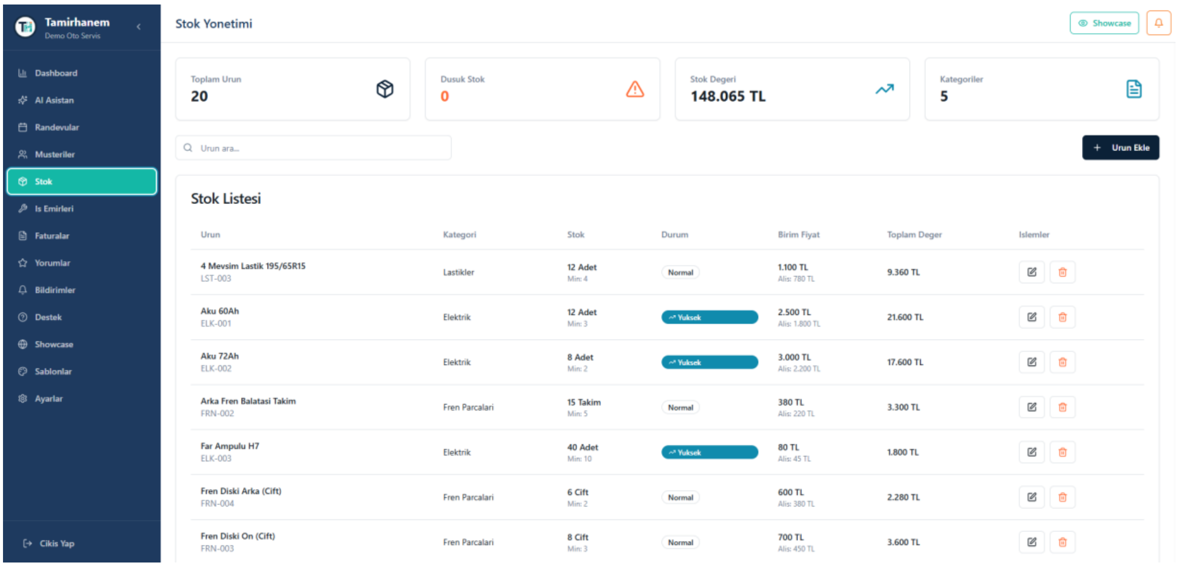
Toplam Gelir 130.863 TL

VIP Musteri 5

Musteri Listesi

Musteri	Iletisim	Ziyaret/Harcama	Tip	Islemler
Ali Celik Istanbul	☎ 0536 555 66 77 ✉ ali@test.com	15 ziyaret 18986 TL	VIP	
Fatma Yildiz Istanbul	☎ 0535 444 55 66 ✉ fatma@test.com	30 ziyaret 37166 TL	VIP	
Ayse Demir Istanbul	☎ 0534 333 44 55 ✉ ayse@test.com	18 ziyaret 22802 TL	VIP	
Mehmet Oz Istanbul	☎ 0533 222 33 44 ✉ mehmet@test.com	22 ziyaret 24964 TL	VIP	
Ahmet Kaya Istanbul	☎ 0532 111 22 33 ✉ ahmet@test.com	22 ziyaret 26945 TL	VIP	

Figure 4.11



Stok Yonetimi

Toplam Urun 20

Dusuk Stok 0

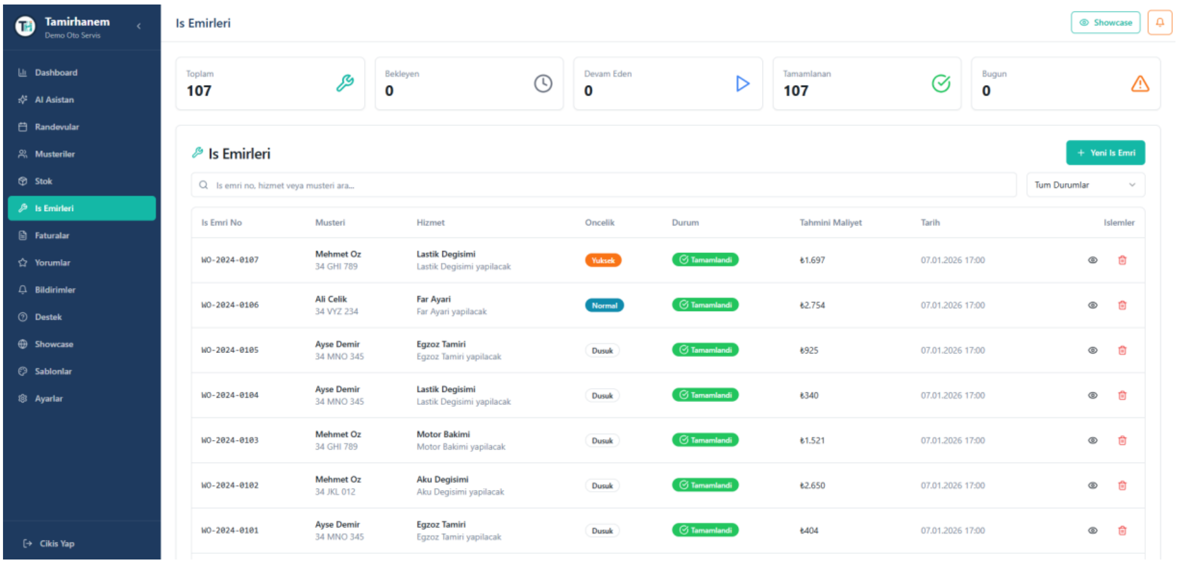
Stok Degeri 148.065 TL

Kategoriler 5

Stok Listesi

Urun	Kategori	Stok	Durum	Birim Fiyat	Toplam Deger	Islemler
4 Mevsim Lastik 195/65R15 LST-003	Lastikler	12 Adet Min: 4	Normal	1.100 TL Alis: 780 TL	9.360 TL	
Aku 60Ah ELK-001	Elektrik	12 Adet Min: 3	Yuksek	2.500 TL Alis: 1.800 TL	21.600 TL	
Aku 72Ah ELK-002	Elektrik	8 Adet Min: 2	Yuksek	3.000 TL Alis: 2.200 TL	17.600 TL	
Arka Fren Balatasi Takim FRN-002	Fren Parcalari	15 Takim Min: 5	Normal	380 TL Alis: 220 TL	3.300 TL	
Far Ampulu H7 ELK-003	Elektrik	40 Adet Min: 10	Yuksek	80 TL Alis: 45 TL	1.800 TL	
Fren Diski Arka (Cift) FRN-004	Fren Parcalari	6 Cift Min: 2	Normal	600 TL Alis: 380 TL	2.280 TL	
Fren Diski On (Cift) FRN-003	Fren Parcalari	8 Cift Min: 3	Normal	700 TL Alis: 450 TL	3.600 TL	

