SPELLTRACKER V2

NSAC15 PROJECT

WHAT IS THIS PROJECT?

A Software Development exercise

- Agile & Project Management

- Databases

- Java/Spring Boot API

- Front-End

- Automated Testing

WHAT IS SPELLTRACKER?

- Interactive, updatable database for use with D&D 5e

- Stores and manipulates basic information about game elements

- Can be used via a simple but effective front-end

- Niche project but with a lot of scope for expansion

HOW WAS IT DONE?

- Jira project management

- Risk Assessment

- Version control with GitHub

- API built in Java

- JUnit automated testing

- HTML front-end with JavaScript integration

- Bootstrap/CSS styling

- Compiled into a .jar file

SPRINT PLAN

- Broken down into Epics, Stories and Tasks
  - 6 Epics
  - 7 Stories
  - 38 Tasks
- Added User Stories and Acceptance Criteria
- MoSCoW priority
- Planning Point assignment

NEW TECHNOLOGIES LEARNED?

- Had some familiarity with most of the technologies used

- Front-end development was completely new
    - HMTL, CSS, JS and Bootstrap

- Enjoyable and rewarding experience
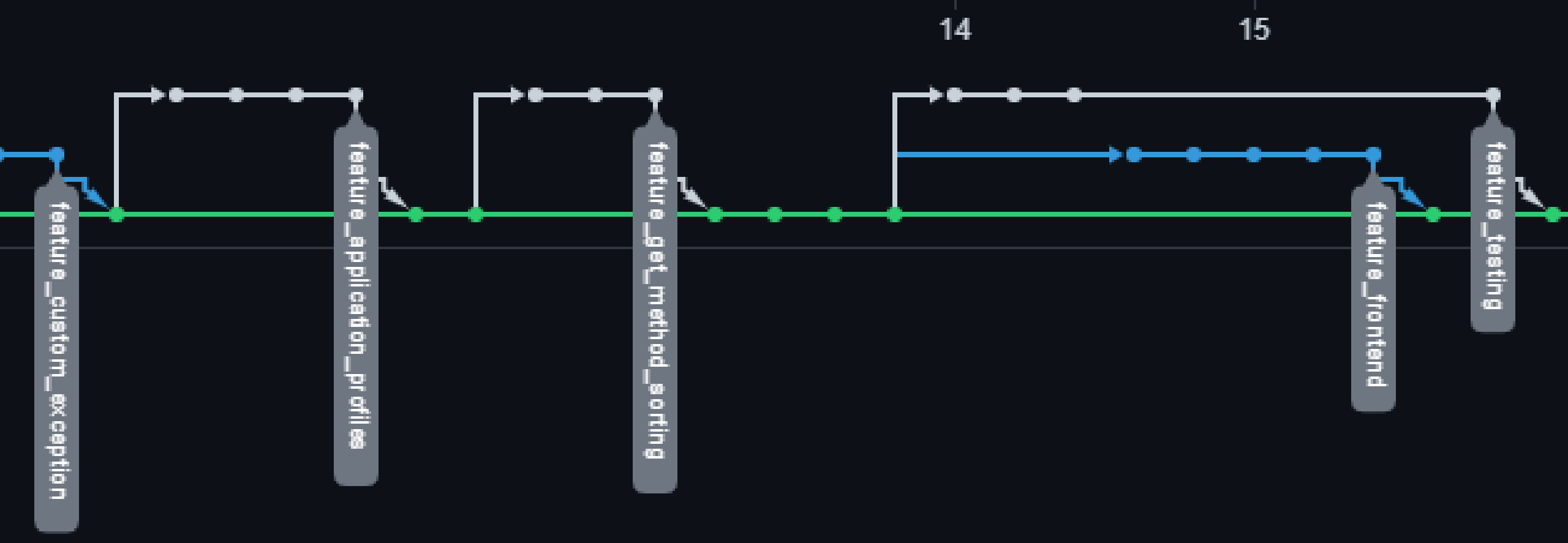
CONTINUOUS INTEGRATION

- Feature-branch model
- 34 commits to date, with 5 merges
- All elements of project kept within same repository

- ○ Main branch
- ○ Dev branch
- ○ 5 Feature branches

TESTING

- Had to adapt previous testing
  - New auto-incrementing PK led to different behaviour
- 51 Tests
- Achieved 95.7% test coverage of src/main/java

SpellTrackerV2 (15 Sept 2022 18:31:50)

Element	Coverage
∨ 📁 SpellTrackerV2	98.2 %
> 📁 src/test/java	98.8 %
∨ 📁 src/main/java	95.7 %
∨ ⊞ com.bae.service	95.5 %
> J SpellService.java	95.5 %
> ⊞ com.bae.exceptions	41.2 %
> ⊞ com.bae	37.5 %
∨ ⊞ com.bae.controller	100.0 %
> J SpellController.java	100.0 %
> ⊞ com.bae.entities	100.0 %

Finished after 15.942 seconds

Runs: 51/51   Errors: 0   Failures: 0

- SpellTrackerV2ApplicationTests [Runner: JUnit 5] (0.341 s)
- SpellControllerSystemIntegrationTest [Runner: JUnit 5] (1.029 s)
- SpellControllerWebIntegrationTest [Runner: JUnit 5] (0.281 s)
- SpellServiceIntegrationTest [Runner: JUnit 5] (0.378 s)
- SpellServiceUnitTest [Runner: JUnit 5] (0.195 s)
- SpellClassTest [Runner: JUnit 5] (0.028 s)

FRONT-END DEMO

SPRINT REVIEW

All Must-Have Tasks and Stories are on track to be completed
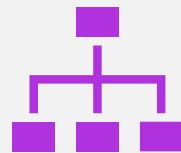
Of the Should-Have and Could-Have priorities, all are set to be completed

PROJECT RETROSPECTIVE

- MVP met and then exceeded

- All technologies worked with only minor issues

- Jira board and feature-branch model used well

- Room for expansion of ideas

  - Class attributes

  - Other entities

  - New methods

- Increased front-end functionality

THANK YOU

ANY QUESTIONS?