# Feathers

A minimalist real-time framework for tomorrow's apps.

The official guide

# Table of Contents


- Introduction 1.1
- Guides 1.2
  - About Feathers 1.2.1
    - Features 1.2.1.1
    - Philosophy 1.2.1.2
    - Feathers vs. X 1.2.1.3
  - The Basics 1.2.2
    - Introduction 1.2.2.1
    - What not to worry about 1.2.2.2
    - Installing the Examples 1.2.2.3
    - Basic Feathers 1.2.2.4
    - A database connector 1.2.2.5
    - A REST API server 1.2.2.6
    - A Feathers REST client 1.2.2.7
    - A Feathers WebSocket client 1.2.2.8
    - The "a-ha!" moment 1.2.2.9
    - Hooks middleware 1.2.2.10
    - All about hooks 1.2.2.11
    - Hooks, part 1 1.2.2.12
    - Hooks, part 2 1.2.2.13
    - Writing your own hooks 1.2.2.14
    - Testing hooks 1.2.2.15
    - Real-time 1.2.2.16
  - The Generator (CLI) 1.2.3
    - Generate the application 1.2.3.1
    - Add authentication 1.2.3.2
    - Add the teams service 1.2.3.3
    - Add the populate hook 1.2.3.4
    - Run the application 1.2.3.5
  - A Chat Application 1.2.4
    - Creating the application 1.2.4.1
    - Generating a service 1.2.4.2
    - Building a frontend 1.2.4.3
    - Adding authentication 1.2.4.4
    - Processing data 1.2.4.5
  - Frameworks 1.2.5
    - React and React Native 1.2.5.1

- VueJS 1.2.5.2
- Angular 1.2.5.3
- Others 1.2.5.4

Authentication 1.2.6

- How JWT works 1.2.6.1
- What's New (external link) 1.2.6.2
- Migration Guide (external link) 1.2.6.3
- Recipe: Custom Login Response 1.2.6.4
- Recipe: Custom JWT Payload 1.2.6.5
- Recipe: Mixed Auth Endpoints 1.2.6.6
- Recipe: Basic OAuth 1.2.6.7

Offline first 1.2.7

- Strategies 1.2.7.1
- Snapshot 1.2.7.2
- Realtime 1.2.7.3
- Optimistic mutation 1.2.7.4
- Own-data, own-net 1.2.7.5
- Sync-data, sync-net 1.2.7.6
- Configure snapshot 1.2.7.7
- Configure realtime 1.2.7.8
- Configure publication 1.2.7.9
- Example snapshot 1.2.7.10
- Example realtime & publication 1.2.7.11
- Example optimistic mutation 1.2.7.12
- Tests as examples 1.2.7.13
- More examples 1.2.7.14

Advanced topics 1.2.8

- Debugging 1.2.8.1
- Configuration 1.2.8.2
- File uploads 1.2.8.3
- Creating a Feathers plugin 1.2.8.4
- Seeding services 1.2.8.5
- Using a view engine 1.2.8.6
- Scaling 1.2.8.7

API 1.3

Core 1.3.1

- Application 1.3.1.1
- Services 1.3.1.2
- Hooks 1.3.1.3
- Common Hooks 1.3.1.4

- Client 1.3.1.5
- Events 1.3.1.6
- Errors 1.3.1.7

Transports 1.3.2

- REST 1.3.2.1
- Express 1.3.2.2
- Socket.io 1.3.2.3
- Primus 1.3.2.4

Authentication 1.3.3

- Server 1.3.3.1
- Client 1.3.3.2
- Local 1.3.3.3
- Local management 1.3.3.4
- JWT 1.3.3.5
- OAuth1 1.3.3.6
- OAuth2 1.3.3.7
- Hooks 1.3.3.8

Databases 1.3.4

- Common API 1.3.4.1
- Querying 1.3.4.2
- Memory 1.3.4.3
- NeDb 1.3.4.4
- LocalStorage 1.3.4.5
- MongoDB 1.3.4.6
- Mongoose 1.3.4.7
- Sequelize 1.3.4.8
- Knex 1.3.4.9
- RethinkDB 1.3.4.10
- Elasticsearch 1.3.4.11

Security 1.4

Ecosystem 1.5

Help 1.6

FAQ 1.7

Contributing 1.8

License 1.9

> An open source REST and realtime API layer for modern applications.

With Feathers it's easy to create scalable, real-time applications. Make creating web and mobile apps fun with Feathers.

This documentation is also available as a PDF.

### Guides
Get familiar with Feathers by building your first apps. Learn topics from beginner to more advanced-levels.

### API
Learn more about Feathers' Universal API, and plugins for Authentication and handling data.

### Security
Understand our commitment to security and how it affects your applications.

### Ecosystem
See what amazing things the Feathers Core Team and Community have built.

### Help
Learn how to plug in to the active and helpful FeathersJS Community.

### FAQ
A collection of Frequently Asked Questions.

### Contributing
Learn how you can contribute to this documentation.

### License

Click here to download full PDF material