Figure 4.12



Is Emirleri

Toplam 107

Bekleyen 0

Devam Eden 0

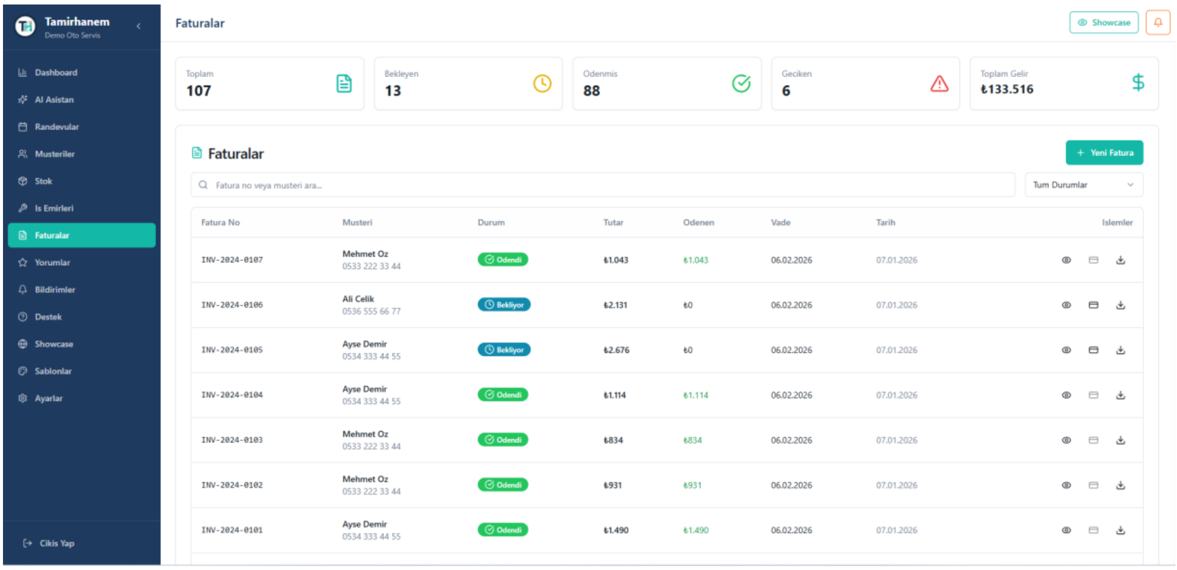
Tamamlanan 107

Bugun 0

Is Emirleri

Is Emri No	Musteri	Hizmet	Oncelik	Durum	Tahmini Maliyet	Tarih	Islemler
WO-2024-0107	Mehmet Oz 34 GHI 789	Lastik Degisimi Lastik Degisimi yapilacak	Yuksek	Tamamlandi	₺1.697	07.01.2026 17:00	
WO-2024-0106	Ali Celik 34 VYZ 234	Far Ayari Far Ayari yapilacak	Normal	Tamamlandi	₺2.754	07.01.2026 17:00	
WO-2024-0105	Ayse Demir 34 MNO 345	Egzoz Tamiri Egzoz Tamiri yapilacak	Dusuk	Tamamlandi	₺925	07.01.2026 17:00	
WO-2024-0104	Ayse Demir 34 MNO 345	Lastik Degisimi Lastik Degisimi yapilacak	Dusuk	Tamamlandi	₺340	07.01.2026 17:00	
WO-2024-0103	Mehmet Oz 34 GHI 789	Motor Bakimi Motor Bakimi yapilacak	Dusuk	Tamamlandi	₺1.521	07.01.2026 17:00	
WO-2024-0102	Mehmet Oz 34 JKL 012	Aku Degisimi Aku Degisimi yapilacak	Dusuk	Tamamlandi	₺2.650	07.01.2026 17:00	
WO-2024-0101	Ayse Demir 34 MNO 345	Egzoz Tamiri Egzoz Tamiri yapilacak	Dusuk	Tamamlandi	₺404	07.01.2026 17:00	

Figure 4.13



Faturalar

Toplam 107

Bekleyen 13

Odenmis 88

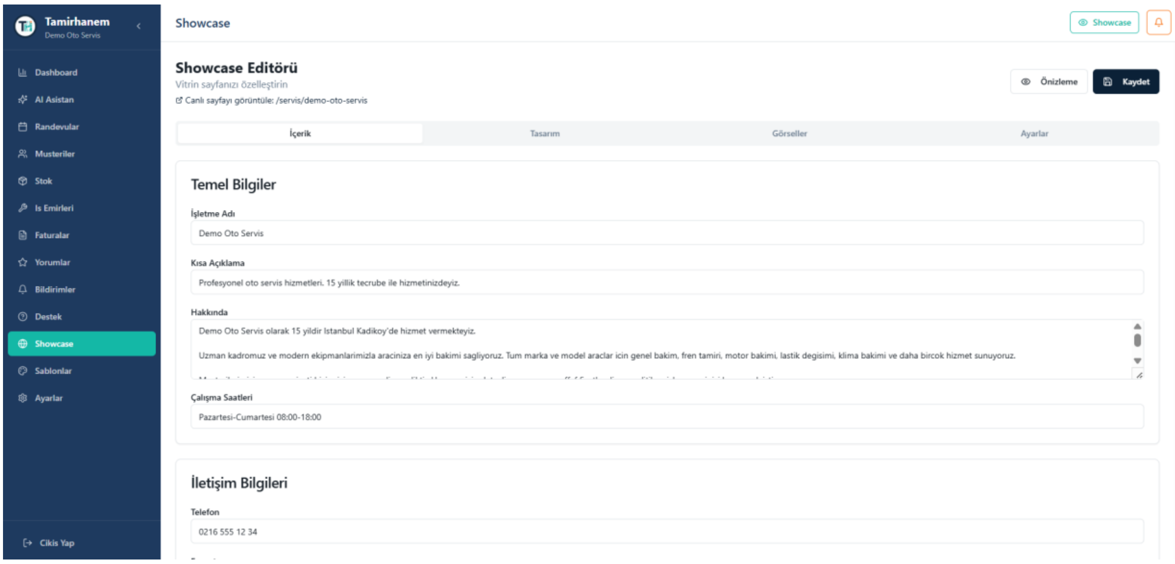
Geciken 6

Toplam Gelir ₺133.516

Faturalar

Fatura No	Musteri	Durum	Tutar	Odenen	Vade	Tarih	Islemler
INV-2024-0107	Mehmet Oz 0533 222 33 44	Odendi	₺1.043	₺1.043	06.02.2026	07.01.2026	
INV-2024-0106	Ali Celik 0536 555 66 77	Bekliyor	₺2.131	₺0	06.02.2026	07.01.2026	
INV-2024-0105	Ayse Demir 0534 333 44 55	Bekliyor	₺2.676	₺0	06.02.2026	07.01.2026	
INV-2024-0104	Ayse Demir 0534 333 44 55	Odendi	₺1.114	₺1.114	06.02.2026	07.01.2026	
INV-2024-0103	Mehmet Oz 0533 222 33 44	Odendi	₺834	₺834	06.02.2026	07.01.2026	
INV-2024-0102	Mehmet Oz 0533 222 33 44	Odendi	₺931	₺931	06.02.2026	07.01.2026	
INV-2024-0101	Ayse Demir 0534 333 44 55	Odendi	₺1.490	₺1.490	06.02.2026	07.01.2026	

