

# API Documentation

Version: 0.1.0  
Generated: 7/22/2025

## API Overview

This document provides detailed specifications for the RESTful API endpoints used by the Blog Web Application. The API is hosted at <https://api.blogapp.com/v1> and uses JWT authentication with rate limiting of 100 requests per minute per user.

## API Standards

All API responses follow a standard format with appropriate HTTP status codes:

```
{
  "code": 200, // HTTP status code
  "success": true, // boolean indicating success
  "message": "Operation successful", // human-readable message
  "data": {}, // response payload
  "timestamp": "2023-07-15T12:34:56Z" // ISO timestamp
}
```

## Authentication Endpoints

### Login

Endpoint: /auth/login

Method: POST

Description: Authenticates a user and returns a JWT token with 24-hour validity

Rate Limit: 10 attempts per minute per IP

Request Body:

```
{
  "username": "string", // Email or username
  "password": "string", // Min 8 characters
  "rememberMe": boolean, // Optional, default: false
  "deviceInfo": { // Device information for security
    "deviceId": "string",
    "platform": "string",
    "browserName": "string"
  }
}
```

Response:

```
{
  "code": 200,
  "success": true,
  "message": "Login successful",
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9... ",
    "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9... ",
    "expiresIn": 86400,
    "userId": "u123456789",
    "userRole": "user",
    "firstName": "John",
    "lastName": "Doe",
    "profileImage": "https://cdn.blogapp.com/profiles/john-doe.jpg"
  }
}
```

### Sign Up

Endpoint: /auth/signup

Method: POST

Description: Registers a new user with email verification

Rate Limit: 5 attempts per hour per IP

Request Body:

```
{
  "username": "string", // 3-20 alphanumeric characters
  "email": "string", // Valid email format
  "password": "string", // Min 8 chars, 1 uppercase, 1 number, 1 special
  "firstName": "string",
  "lastName": "string",
  "termsAccepted": boolean, // Must be true
  "marketingConsent": boolean // Optional
}
```

Response:

```
{
  "code": 201,
  "success": true,
  "message": "User registered successfully. Please verify your email.",
  "data": {
    "userId": "ul23456789",
    "verificationSent": true,
    "verificationExpiry": "2023-07-15T14:34:56Z"
  }
}
```

## Refresh Token

Endpoint: /auth/refresh

Method: POST

Description: Obtains a new access token using a refresh token

Request Body:

```
{
  "refreshToken": "string" // Valid refresh token
}
```

Response:

```
{
  "code": 200,
  "success": true,
  "message": "Token refreshed successfully",
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "expiresIn": 86400
  }
}
```

## Blog Post Endpoints

### Get All Posts

Endpoint: /posts

Method: GET

Description: Retrieves paginated blog posts with optional filtering

Authentication: Optional - Authenticated users receive personalized results

Query Parameters:

- page: number (default: 1) - Current page number
- limit: number (default: 10, max: 50) - Items per page
- sort: string (default: "createdAt") - Sort field (createdAt, title, likes)
- order: string (default: "desc") - Sort order (asc, desc)
- category: string - Filter by category
- tag: string - Filter by tag
- author: string - Filter by author ID
- search: string - Search in title and content

Response:

```
{
  "code": 200,
  "success": true,
  "message": "Posts retrieved successfully",

```

```

"data": {
  "posts": [
    {
      "id": "p123456789",
      "title": "Getting Started with React 18",
      "excerpt": "Learn the basics of React 18 and its new features",
      "author": {
        "id": "u987654321",
        "username": "reactmaster",
        "profileImage": "https://cdn.blogapp.com/profiles/reactmaster.jpg"
      },
      "coverImage": "https://cdn.blogapp.com/posts/react18-cover.jpg",
      "createdAt": "2023-07-10T08:15:30Z",
      "updatedAt": "2023-07-10T10:22:15Z",
      "category": "Programming",
      "tags": ["React", "JavaScript", "Frontend"],
      "readTime": 8,
      "likes": 124,
      "comments": 18,
      "isLiked": true, // Only present for authenticated users
      "isBookmarked": false // Only present for authenticated users
    },
    // More posts...
  ],
  "pagination": {
    "total": 142,
    "pages": 15,
    "currentPage": 1,
    "hasNext": true,
    "hasPrev": false
  }
}
}

```

## Create Blog Post

Endpoint: /posts

Method: POST

Description: Creates a new blog post

Authentication: Required (User or Admin role)

Request Body:

```

{
  "title": "string", // 10-150 characters
  "content": "string", // HTML content, min 100 characters
  "excerpt": "string", // Optional, max 300 characters
  "coverImage": "string", // Optional, URL or base64
  "category": "string", // Must be one of predefined categories
  "tags": ["string"], // Optional, max 5 tags
  "status": "string", // "draft" or "published"
  "seoTitle": "string", // Optional, max 70 characters
  "seoDescription": "string", // Optional, max 160 characters
  "allowComments": boolean // Optional, default: true
}

```

Response:

```

{
  "code": 201,
  "success": true,
  "message": "Blog post created successfully",
  "data": {
    "id": "p123456789",
    "title": "Getting Started with React 18",
    "slug": "getting-started-with-react-18",
    "url": "https://blogapp.com/posts/getting-started-with-react-18",
    "status": "published",
    "createdAt": "2023-07-15T12:34:56Z"
  }
}

```

## Comment Endpoints

## **Get Comments for Post**

Endpoint: /posts/{postId}/comments

Method: GET

Description: Retrieves comments for a specific blog post with pagination

Path Parameters: postId - ID of the blog post

Query Parameters: page, limit, sort (createdAt, likes), order (asc, desc)

Response Structure: Paginated list of comments with nested replies