Introducing JBehave

Cătălin Tudose

PHD IN COMPUTER SCIENCE, JAVA AND WEB TECHNOLOGIES EXPERT
https://www.linkedin.com/in/catalin-tudose-847667a1

Overview

- Introduction to JBehave

- Functionality and terminology

- Install and configure JBehave

- Move the original non-BDD application to BDD using JBehave

- Add new features in BDD style using JBehave

- Comparison between Cucumber and JBehave

JBehave Functionality

Write stories

Map steps

Configure stories

Run stories

JBehave Terminology

Story

Scenarios

Steps

JBehave Installation and Configuration

Demo

- Flights management application

- Introduce JBehave dependencies in the Maven configuration

- Create first feature and generate the skeleton of the JBehave tests
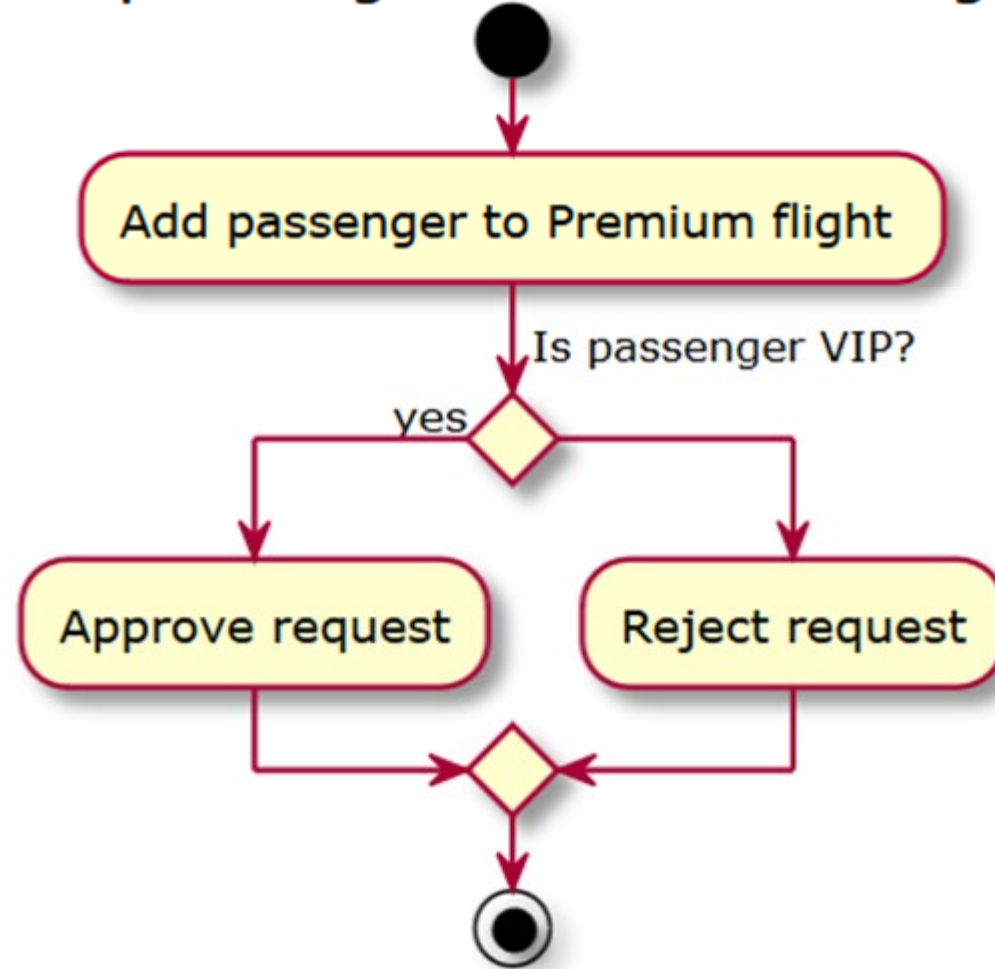
- Move all existing tests to JBehave
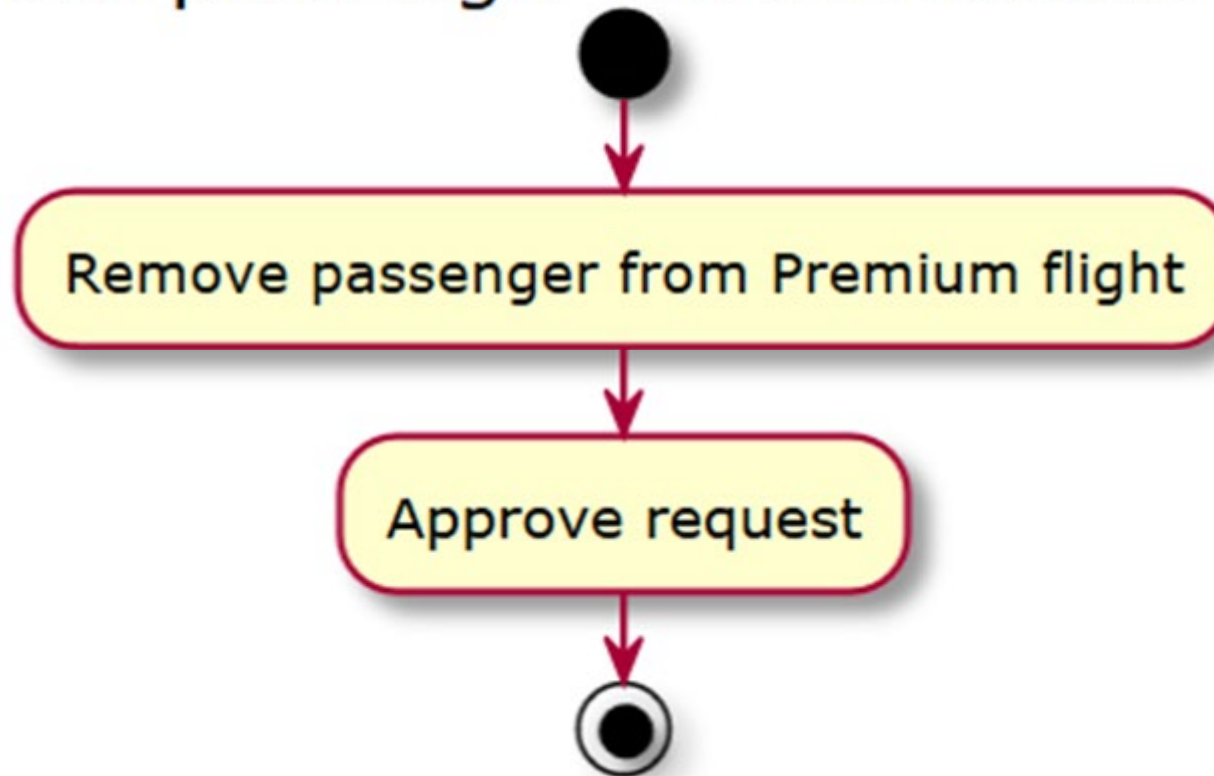
Adding a Premium Flight with JBehave