Figure 4.14



Showcase Editörü

Vitrin sayfanızı özelleştirin

Canlı sayfayı görüntüle: /servis/demo-oto-servis

Temel Bilgiler

İşletme Adı: Demo Oto Servis

Kısa Açıklama: Profesyonel oto servis hizmetleri. 15 yillik tecrube ile hizmetinizdeyiz.

Hakkında: Demo Oto Servis olarak 15 yildir Istanbul Kadikoy'de hizmet vermekteyiz.

Uzman kadromuz ve modern ekipmanlarimizla araciniza en iyi bakimi sagliyoruz. Tum marka ve model araclar icin genel bakim, fren tamiri, motor bakimi, lastik degisimi, klima bakimi ve daha bircok hizmet sunuyoruz.

Çalışma Saatleri: Pazartesi-Cumartesi 08:00-18:00

İletişim Bilgileri

Telefon: 0216 555 12 34

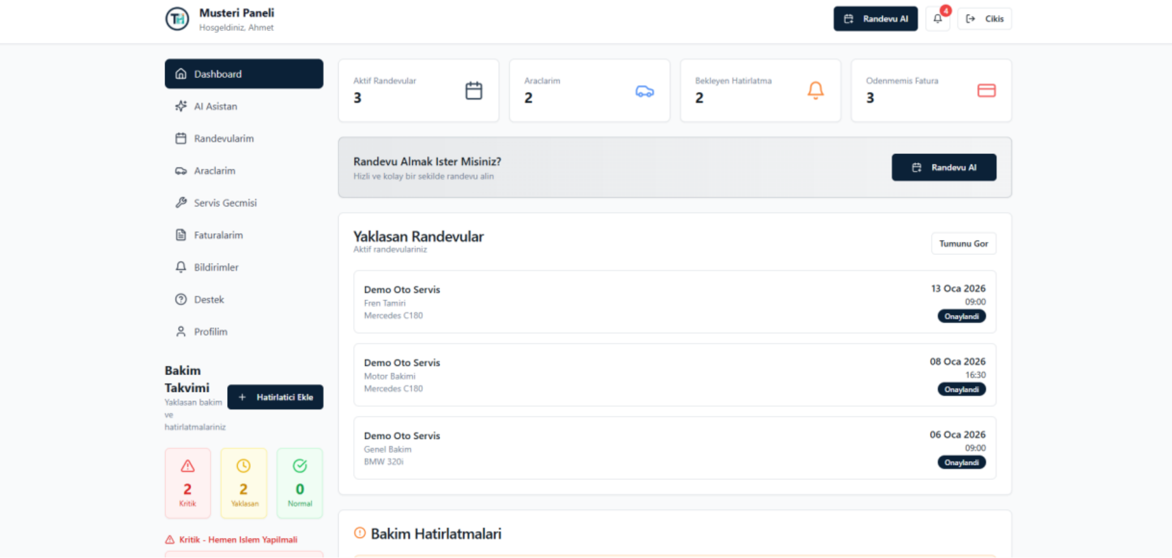
Figure 4.15

4.6 CUSTOMER PANEL FEATURES

The customer panel, developed by Şevval Nur KASAVET, provides vehicle owners with a convenient interface for managing their service appointments, vehicles, and interactions with auto service businesses.

Table 4.3: Customer Panel Modules

Module	Description	Key Features
Dashboard	Personal overview	Active appointments, vehicles, reminders, pending invoices
Appointment Booking	Service scheduling	Business selection, date/time picker, service type, vehicle selection
Appointments	Appointment management	List view, status tracking, cancellation, rescheduling
Vehicles	Vehicle database	Add, edit, delete vehicles, vehicle details
Service History	Past services	Completed services, work details, costs
Invoices	Billing view	Invoice list, status, PDF download
Reminders	Maintenance alerts	Upcoming maintenance, custom reminders
Notifications	Communication center	All notifications, read status, filtering
AI Assistant	Intelligent help	Vehicle questions, recommendations, image analysis
Support	Help requests	Ticket creation, history, responses



Aktif Randevular 3

Araclarim 2

Bekleyen Hatirlatma 2

Odenmemis Fatura 3

Randevu Almak Ister Misiniz?
Hizli ve kolay bir sekilde randevu alin

Yaklasan Randevular
Aktif randevulariniz

Demo Oto Servis Fren Tamiri Mercedes C180	13 Oca 2026 09:00 Onaylandi
Demo Oto Servis Motor Bakimi Mercedes C180	08 Oca 2026 16:30 Onaylandi
Demo Oto Servis Genel Bakim BMW 320i	06 Oca 2026 09:00 Onaylandi

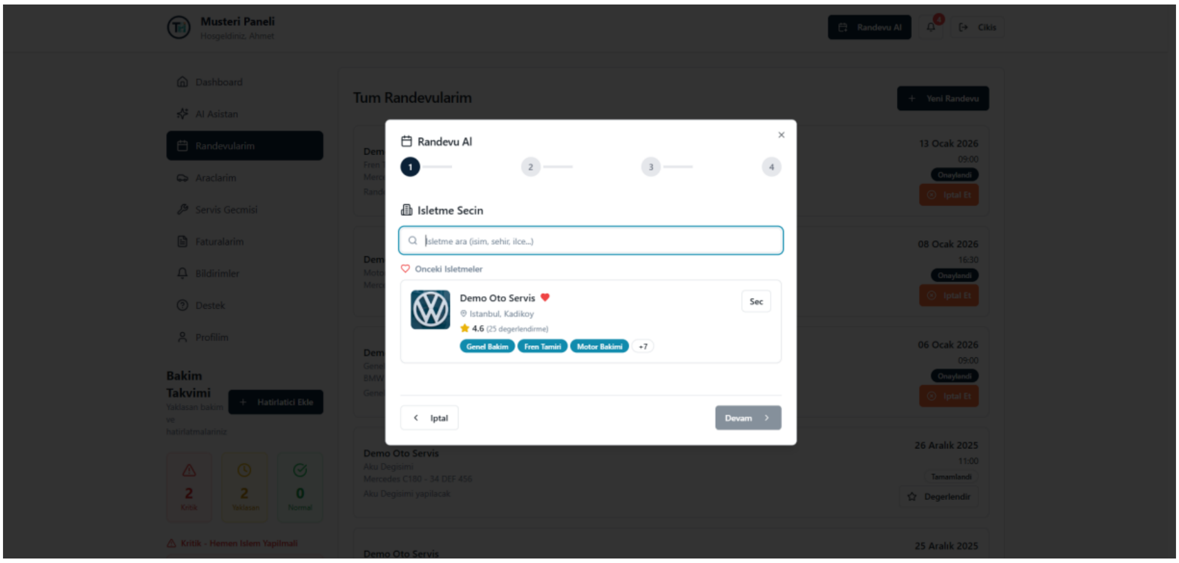
Bakim Takvimi
Yaklasan bakim ve hatirlatmalariniz

