Frontend Developer API Guide

Chinese Language Learning Platform API

This guide provides comprehensive documentation for frontend developers working with the Chinese Language Learning Platform API. It includes user flows, admin workflows, authentication patterns, and practical usage examples.

🔐 Authentication Overview

Base URL

```
Development: <http://localhost:3000/api>
Production: TBD
```

Authentication Flow

1. User Registration

```
POST /auth/signup
Content-Type: application/json

{
  "email": "student@example.com",
  "password": "SecurePassword123!",
  "displayName": "John Doe"
}
```

Response:

```
{
  "id": 1,
  "email": "student@example.com",
  "displayName": "John Doe",
```

```
  "role": "user",
  "isActive": true,
  "createdAt": "2024-01-15T10:30:00.000Z"
}
```

2. User Login

```
POST /auth/login
Content-Type: application/json

{
  "email": "student@example.com",
  "password": "SecurePassword123!"
}
```

Response:

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
}
```

3. Authentication Status Check

```
GET /auth/status
Authorization: Bearer YOUR_JWT_TOKEN
```

JWT Token Usage

All protected endpoints require the JWT token in the Authorization header:

```
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

👤 User Role System

Roles

- **User**: Students learning Chinese (read-only access to learning content)
- **Admin**: System administrators (full CRUD access to all content)

Permissions Overview

Endpoint Category	User	Admin
Profile Management	✅ (Own profile)	✅ (All profiles)
Course Viewing	✅	✅
Course Management	❌	✅
Lesson Content	✅	✅
Lesson Management	❌	✅
Word Dictionary	✅	✅
Word Management	❌	✅
Grammar Patterns	✅	✅
Grammar Management	❌	✅

🎓 User Learning Journey Flow

Phase 1: User Onboarding

Step 1: Get User Profile

```
GET /users/profile
Authorization: Bearer YOUR_JWT_TOKEN
```

Response:

```
{
  "id": 1,
  "email": "student@example.com",
  "displayName": "John Doe",
  "role": "user",
```

```
  "currentHskLevel": 1,
  "nativeLanguage": "en",
  "totalStudyDays": 0,
  "currentStreak": 0,
  "longestStreak": 0,
  "lastStudyDate": null,
  "isActive": true,
  "createdAt": "2024-01-15T10:30:00.000Z"
}
```

Step 2: Update User Profile (HSK Level, Native Language)

```
PUT /users/profile
Authorization: Bearer YOUR_JWT_TOKEN
Content-Type: application/json

{
  "currentHskLevel": 1,
  "nativeLanguage": "en"
}
```

Phase 2: Course Discovery

Step 1: Get Available Courses by HSK Level

```
GET /courses/hsk/1
Authorization: Bearer YOUR_JWT_TOKEN
```

Response:

```
[
  {
    "id": 1,
    "hskLevel": 1,
    "title": "HSK 1 - Basic Chinese Characters",
```

```
  "description": "Introduction to basic Chinese characters and pronunciation",
  "totalLessons": 10,
  "prerequisiteCourseId": null,
  "isActive": true,
  "orderIndex": 1,
  "createdAt": "2024-01-15T09:00:00.000Z"
    }
  ]
```

Step 2: Get Course Details

```
GET /courses/1
Authorization: Bearer YOUR_JWT_TOKEN
```

Phase 3: Learning Content Access

Step 1: Get Lessons for a Course

```
GET /lessons/course/1
Authorization: Bearer YOUR_JWT_TOKEN
```

Response:

```
[
  {
    "id": 1,
    "title": "Introduction to Numbers",
    "description": "Learn numbers 1-10 in Chinese",
    "courseId": 1,
    "orderIndex": 1,
    "isActive": true,
    "createdAt": "2024-01-15T09:30:00.000Z"
  }
]
```

Step 2: Get Complete Lesson with Content and Questions

```
GET /lessons/content/1
Authorization: Bearer YOUR_JWT_TOKEN
```

