

SAN JUAN ISLANDS ORCA SIGHTINGS SERVICE

API documentation presentation

<https://juliebro.github.io/orca-sightings-api/>



PROJECT OVERVIEW

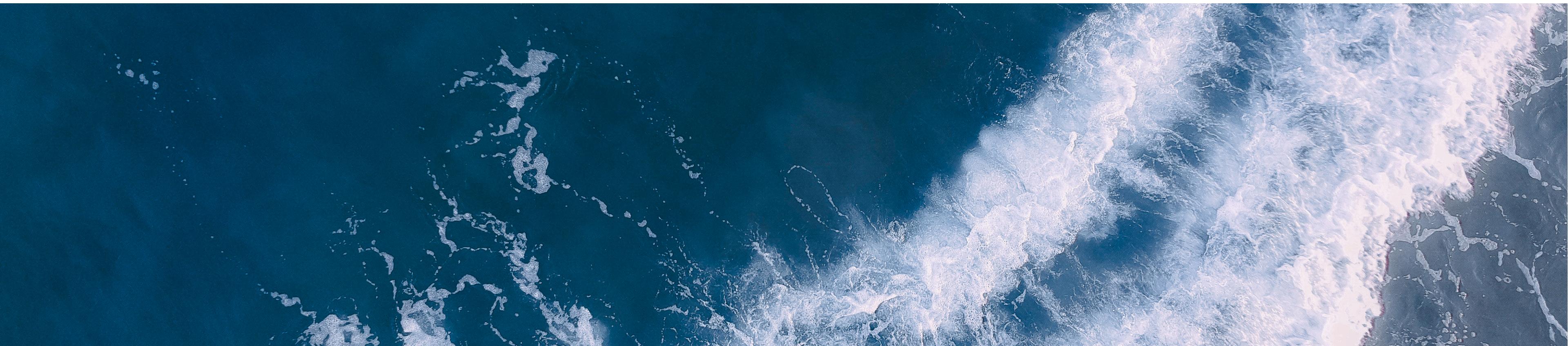
The San Juan Islands Orca Sightings service is a simulated cloud-based RESTful API designed to track orca movements around the San Juan Islands. It allows orca enthusiasts to report sightings and helps others to spot whales from land.

The documentation teaches developers how to easily set up, understand, and interact with the API. It also includes a comprehensive reference.



TEAM

Technical writer: Julie Brodeur
Subject matter expert: Jeff Naemura



KEY FEATURES

- Well-structured, searchable, and navigable documentation.
- Clear and concise explanations.
- Comprehensive API reference and helpful tutorials.
- Focus on user experience and ease of adoption.



USER ONBOARDING

Overview, getting started, and tutorial topics
Search function makes it easy to find the right info

The screenshot shows a user interface for the "Orca Sightings service". On the left, a sidebar menu lists several options: Overview (which is highlighted in purple), Tutorials, Set up your environment, Start the service, Get started by listing orca sightings, List users, Add a new user, and Update a user. To the right of the sidebar, there is a search bar labeled "Search Orca Sightings service" and a link labeled "Service repo". The main content area features a large heading "San Juan Islands Orca Sightings service". Below the heading, a paragraph explains the service's purpose: "This cloud-based service tracks the movements of orca pods around the San Juan Islands in the Salish Sea. Orcas are also known as killer whales." Another paragraph describes the service's functionality: "Using this service, orca enthusiasts can report the location and time of their sightings off the coast of any of the San Juan Islands, making it easy for others to sight these whales from land so they don't have to charter a boat. As a developer, you can include this service in your app." At the bottom, there is a call-to-action: "To get started, learn how to [set up your environment](#) and [list all sightings](#) in the service, or jump down to learn more about the [user community](#) and [features](#)".

API REFERENCE

- Provides details on API resources `/users` and `/sightings`
- Supported HTTP Methods: GET, POST, PUT, PATCH, DELETE clearly defined for each endpoint.

The screenshot shows a sidebar menu on the left and a main content area on the right. The sidebar has sections for Overview, Tutorials, API reference, Base URL, and a expanded section for Users resource. The expanded Users resource section includes links for GET users, POST users, PUT users, PATCH users, and DELETE users. The main content area has a title for the /users resource, a description of its functions, a link to the /sightings resource, an Endpoint section with a URL template, and a Properties section with a placeholder example.

Orca Sightings service

Overview ▾

Tutorials ▾

API reference ▾

Base URL

Users resource ▾

- GET users
- POST users
- PUT users
- PATCH users
- DELETE users

/users resource

The `/users` resource lets you add a new user, update all or part of an existing user, list all users, list a user by its ID, or delete an existing user by its ID. Users represent the whale watching enthusiasts who report orca sightings.

Also see [/sightings resource](#).

Endpoint

`{base_url}/users`

Also see [Base URL](#).

Properties

Shows an example of a full `/users` resource:

CHALLENGES

- Git merge issues
- Problems applying a theme
- Terminology ambiguity
- Challenges editing files at work
- Some API responses led me down rabbit holes



SUCCESSES

- Effective collaboration with SME
- Applied Just the Docs theme
- Content visualization and planning
- Consistent additions to repo
- AI-assisted content generation saved time
- Learned new Markdown tricks
- Improved docs with usability testing



VALUE

For developers:

- Faster integration—clear setup instructions reduce time to engagement
- Self-service support—comprehensive examples reduce support tickets
- Confidence building—progressive tutorials ensure success

For business:

- Reduced support costs—documentation answers common questions
- Improved adoption—better developer experience drives usage
- Professional credibility—quality docs enhance product perception

THANK YOU

<https://juliebro.github.io/orca-sightings-api/>