2 Kritik | 2 Yaklasan | 0 Normal

Kritik - Hemen Islem Yapilmali

Bakim Hatirlatmalari

Figure 4.16



Randevu Al

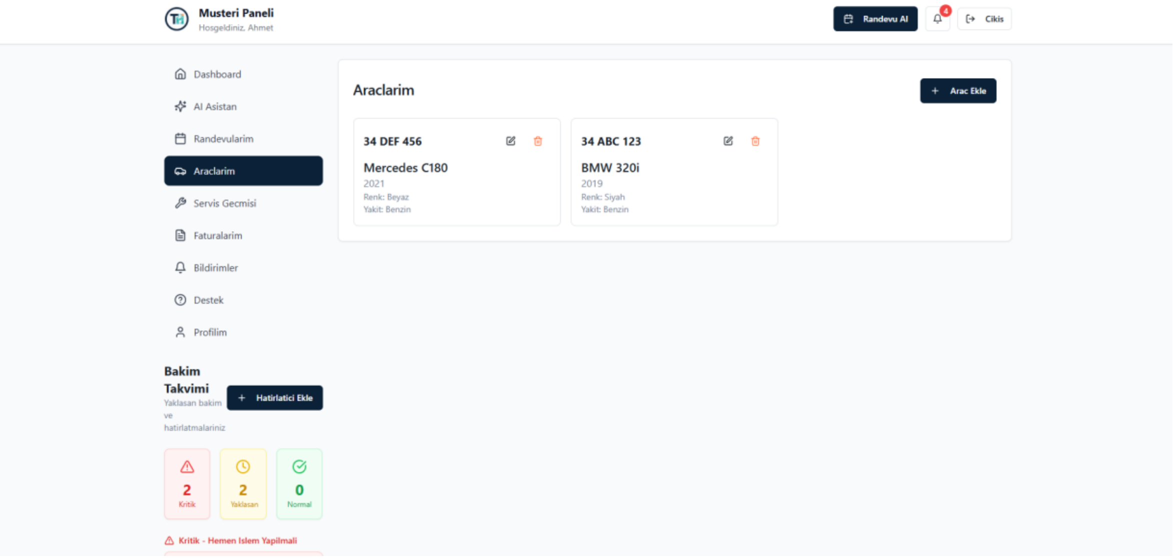
Isletme Secin

Isletme ara (isim, sehir, ilce...)

Onceki Isletmeler

Demo Oto Servis
Istanbul, Kadikoy
4.6 (25 degerlendirme)
Genel Bakim | Fren Tamiri | Motor Bakimi | +7

Figure 4.17



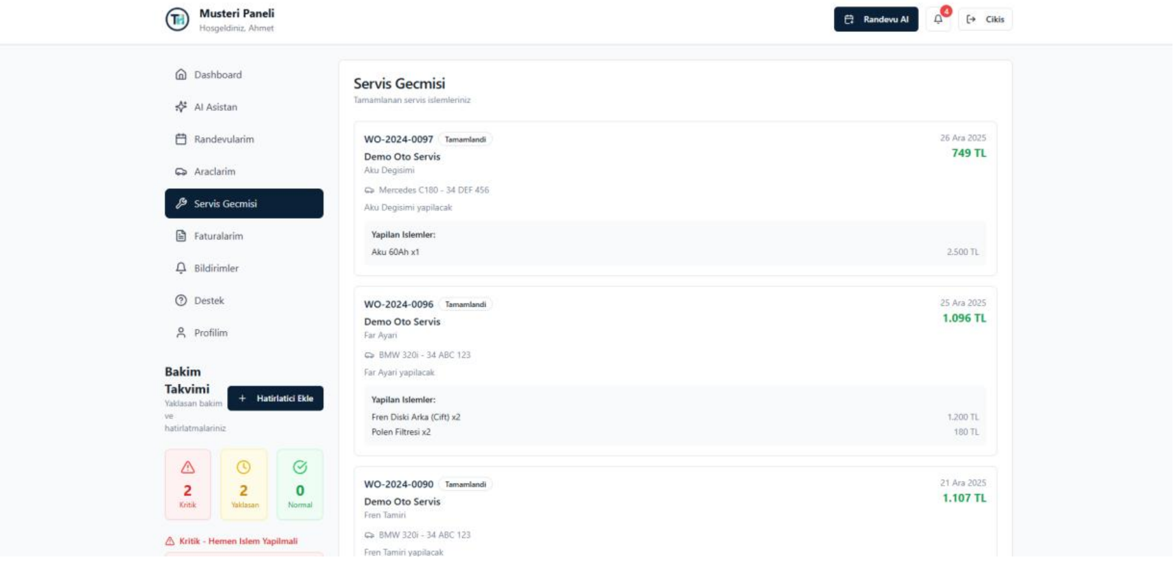
34 DEF 456

Mercedes C180
2021
Renk: Beyaz
Yakit: Benzin

34 ABC 123

BMW 320i
2019
Renk: Siyah
Yakit: Benzin

Figure 4.18



Servis Gecmisi

Tamamlanan servis islemleriniz

WO-2024-0097 Tamamlandi — 26 Ara 2025 — 749 TL
Demo Oto Servis
Aku Degisimi
Mercedes C180 - 34 DEF 456
Aku Degisimi yapilacak

Yapilan Islemler:

