Documentation via Docsify

Agenda

1. Learning Objectives
2. Activity: Custom API (25 minutes)
3. Documentation Driven Development (15 minutes)
4. BREAK (10 minutes)
5. In Class Activity: Docsify (30 Minutes)

Learning Objectives

1. Define what Document Driven Development means with regard to software engineering.
2. Practice using Document Driven Development in conjunction with Docsify in order to kick off the Custom API project.
3. Deploy a live, autogenerated documentation website on Github Pages.

Go over Homework (15 minutes)

Go over the **TDD/BDD homework** and answer any questions.

Activity: Custom API (25 minutes)

Brainstorm APIs (10 minutes)

Go to the **Fun API List** and browse the available APIs. Choose your top 3 favorites, and explore their documentation pages.

Share your top 3 favorites with a partner.

Fill out Worksheet (15 minutes)

Fill out the **Custom API Worksheet** to get started on your project idea!

Documentation Driven Development (15 minutes)

Philosophy

1. From a user's perspective: if a **feature is not documented**, then it **doesn't exist**.
2. If a feature is **documented incorrectly**, then it's **broken**.

Strategy

- **Document the feature first**. Figure out how you're going to describe the feature to users.
  - **Remember**: if it's not documented, it doesn't exist.
  - Documentation is the **best way to define a feature**; without it, how will the user know a feature exists?
- Whenever possible, **review documentation with users *before* ANY development** begins.
- Once documentation has been written, commence development. TDD is preferred.
- Unit tests should be written that test the features as described by the documentation.

  - If **functionality no longer aligns with the test, the test should fail**.
- When a feature is being modified, it should be **modified documentation-first**.
- When **docs are modified, tests are modified**. No exceptions.
- **Documentation and software** should both be **versioned**:
  - **Versions should match**: someone working with an old version of your software still needs to locate and use the proper

Order of Operations

1. Write docs
2. Get feedback
3. TDD (tests fully align with documentation)
4. Push features live
5. Test live functionality
6. Publish documentation
7. Increment versions
8. Deliver feature to users

BREAK (10 minutes)

In Class Activity: Docsify (30 Minutes)

Everyone Do: Code With Me

Using the **docsify Quick Start guide**, instructor will present a live demonstration of a fresh `docsify` integration, step-by-step their own working documentation website for a sample project.

Students Do: Custom API Project

Create a documentation site for the Custom Api you designed in the first half of class. This will be the basis for your **final proj**

Wrap-Up

Write documentation for your **Custom API Project** based on what we learned in class today. You'll turn in your worksheet, doc

Fill out our **Vibe Check form** with any feedback you have for the class.

Additional Resources

- **docsify**