Adding Passenger New Business Logic

Removing Passenger New Business Logic

Demo

Define the scenarios of using the Premium Flight

Write the JBehave tests that describe the scenarios

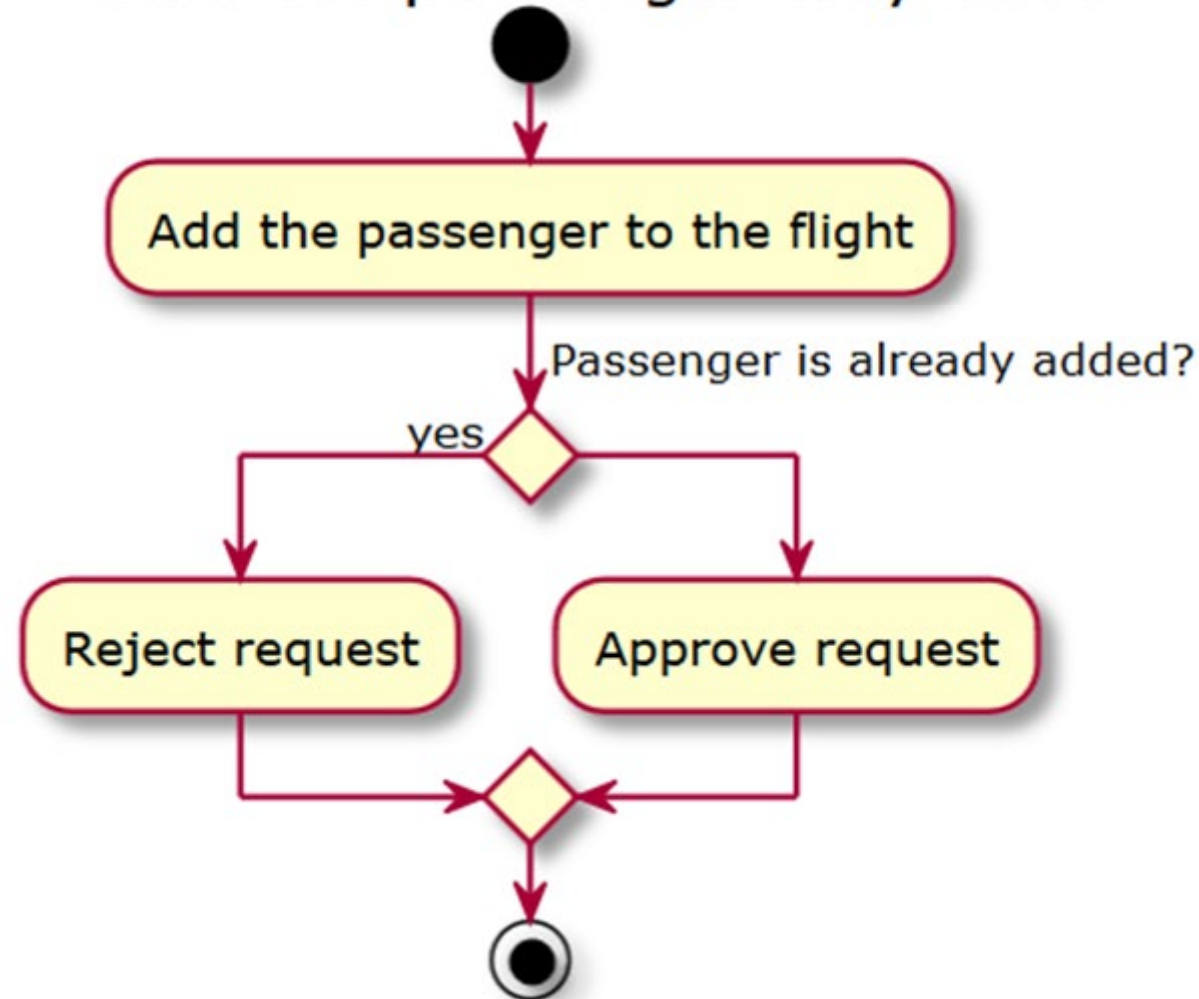Fix the tests by implementing the new story

