

Module 3: Branching and Merging in Git

Case Study-1

edureka!

edureka!

© Brain4ce Education Solutions Pvt. Ltd.

Case Study

Problem Statement

Focus: Git repository & Commands

Business challenge/requirement

In previous modules, you got a fair understanding of git and basic git commands. In this module, you will work on remote repositories and will perform operations on them. Also, you will learn to create tags.

An International logistics company is looking towards making their deliveries more agile and high speed and to achieve this, they wanted to drive their application lifecycle management through their devops initiatives. With hundreds of developers working across remote locations in different timezones and their old traditional development methods needed to be rewritten, super agile git helped them achieve better market agility and win more customers. Git transformed how they used to code and that helped them manage their ALM better.

Logistics are also migrating to SOA or Micro services architectures. They ended up creating lots of APIs. For testing, they use Swagger / Postman etc but when it comes to meta data it's hard to manage. Swagger is one of the best ways to do that. In Swagger documentation, you can add meta data about resource, end points and individual fields. This makes things easy for other people as they can now get information about any API whichever required.

In below mentioned case study we will explore swagger parser. This is a pet store where you check different kind of pets and order them online.

Swagger example: <http://petstore.swagger.io>

Problem:

- Go through swagger and figure out how many micro services (repositories) are required.

This case study is broken into 3 phases so that you can solve one by one.

Stage 0

1. You have given an empty repository having only README file.
2. If Repository is not given create one having README file which will contains some information about repository.

Stage 1

1. Pull master branch.
2. Create program Which will take Swagger JSON as input (eg. <http://petstore.swagger.io>)
3. Output should be all end points present in swagger (eg. "[/pet](#)")
4. Create a new feature branch and push the changes

Stage 2

1. Create one more branch from GIT HUB UI
2. Check in command line if you can see new branch details
3. Use appropriate command to pull branch from remote.

Stage 3

1. Create tag release 1.1 & push to remote
2. Change in file and create new tag release 1.2 & push to remote
3. List all tags
4. Delete release1.1 from remote
5. Delete release 1.1 from local

edureka!