Response:

```
{
  "lesson": {
    "id": 1,
    "title": "Introduction to Numbers",
    "description": "Learn numbers 1-10 in Chinese",
    "courseId": 1,
    "orderIndex": 1
  },
  "content": [
    {
      "id": 1,
      "type": "text",
      "data": {
        "title": "Numbers 1-5",
        "content": "Let's start with the first five numbers...",
        "mediaUrl": null
      }
    }
  ],
  "questions": [
    {
      "id": 1,
      "type": "multiple-choice",
      "data": {
        "question": "How do you say '3' in Chinese?",
        "options": ["一", "二", "三", "四"],
        "correctAnswer": 2,
        "explanation": "三 (sān) means three in Chinese"
      }
```

```
    }
  ]
}
```

Step 3: Get Lesson Vocabulary

```
GET /lessons/1/words
Authorization: Bearer YOUR_JWT_TOKEN
```

Step 4: Get Lesson Grammar Patterns

```
GET /lessons/1/grammar-patterns
Authorization: Bearer YOUR_JWT_TOKEN
```

Phase 4: Dictionary and Reference

Word Lookup by Simplified Character

```
GET /words/simplified/你好
Authorization: Bearer YOUR_JWT_TOKEN
```

Grammar Pattern Search

```
GET /grammar-patterns/pattern/是...的
Authorization: Bearer YOUR_JWT_TOKEN
```

🛠️ Admin Management Workflows

Content Management Flow

Step 1: Get System Statistics

```
# User Statistics
GET /users/stats
Authorization: Bearer ADMIN_JWT_TOKEN

# Course Statistics
GET /courses/stats
Authorization: Bearer ADMIN_JWT_TOKEN
```

Step 2: Manage Courses

Create Course:

```
POST /courses
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "hskLevel": 1,
  "title": "HSK 1 - Basic Greetings",
  "description": "Learn essential Chinese greetings",
  "prerequisiteCourseId": null,
  "orderIndex": 1
}
```

Update Course:

```
PUT /courses/1
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "title": "HSK 1 - Basic Greetings (Updated)",
  "description": "Updated description",
```

```
  "isActive": true
}
```

Soft Delete Course:

```
DELETE /courses/1
Authorization: Bearer ADMIN_JWT_TOKEN
```

Restore Course:

```
PUT /courses/1/restore
Authorization: Bearer ADMIN_JWT_TOKEN
```

Step 3: Manage Lessons

Create Lesson:

```
POST /lessons
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "title": "Greetings Vocabulary",
  "description": "Essential greeting words and phrases",
  "courseId": 1,
  "orderIndex": 1
}
```

Create Lesson Content and Questions (Unified Endpoint):

```
POST /lessons/items
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "lessonId": 1,
```

```
  "type": "content", // or "question"
  "data": {
    "title": "Basic Greetings",
    "content": "Chinese greetings are essential...",
    "mediaUrl": "<https://example.com/audio.mp3>"
  }
}
```

Step 4: Assign Learning Materials

Add Words to Lesson:

```
POST /lessons/1/words
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

[
  {
    "wordSenseId": 1,
    "orderIndex": 1
  },
  {
    "wordSenseId": 2,
    "orderIndex": 2
  }
]
```

Add Grammar Patterns to Lesson:

```
POST /lessons/1/grammar-patterns
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

[
  {
    "grammarPatternId": 1,
```

```
    "orderIndex": 1
  }
]
```

Step 5: Content Management

Create Word:

```
POST /words
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "simplified": "你好",
  "traditional": "你好",
  "pinyin": "nǐ hǎo"
}
```

Create Grammar Pattern:

```
POST /grammar-patterns
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "pattern": "是...的",
  "description": "Emphasis pattern for completed actions",
  "difficulty": 2
}
```

Step 6: User Management

Get All Users:

```
GET /users?page=1&limit=10&role=user&isActive=true
```

```
Authorization: Bearer ADMIN_JWT_TOKEN
```