Aku 60Ah x1	2.500 TL

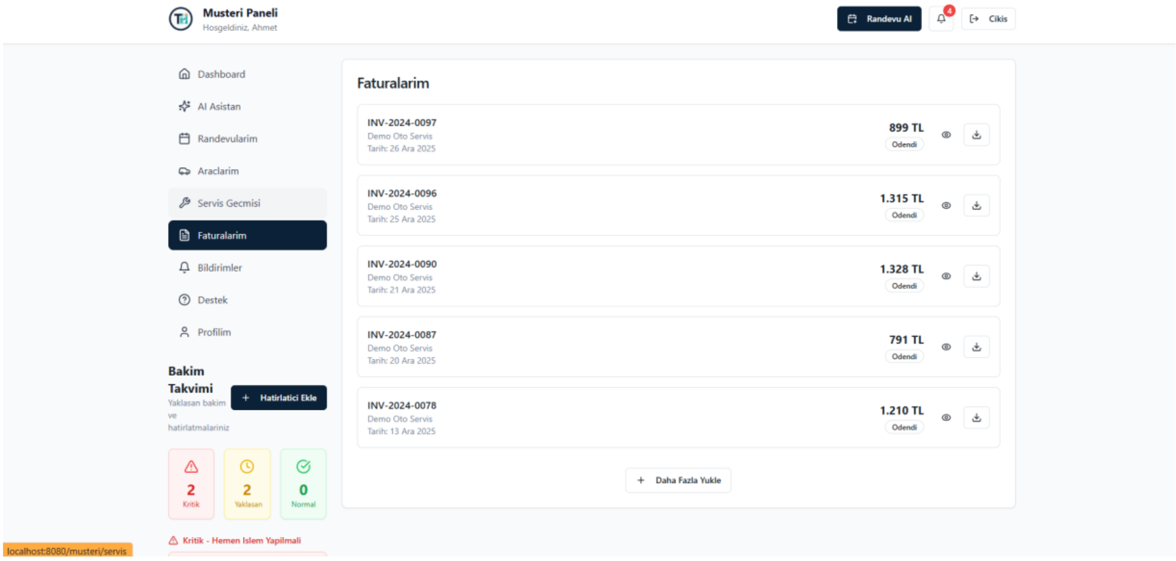
WO-2024-0096 Tamamlandi — 25 Ara 2025 — 1.096 TL
Demo Oto Servis
Far Ayari
BMW 320i - 34 ABC 123
Far Ayari yapilacak

Yapilan Islemler:

Fren Diski Arka (Cift) x2	1.200 TL
Polen Filtresi x2	180 TL

WO-2024-0090 Tamamlandi — 21 Ara 2025 — 1.107 TL
Demo Oto Servis
Fren Tamiri
BMW 320i - 34 ABC 123
Fren Tamiri yapilacak

Figure 4.19



Faturalarim

INV-2024-0097
Demo Oto Servis
Tarih: 26 Ara 2025
899 TL
Odendi

INV-2024-0096
Demo Oto Servis
Tarih: 25 Ara 2025
1.315 TL
Odendi

INV-2024-0090
Demo Oto Servis
Tarih: 21 Ara 2025
1.328 TL
Odendi

INV-2024-0087
Demo Oto Servis
Tarih: 20 Ara 2025
791 TL
Odendi

INV-2024-0078
Demo Oto Servis
Tarih: 13 Ara 2025
1.210 TL
Odendi

Bakim Takvimi
Yaklasan bakim ve hatirlatmalariniz

2 Kritik
2 Yaklasan
0 Normal

Kritik - Hemen Islem Yapilmali

Figure 4.20

4.7 AI ASSISTANT IMPLEMENTATION

The AI assistant feature represents an innovative aspect of the Tamirhanem platform, utilizing Google's Gemini API to provide intelligent support for both customers and business operators.

4.7.1 Customer AI Assistant

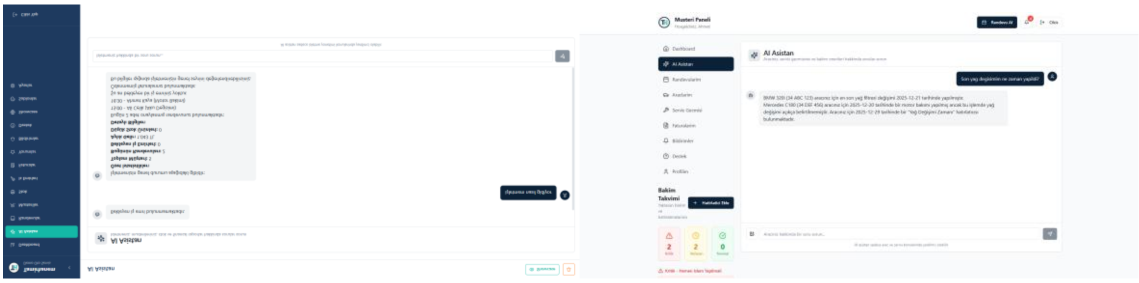
The customer AI assistant is designed to answer vehicle-related questions, provide maintenance recommendations based on service history, explain service requirements, and perform preliminary visual analysis of vehicle images to identify potential issues. The assistant maintains context awareness by accessing the customer's vehicle information, appointment history, and service records.

4.7.2 Business AI Assistant

The business AI assistant supports service operators with operational queries including customer information lookup, work order and inventory status reports, financial summaries, appointment information, unpaid invoice alerts, and low stock warnings. The assistant can synthesize data from multiple sources to provide comprehensive business insights.

4.7.3 Technical Implementation

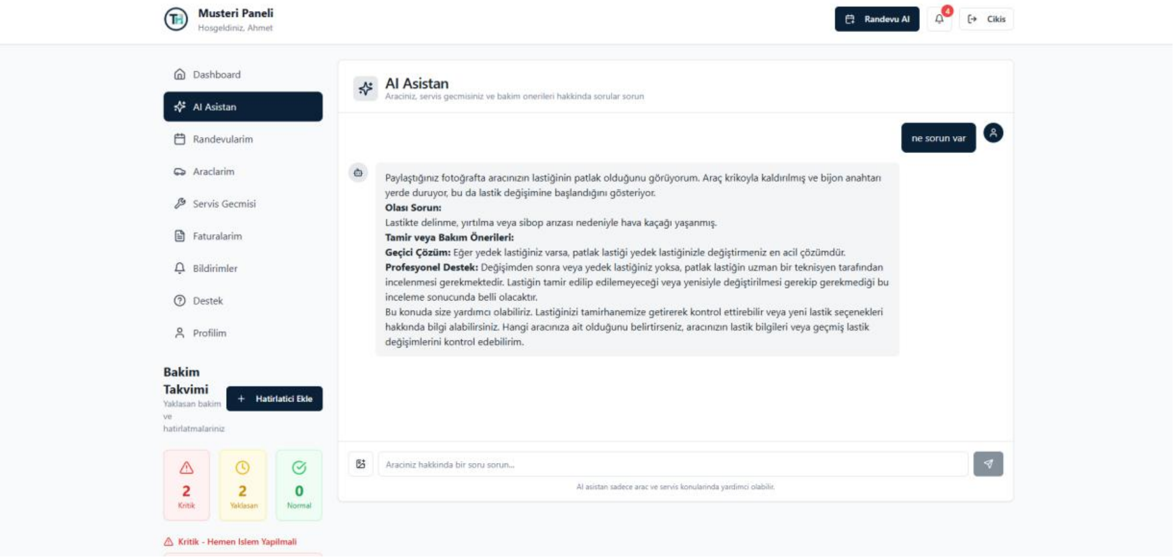
The AI implementation comprises two main service components. The AI Assistant Service handles communication with the Gemini API, managing prompt construction, response parsing, and rate limiting (15 requests per minute). The AI Context Service retrieves relevant data from the database to provide context for AI responses, ensuring that answers are personalized and accurate.