Adding a Passenger Only Once

Adding Passenger Only Once

Demo

Define the scenarios of adding a passenger more than once

Generate the JBehave tests that describe the scenarios

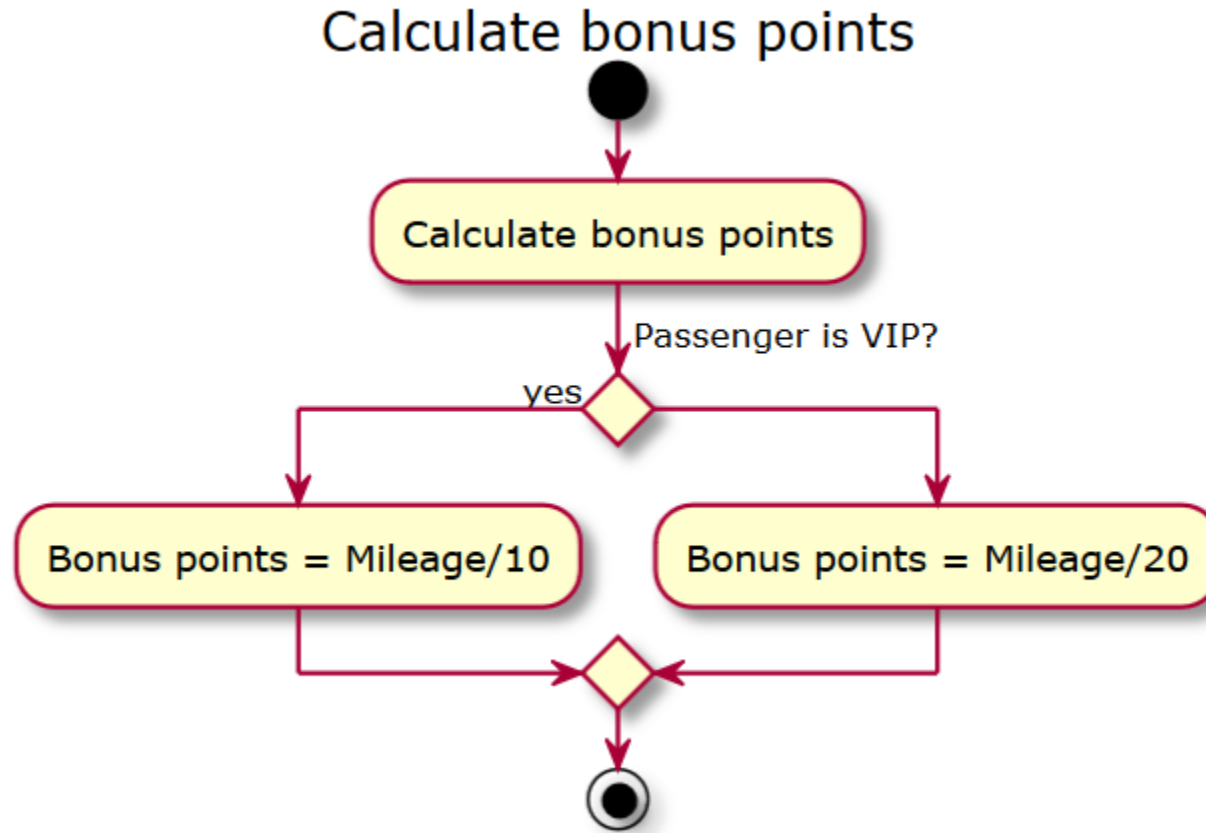Effectively add the code that implements the restriction

Run the tests - to be green

Adding a Policy for Bonus Points

Calculate Passenger's Bonus Points

Demo

Define the scenarios of awarding the bonus points

Generate the JBehave tests that describe the scenarios

Effectively add the code that implements the bonus award

Run the tests - to be green

Cucumber vs. JBehave

Cucumber vs. JBehave

BDD principles

Features vs. Stories

BDD keywords

IDE support

Summary

Introduced JBehave

Installed and configured JBehave

Moved the existing tests to JBehave

Introduced 3 new features BDD way, with the help of JBehave

Compared Cucumber and JBehave