Update Any User:

```
PUT /users/1
Authorization: Bearer ADMIN_JWT_TOKEN
Content-Type: application/json

{
  "displayName": "Updated Name",
  "currentHskLevel": 2,
  "role": "user",
  "isActive": true
}
```

📚 API Endpoint Reference

Authentication Endpoints

Method	Endpoint	Description	Role Required
POST	/auth/signup	Register new user	None
POST	/auth/login	Login user	None
GET	/auth/status	Check auth status	User/Admin

User Management Endpoints

Method	Endpoint	Description	Role Required
GET	/users/profile	Get current user profile	User/Admin
PUT	/users/profile	Update current user profile	User/Admin
GET	/users/stats	Get user statistics	Admin
GET	/users	Get all users with pagination	Admin
GET	/users/:id	Get user by ID	Admin
PUT	/users/:id	Update any user	Admin

Method	Endpoint	Description	Role Required
DELETE	/users/:id	Soft delete user	Admin
DELETE	/users/:id/hard	Hard delete user	Admin
PUT	/users/:id/restore	Restore soft-deleted user	Admin

Course Management Endpoints

Method	Endpoint	Description	Role Required
POST	/courses	Create new course	Admin
GET	/courses/stats	Get course statistics	Admin
GET	/courses/hsk/:level	Get courses by HSK level	Admin
GET	/courses	Get all courses with pagination	Admin
GET	/courses/:id	Get course by ID	Admin
PUT	/courses/:id	Update course	Admin
DELETE	/courses/:id	Soft delete course	Admin
DELETE	/courses/:id/hard	Hard delete course	Admin
PUT	/courses/:id/restore	Restore course	Admin

Lesson Management Endpoints

Method	Endpoint	Description	Role Required
POST	/lessons	Create new lesson	Admin
GET	/lessons	Get all lessons with pagination	User/Admin
GET	/lessons/content/:id	Get complete lesson data	User/Admin
GET	/lessons/:id	Get lesson by ID	User/Admin
GET	/lessons/course/:courseId	Get lessons by course (active only)	User/Admin
GET	/lessons/course/:courseId/all	Get all lessons by course	Admin
PUT	/lessons/:id	Update lesson	Admin
DELETE	/lessons/:id/soft	Soft delete lesson	Admin
DELETE	/lessons/:id/hard	Hard delete lesson	Admin

Method	Endpoint	Description	Role Required
PATCH	/lessons/:id/restore	Restore lesson	Admin

Lesson Content Management

Method	Endpoint	Description	Role Required
GET	/lessons/:id/words	Get lesson vocabulary	User/Admin
POST	/lessons/:id/words	Add words to lesson	Admin
DELETE	/lessons/:id/words	Remove words from lesson	Admin
GET	/lessons/:id/grammar-patterns	Get lesson grammar patterns	User/Admin
POST	/lessons/:id/grammar-patterns	Add grammar patterns to lesson	Admin
DELETE	/lessons/:id/grammar-patterns	Remove grammar patterns from lesson	Admin
POST	/lessons/items	Create lesson content/questions	Admin

Word Dictionary Endpoints

Method	Endpoint	Description	Role Required
POST	/words	Create new word	Admin
GET	/words/stats	Get word statistics	Admin
GET	/words/simplified/:simplified	Get word by simplified form	User/Admin
GET	/words	Get all words with pagination	User/Admin
GET	/words/:id	Get word by ID	User/Admin
PUT	/words/:id	Update word	Admin
DELETE	/words/:id	Delete word	Admin

Grammar Pattern Endpoints

Method	Endpoint	Description	Role Required
POST	/grammar-patterns	Create new pattern	Admin

Method	Endpoint	Description	Role Required
GET	/grammar-patterns	Get all patterns	User/Admin
GET	/grammar-patterns/stats	Get pattern statistics	Admin
GET	/grammar-patterns/pattern/:pattern	Get pattern by pattern string	User/Admin
GET	/grammar-patterns/:id	Get pattern by ID	User/Admin
PATCH	/grammar-patterns/:id	Update pattern	Admin
DELETE	/grammar-patterns/:id	Delete pattern	Admin