Son yag degisimim ne zaman yapildi?

BMW 320i (34 ABC 123) aracınız için en son yağ filtresi değişimi 2025-12-21 tarihinde yapılmıştır. Mercedes C180 (34 DEF 456) aracınız için 2025-12-20 tarihinde bir motor bakımı yapılmış ancak bu işlemde yağ değişimi açıkça belirtilmemiştir. Aracınız için 2025-12-29 tarihinde bir "Yağ Değişimi Zamanı" hatırlatıcısı bulunmaktadır.

Figure 4.21



Musteri Paneli
Hosgeldiniz, Ahmet
Randevu Al
4
Cikis
Dashboard
AI Asistan
Randevularim
Araclarim
Servis Gecmisi
Faturalarim
Bildirimler
Destek
Profilim
Bakim Takvimi
Yaklasan bakim ve hatirlatmalariniz
+ Hatirlatici Ekle
2
Kritik
2
Yaklasan
0
Normal
Kritik - Hemen Islem Yapilmali
AI Asistan
Araciniz, servis gecmisiniz ve bakim onerileri hakkinda sorular sorun
Araciniz hakkinda bir soru sorun...
AI asistan sadece arac ve servis konularinda yardimci olabilir.

ne sorun var

Paylaştığınız fotoğrafta aracınızın lastiğinin patlak olduğunu görüyorum. Araç krikoyla kaldırılmış ve bijon anahtarı yerde duruyor, bu da lastik değişimine başlandığını gösteriyor.
Olası Sorun:
Lastikte delinme, yırtılma veya sibop arızası nedeniyle hava kaçağı yaşanmış.
Tamir veya Bakım Önerileri:
Geçici Çözüm: Eğer yedek lastiğiniz varsa, patlak lastiği yedek lastiğinizle değiştirmeniz en acil çözümdür.
Profesyonel Destek: Değişimden sonra veya yedek lastiğiniz yoksa, patlak lastiğin uzman bir teknisyen tarafından incelenmesi gerekmektedir. Lastiğin tamir edilip edilemeyeceği veya yenisiyle değiştirilmesi gerekip gerekmediği bu inceleme sonucunda belli olacaktır.
Bu konuda size yardımcı olabiliriz. Lastiğinizi tamirhanemize getirerek kontrol ettirebilir veya yeni lastik seçenekleri hakkında bilgi alabilirsiniz. Hangi aracınıza ait olduğunu belirtirseniz, aracınızın lastik bilgileri veya geçmiş lastik değişimlerini kontrol edebilirim.

Figure 4.22

5. TESTING AND EVALUATION

This chapter presents the testing strategy employed for the Tamirhanem platform and evaluates the system's performance against defined metrics and requirements.

5.1 TESTING STRATEGY

Quality assurance for Tamirhanem followed a practical testing approach suited to our team size and timeline. Şevval Nur KASAVET coordinated systematic testing across all three panels, documenting test cases in GitHub Issues and verifying fixes through follow-up testing. Each major feature underwent functional verification, cross-browser checks, and responsive layout testing before merging to the main branch.

Test cases were documented and tracked using GitHub Issues, enabling systematic coverage of all platform features. Code reviews were conducted for all pull requests, providing an additional layer of quality control before code integration.

5.2 FUNCTIONAL TESTING

Functional testing verified that all platform features operate according to specifications. Test scenarios covered the complete user journey for each role, from registration through core functionality usage.

Table 5.1: Test Scenarios and Results

Category	Test Scenario	Result
Registration	Customer registration with valid data	PASS
Registration	Business registration with valid data	PASS
Registration	Registration with invalid email format	PASS
Registration	Registration with weak password	PASS
Authentication	Login with correct credentials	PASS
Authentication	Login with incorrect password	PASS
Authentication	Token refresh on expiry	PASS
Authentication	Logout and token invalidation	PASS
Appointments	Create appointment with valid data	PASS

Appointments	Update appointment status	PASS
Appointments	Cancel appointment	PASS
Work Orders	Create work order from appointment	PASS
Work Orders	Add items to work order	PASS
Work Orders	Complete work order	PASS
Invoices	Generate invoice from work order	PASS
Invoices	Download invoice PDF	PASS
Invoices	Record payment	PASS
Stock	Add stock item	PASS
Stock	Record stock movement	PASS
AI Assistant	Customer chat query	PASS
AI Assistant	Image analysis request	PASS

5.3 USER INTERFACE TESTING

User interface testing ensured consistent functionality and appearance across different browsers and device sizes.

5.3.1 Cross-Browser Testing

The platform was tested on major web browsers including Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge. All core functionalities operated correctly across these browsers, with minor visual variations addressed through CSS normalization.

5.3.2 Responsive Design Testing

Responsive design testing verified the platform's usability across different screen sizes: desktop (1920x1080), laptop (1366x768), tablet (768x1024), and mobile (375x667). The UI adapts appropriately to each viewport, maintaining usability and visual appeal.

