

# TopGear Philippines Clone - Software Requirements Specification

Development Team

2024-11-13

## Contents

<b>1</b>	<b>TopGear Philippines Clone - Software Requirements Specification</b>	<b>2</b>
1.1	1. Introduction	2
1.1.1	1.1 Purpose	2
1.1.2	1.2 Project Scope	2
1.1.3	1.3 Technology Stack	2
1.2	2. System Architecture	2
1.2.1	2.1 Frontend Architecture	2
1.2.2	2.2 Backend Architecture	2
1.2.3	2.3 CMS Architecture	2
1.3	3. Feature Requirements	2
1.3.1	3.1 Vehicle Library Extensions	2
1.3.2	3.2 News/Blog Extensions	3
1.3.3	3.3 Article Elements	3
1.3.4	3.4 SEO Features	3
1.3.5	3.5 User Generated Content (UGC)	4
1.3.6	3.6 Lead Generation	4
1.3.7	3.7 Article Recommendations	4
1.4	4. Non-Functional Requirements	5
1.4.1	4.1 Performance	5
1.4.2	4.2 Security	5
1.4.3	4.3 Scalability	5
1.4.4	4.4 Accessibility	5
1.5	5. Development and Deployment	5
1.5.1	5.1 Development Environment	5
1.5.2	5.2 Deployment Process	5
1.6	6. Maintenance and Support	6
1.6.1	6.1 System Maintenance	6
1.6.2	6.2 Technical Support	6
1.7	7. Project Timeline and Milestones	6
1.7.1	7.1 Development Phases	6
1.7.2	7.2 Deliverables	6
1.8	8. Conclusion	6

# 1 TopGear Philippines Clone - Software Requirements Specification

## 1.1 1. Introduction

### 1.1.1 1.1 Purpose

This document provides a comprehensive description of the requirements for the TopGear Philippines clone website. It details the system's functionality, technical specifications, and implementation requirements.

### 1.1.2 1.2 Project Scope

Development of a full-featured automotive news and review website that replicates the core functionality of TopGear Philippines, with additional features for user engagement and lead generation.

### 1.1.3 1.3 Technology Stack

- Frontend: React.js
- Backend: NestJS
- Database: MongoDB
- CMS: Payload CMS
- Deployment: Render.com

## 1.2 2. System Architecture

### 1.2.1 2.1 Frontend Architecture

- React.js-based SPA (Single Page Application)
- Component-based architecture
- Responsive design for mobile and desktop
- State management for user sessions and data
- SEO optimization with meta tags

### 1.2.2 2.2 Backend Architecture

- NestJS RESTful API
- MongoDB database integration
- Authentication and authorization system
- File upload and media management
- API rate limiting and security measures

### 1.2.3 2.3 CMS Architecture

- Payload CMS for content management
- Custom collections and fields
- Role-based access control
- Media library management
- Content versioning and drafts

## 1.3 3. Feature Requirements

### 1.3.1 3.1 Vehicle Library Extensions

#### 1.3.1.1 3.1.1 Vehicle Entries

- Comprehensive vehicle specifications
- Multiple image support

- Technical specifications
- Price information
- Performance data
- Feature lists

#### **1.3.1.2 3.1.2 Vehicle Categories**

- Classification by type
- Brand/manufacturer grouping
- Price range categorization
- Body type filtering

### **1.3.2 3.2 News/Blog Extensions**

#### **1.3.2.1 3.2.1 Article Management**

- Rich text editor support
- Multiple authors
- Publishing workflow
- Content categorization
- Image integration

#### **1.3.2.2 3.2.2 Content Categories**

- News
- Reviews
- Features
- Guides
- Events

### **1.3.3 3.3 Article Elements**

#### **1.3.3.1 3.3.1 Core Components**

- Headlines
- Subheadings
- Body text
- Image galleries
- Video embedding
- Social media integration

#### **1.3.3.2 3.3.2 Interactive Elements**

- Image sliders
- Specification tables
- Price comparison tools
- Rating systems

### **1.3.4 3.4 SEO Features**

#### **1.3.4.1 3.4.1 On-Page SEO**

- Meta title optimization
- Meta description management
- URL structure optimization
- Image alt text
- Schema markup

#### **1.3.4.2 3.4.2 Technical SEO**

- XML sitemap generation
- Robots.txt configuration
- Canonical URL management
- Mobile optimization
- Page speed optimization

#### **1.3.5 3.5 User Generated Content (UGC)**

##### **1.3.5.1 3.5.1 Comments System**

- User authentication
- Comment moderation
- Nested replies
- Like/dislike functionality
- Report inappropriate content

##### **1.3.5.2 3.5.2 User Reviews**

- Rating system
- Review moderation
- User profile integration
- Review helpfulness voting

#### **1.3.6 3.6 Lead Generation**

##### **1.3.6.1 3.6.1 Contact Forms**

- Vehicle inquiry forms
- Test drive requests
- Price quote requests
- Newsletter subscription
- Dealer locator

##### **1.3.6.2 3.6.2 Lead Management**

- Lead tracking
- Status management
- Source attribution
- Follow-up automation
- Analytics and reporting

#### **1.3.7 3.7 Article Recommendations**

##### **1.3.7.1 3.7.1 Recommendation Engine**

- Category-based suggestions
- Tag-based matching
- Vehicle relationship mapping
- Popularity metrics
- User behavior analysis

##### **1.3.7.2 3.7.2 Display Integration**

- Sidebar recommendations
- End-of-article suggestions
- Related content widgets

- Popular articles section

## **1.4 4. Non-Functional Requirements**

### **1.4.1 4.1 Performance**

- Page load time < 3 seconds
- API response time < 500ms
- Support for 1000+ concurrent users
- 99.9% uptime SLA

### **1.4.2 4.2 Security**

- HTTPS encryption
- JWT authentication
- CSRF protection
- Rate limiting
- Data encryption at rest

### **1.4.3 4.3 Scalability**

- Horizontal scaling capability
- Load balancing support
- Caching implementation
- Database indexing
- CDN integration

### **1.4.4 4.4 Accessibility**

- WCAG 2.1 compliance
- Screen reader compatibility
- Keyboard navigation
- Color contrast requirements
- Alt text for images

## **1.5 5. Development and Deployment**

### **1.5.1 5.1 Development Environment**

- Git version control
- Development, staging, and production environments
- Automated testing setup
- Code review process
- Documentation requirements

### **1.5.2 5.2 Deployment Process**

- Render.com deployment configuration
- CI/CD pipeline setup
- Database migration strategy
- Backup and recovery procedures
- Monitoring and logging

## **1.6 6. Maintenance and Support**

### **1.6.1 6.1 System Maintenance**

- Regular security updates
- Performance optimization
- Bug fixes and patches
- Feature updates
- Database maintenance

### **1.6.2 6.2 Technical Support**

- Issue tracking system
- Support documentation
- User training materials
- System status monitoring
- Incident response plan

## **1.7 7. Project Timeline and Milestones**

### **1.7.1 7.1 Development Phases**

1. Initial Setup and Configuration
2. Core Features Implementation
3. Content Management System Integration
4. User Interface Development
5. Testing and Quality Assurance
6. Deployment and Launch
7. Post-Launch Support

### **1.7.2 7.2 Deliverables**

- Source code repository
- Technical documentation
- User manuals
- API documentation
- Deployment guides
- Training materials

## **1.8 8. Conclusion**

This requirements specification outlines the comprehensive feature set and technical requirements for the TopGear Philippines clone website. The document serves as a guide for development, implementation, and maintenance of the system.