

API Documentation

Overview

The Berean Bible Reading Plan Application integrates with external Bible APIs and provides internal API endpoints for application functionality.

External API Integration

ESV API (Crossway)

Base URL: `https://api.esv.org/v3/`

Authentication

- **Method:** Token-based authentication
- **Header:** `Authorization: Token {ESV_API_KEY}`
- **Environment Variable:** `ESV_API_KEY`

Endpoints Used

Get Passage Text

```
GET /passage/text/
```

Parameters:

- `q` (string): Bible reference (e.g., "John 3:16", "Genesis 1:1-31")
- `include-headings` (boolean): Include section headings
- `include-footnotes` (boolean): Include footnotes
- `include-verse-numbers` (boolean): Include verse numbers
- `include-short-copyright` (boolean): Include copyright notice

Example Request:

```
const response = await fetch('https://api.esv.org/v3/passage/text/?q=John+3:16', {
  headers: {
    'Authorization': `Token ${process.env.ESV_API_KEY}`
  }
});
```

Example Response:

```
{
  "query": "John 3:16",
  "canonical": "John 3:16",
  "parsed": [[43003016, 43003016]],
  "passage_meta": [
    {
      "canonical": "John 3:16",
      "chapter_start": [43003001, 43003036],
      "chapter_end": [43003001, 43003036],
      "prev_verse": 43003015,
      "next_verse": 43003017,
      "prev_chapter": [43002001, 43002025],
      "next_chapter": [43004001, 43004054]
    }
  ],
  "passages": [
    "John 3:16\n\n [16] For God so loved the world, that he gave his only Son, that whoever believes in him should not perish but have eternal life. (ESV)\n\n"
  ]
}
```

Internal API Endpoints

Reading Progress

Get User Progress

GET /api/progress

Response:

```
{
  "currentDay": 1,
  "completedDays": [1, 2, 3],
  "totalDays": 365,
  "lastReadDate": "2025-07-10",
  "streak": 5
}
```

Update Progress

POST /api/progress

Request Body:

```
{
  "day": 1,
  "completed": true,
  "readingTime": 1200
}
```

Reading Plans

Get Available Plans

```
GET /api/plans
```

Response:

```
{
  "plans": [
    {
      "id": "chronological",
      "name": "Chronological Bible Reading",
      "description": "Read the Bible in chronological order",
      "duration": 365,
      "dailyReadings": [...]
    }
  ]
}
```

Get Plan Details

```
GET /api/plans/{planId}
```

Response:

```
{
  "id": "chronological",
  "name": "Chronological Bible Reading",
  "description": "Read the Bible in chronological order",
  "duration": 365,
  "dailyReadings": [
    {
      "day": 1,
      "readings": [
        {
          "book": "Genesis",
          "chapters": [1, 2, 3]
        }
      ]
    }
  ]
}
```

Bible Translations

Get Available Translations

```
GET /api/translations
```

Response:

```
{
  "translations": [
    {
      "id": "esv",
      "name": "English Standard Version",
      "abbreviation": "ESV",
      "language": "English",
      "available": true
    },
    {
      "id": "niv",
      "name": "New International Version",
      "abbreviation": "NIV",
      "language": "English",
      "available": true
    }
  ]
}
```

Error Handling

Standard Error Response

```
{
  "error": {
    "code": "INVALID_REFERENCE",
    "message": "The provided Bible reference is invalid",
    "details": "Reference 'John 99:99' does not exist"
  }
}
```

Common Error Codes

- `INVALID_REFERENCE` : Bible reference is malformed or doesn't exist
- `API_LIMIT_EXCEEDED` : External API rate limit exceeded
- `UNAUTHORIZED` : Invalid or missing API key
- `TRANSLATION_UNAVAILABLE` : Requested translation is not available
- `INTERNAL_ERROR` : Server-side error occurred

Rate Limiting

ESV API Limits

- **Free Tier**: 5,000 requests per day
- **Paid Tier**: Higher limits available

Application Limits

- **Reading Progress**: 100 requests per hour per user
- **Bible Text**: 1,000 requests per hour per user

Caching Strategy

Client-Side Caching

- Bible passages cached in localStorage for 24 hours
- Translation metadata cached for 7 days
- User progress synced on app load and periodically

Server-Side Caching

- Popular passages cached for 1 hour
- Translation lists cached for 24 hours
- Reading plans cached indefinitely (updated on deployment)

Authentication & Security

API Key Management

- ESV API keys stored as environment variables
- Never exposed to client-side code
- Rotated regularly for security

Data Privacy

- User progress stored locally (no server-side user data)
- No personal information collected
- GDPR compliant data handling

Development & Testing

Mock API for Development

```
// For development without API key
const mockResponse = {
  passages: ["Mock Bible text for development"],
  query: "John 3:16"
};
```

Testing Endpoints

- Use test API keys for development
- Mock responses for unit tests
- Integration tests with rate limiting considerations

For implementation examples and code samples, see the `/app/lib/api/` directory in the project.