5.3.3 Accessibility Testing

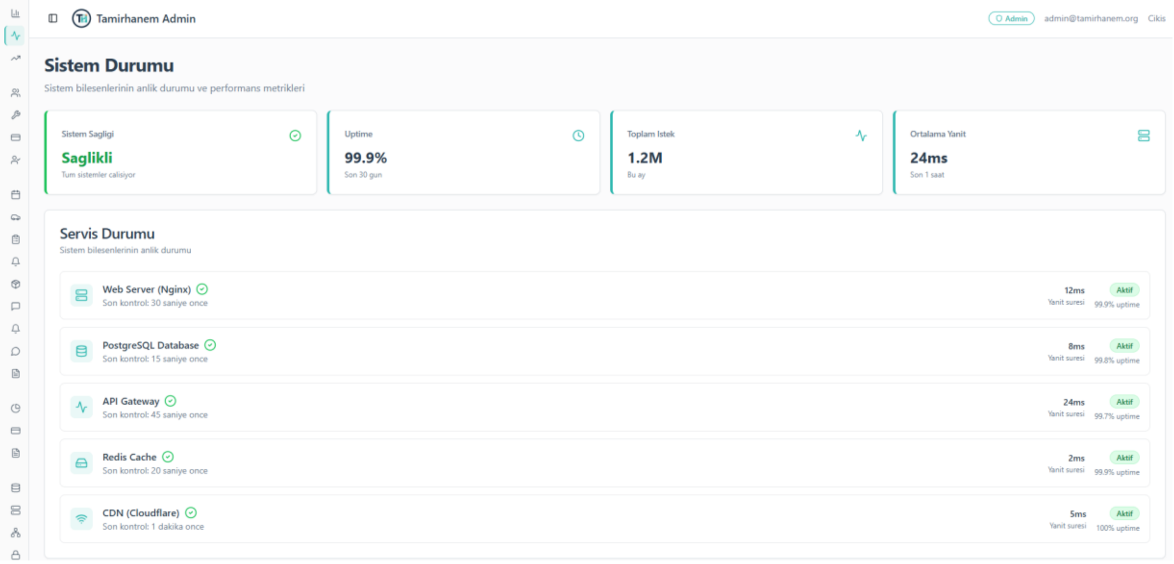
Basic accessibility testing was conducted to ensure keyboard navigation functionality, adequate color contrast ratios, and semantic HTML structure. The use of shadcn/ui components built on Radix UI primitives provides inherent accessibility features including ARIA attributes and keyboard interaction patterns.

5.4 PERFORMANCE EVALUATION

Performance testing measured the system's responsiveness and efficiency against defined targets.

Table 5.2: Performance Metrics

Metric	Target	Actual	Status
API Response Time (average)	< 200ms	~150ms	ACHIEVED
Frontend Initial Load	< 3 seconds	~2.5 seconds	ACHIEVED
Database Query Time	< 100ms	~50ms	ACHIEVED
PDF Generation Time	< 2 seconds	~1.5 seconds	ACHIEVED
AI Response Time	< 5 seconds	~3 seconds	ACHIEVED



Sistem Durumu

Sistem bilesenlerinin anlik durumu ve performans metrikleri

Sistem Sagligi	Uptime	Toplam Istek	Ortalama Yanit
Saglikli	99.9%	1.2M	24ms
Tum sistemler calisiyor	Son 30 gun	Bu ay	Son 1 saat

Servis Durumu

Sistem bilesenlerinin anlik durumu

Web Server (Nginx) ✓ Son kontrol: 30 saniye once	12ms Yanit suresi	Aktif 99.9% uptime
PostgreSQL Database ✓ Son kontrol: 15 saniye once	8ms Yanit suresi	Aktif 99.8% uptime
API Gateway ✓ Son kontrol: 45 saniye once	24ms Yanit suresi	Aktif 99.7% uptime
Redis Cache ✓ Son kontrol: 20 saniye once	2ms Yanit suresi	Aktif 99.9% uptime
CDN (Cloudflare) ✓ Son kontrol: 1 dakika once	5ms Yanit suresi	Aktif 100% uptime

Figure 5.1

6. CONCLUSION AND FUTURE WORK

This chapter summarizes the achievements of the Tamirhanem project, acknowledges its limitations, and outlines directions for future development.

6.1 PROJECT SUMMARY

The Tamirhanem project successfully delivered a comprehensive web-based platform for auto service management, addressing the identified challenges of digital transformation in the Turkish automotive service industry. The platform serves three distinct user roles with dedicated interfaces and functionality.

The development effort resulted in a fully functional system comprising over 200 API endpoints, 21 database tables, 141 frontend components, and integration with Google's Gemini AI for intelligent assistance. The platform demonstrates the viability of modern web technologies for creating sophisticated business management solutions.

6.2 ACHIEVEMENTS

Table 6.1 summarizes the achievement of project objectives defined in Chapter 1.

Table 6.1: Project Achievement Summary

Objective	Status	Notes
Digital appointment management system	COMPLETED	List and calendar views, status management
Customer Relationship Management (CRM)	COMPLETED	Customer database, vehicle history, contact tracking
Enterprise Resource Planning (ERP)	COMPLETED	Inventory, suppliers, stock movements, alerts
Work order management system	COMPLETED	Full lifecycle tracking, parts integration
Invoicing with PDF generation	COMPLETED	Automatic calculation, PDF export, payments
Business showcase pages	COMPLETED	Three templates, customization, publishing
AI-powered assistant	COMPLETED	Customer and business support, image analysis

Secure platform architecture	COMPLETED	JWT auth, RBAC, input validation, security headers

6.3 LIMITATIONS

Despite the successful implementation of core features, the current version of Tamirhanem has several limitations that should be acknowledged.

The platform currently lacks native mobile applications for iOS and Android. While the web interface is responsive and mobile-friendly, dedicated mobile apps would provide better user experience and access to device-specific features such as push notifications and camera integration.

Online payment integration is not included in the current version. Businesses must record payments manually, and customers cannot pay invoices directly through the platform. Integration with Turkish payment providers such as iyzico or PayTR would enhance the platform's utility.

SMS notification capabilities are absent, limiting the platform's ability to reach customers who may not regularly check email or platform notifications. Integration with SMS gateways would improve communication effectiveness.

The AI assistant, while functional, operates within rate limits and may experience latency during peak usage. The image analysis feature should be considered a preliminary assessment tool rather than a definitive diagnostic system.

6.4 FUTURE WORK

Based on the identified limitations and stakeholder feedback, the following areas are recommended for future development.

Mobile Application Development: Development of native mobile applications using React Native would provide a consistent codebase while enabling platform-specific optimizations. Push notifications, offline capabilities, and camera integration would enhance the mobile user experience.

Payment Gateway Integration: Integration with Turkish payment providers would enable online invoice payment, automatic payment tracking, and potentially recurring payment support for subscription services.

SMS Gateway Integration: Adding SMS notification capabilities through providers such as Netgsm would improve communication reach, particularly for appointment reminders and urgent notifications.

Multi-Language Support: Implementation of internationalization (i18n) infrastructure would enable the platform to serve diverse user populations and potentially expand to international markets.

Advanced Analytics: Development of predictive maintenance algorithms, customer lifetime value calculations, and business benchmarking features would provide additional value for business operators seeking data-driven insights.

REFERENCES

[1] Turkish Automotive Aftermarket Association, "Turkish Automotive Aftermarket Report 2023," TAYSAD Publications, Istanbul, 2023.

[2] A. Yılmaz and B. Kaya, "Digital Transformation Challenges in Turkish SMEs: A Case Study of Automotive Service Sector," Journal of Business Research Turkey, vol. 15, no. 2, pp. 45-62, 2022.

[3] McKinsey & Company, "The Future of Automotive Aftermarket Services," McKinsey Insights, 2022. [Online]. Available: https://www.mckinsey.com/industries/automotive-and-assembly/

[4] O. Demir, "Impact of COVID-19 on Automotive Service Industry Digital Adoption," Transportation Research Procedia, vol. 58, pp. 234-241, 2021.

[5] Shop-Ware Inc., "Shop-Ware Auto Repair Software Documentation," 2024. [Online]. Available: https://www.shop-ware.com/

[6] Mitchell 1, "ProDemand and Manager SE Product Guide," Mitchell International, 2024.

[7] Market Research Turkey, "Automotive Service Software Market Analysis," MRT Publications, Istanbul, 2023.

[8] Meta Platforms, "React Documentation," 2024. [Online]. Available: https://react.dev/

[9] Z. Gao, C. Bird, and E. T. Barr, "To Type or Not to Type: Quantifying Detectable Bugs in JavaScript," in Proc. IEEE/ACM International Conference on Software Engineering (ICSE), pp. 758-769, 2017.

[10] A. Wathan and S. Schoger, "Tailwind CSS Documentation," Tailwind Labs, 2024. [Online]. Available: https://tailwindcss.com/docs

[11] Node.js Foundation, "Node.js Documentation," 2024. [Online]. Available: https://nodejs.org/docs/

[12] Express.js, "Express.js Guide," 2024. [Online]. Available: https://expressjs.com/

[13] Prisma, "Prisma Documentation," Prisma Data, Inc., 2024. [Online]. Available: https://www.prisma.io/docs/

[14] PostgreSQL Global Development Group, "PostgreSQL Documentation," 2024. [Online]. Available: https://www.postgresql.org/docs/

[15] J. Wei et al., "Emergent Abilities of Large Language Models," Transactions on Machine Learning Research, 2022.

[16] Google, "Gemini API Documentation," Google AI, 2024. [Online]. Available: https://ai.google.dev/docs

[17] Gartner, "AI in Customer Service: Market Guide," Gartner Research, 2023.

[18] S. Kumar and R. Singh, "Computer Vision Applications in Automotive Diagnostics: A Review," IEEE Access, vol. 11, pp. 45678-45692, 2023.

[19] K. Schwaber and J. Sutherland, "The Scrum Guide," Scrum.org, 2020. [Online]. Available: https://scrumguides.org/

APPENDIX A

DATABASE SCHEMA DETAILS

This appendix provides detailed information about the database schema used in the Tamirhanem platform.

A.1 USER MODEL

The User model is the central authentication entity. Fields: id (UUID, Primary Key), email (String, Unique), passwordHash (String), role (UserRole enum: ADMIN, INDUSTRIALIST, CUSTOMER), status (UserStatus enum: ACTIVE, INACTIVE, PENDING, SUSPENDED), emailVerified (Boolean), lastLogin (DateTime, nullable), createdAt (DateTime), updatedAt (DateTime).

A.2 CUSTOMER PROFILE MODEL

The CustomerProfile model extends User with customer-specific information. Fields: id (UUID, Primary Key), userId (Foreign Key to User, Unique), firstName (String), lastName (String), phone (String, nullable), address (String, nullable), city (String, nullable), notes (String, nullable), totalVisits (Int), totalSpent (Decimal), rating (Decimal), createdAt (DateTime), updatedAt (DateTime).

A.3 INDUSTRIALIST PROFILE MODEL

The IndustrialistProfile model contains business information for service providers. Key fields include: businessName, ownerName, phone, email, address, city, district, taxNumber, description, logoUrl, coverImageUrl, website, workingHours (JSON), services (String array), subscriptionPlan, subscriptionStatus, slug (unique URL identifier), showcaseEnabled, showcaseTemplate, showcaseStatus, showcaseData (JSON), averageRating, and totalReviews.

A.4 COMPLETE TABLE LIST

The complete database schema includes 21 tables: User, CustomerProfile, IndustrialistProfile, Vehicle, Appointment, Category, Supplier, StockItem, StockMovement, WorkOrder, WorkOrderItem, Invoice, InvoiceItem, Payment, RefreshToken, Review, MaintenanceReminder, Notification, SupportTicket, TicketResponse, and Setting.

APPENDIX B

API ENDPOINTS REFERENCE

This appendix provides a reference list of the API endpoints implemented in the Tamirhanem platform.

B.1 AUTHENTICATION ENDPOINTS (/api/auth)

POST /api/auth/register - User registration

POST /api/auth/login - User authentication

POST /api/auth/refresh - Token refresh

POST /api/auth/logout - User logout

GET /api/auth/me - Current user profile

PUT /api/auth/profile - Update profile

B.2 APPOINTMENT ENDPOINTS (/api/appointments)

GET /api/appointments - List all appointments

POST /api/appointments - Create appointment

GET /api/appointments/:id - Get appointment details

PATCH /api/appointments/:id/status - Update status

GET /api/appointments/stats - Appointment statistics

B.3 AI ASSISTANT ENDPOINTS (/api/ai)

POST /api/ai/customer/chat - Customer AI chat

POST /api/ai/customer/chat-with-image - Chat with image analysis

POST /api/ai/industrialist/chat - Business AI chat

APPENDIX C

TEAM RESPONSIBILITIES MATRIX

This appendix provides a detailed breakdown of the responsibilities assigned to each team member during the project development.

C.1 ŞEVVAL NUR KASAVET (210502087)

Primary Responsibility: Customer Panel Development and Quality Assurance. Developed all customer-facing modules including dashboard, appointment booking, vehicle management, service history, invoice viewing, maintenance reminders, notification center, profile management, AI assistant interface, and support ticket system. Led quality assurance activities including functional testing, UI/UX testing, cross-browser testing, responsive testing, user flow testing, and regression testing.

C.2 YAMAÇ BEZİRGAN (200502023)

Primary Responsibility: Admin Panel and Backend Infrastructure. Developed the complete admin panel including dashboard, user management, business management, showcase management, appointment overview, invoice management, revenue reports, support tickets, notification management, and platform settings. Designed and implemented the backend architecture including RESTful API design, JWT authentication system, role-based access control, Prisma schema design (21 tables), Express middleware chain, error handling standards, and security configurations.

C.3 BEKİR ÇAKIR (210502084)

Primary Responsibility: Business (Industrialist) Panel Development. Developed all business panel modules including dashboard, appointment management (list and calendar views), customer management (CRM), inventory management (ERP), work order system, invoice management with PDF generation, review management, notification center, showcase editor, and settings. Implemented backend services for work orders, invoices, stock management, and review handling.

C.4 SHARED RESPONSIBILITIES

All team members collaborated on: project architecture decisions, technology stack selection, coding standards establishment, Git workflow and branch strategy, code review processes, UI/UX design decisions, landing page development, login/register page development, component library selection and customization, and responsive design implementation.