

A series of thin, light brown lines forming various overlapping polygons and triangles in the top-left corner of the page.

HOBBY WEB APPLICATION

By Charles Cairney



INTRODUCTION

Project lead: Charles Cairney - Software Specialist - QA Ltd

MVP:

- Full-Stack Web Application
- Full CRUD functionality Back/FrontEnd
- Local Database in MySQL
- Intuitive FrontEnd
- Testing Coverage of 80%
- Application must Compile

Start:

- Brainstormed hobbies that apply
- Determined which would display the most skills learned
- Create Initial Risk Assessment as this would apply to all
- Created Repo on Github
- Setup up JIRA
- Start planning my sprint



SPRINT PLAN - JIRA

Sprint 1	Sprint 2	Sprint 3	Sprint 4	Presentation Day
<ul style="list-style-type: none">• Planning• Workflow setup• Research	<ul style="list-style-type: none">• API Creation• FrontEnd implementation	<ul style="list-style-type: none">• Full Testing	<ul style="list-style-type: none">• Presentation• Documentation• Compile & Push	
Date: 09:00 15/08/22 To 18:00 15/08/22	Date: 09:00 16/08/22 To 18:00 16/08/22	Date: 09:00 17/08/22 To 18:00 17/08/22	Date: 09:00 18/08/22 To 18:00 18/08/22	









INITIAL RISK ASSESSMENT

Description	Evaluation	Likelihood	Impact level	Responsibility	Response	Control Measure	ID
Github crashes	I lose all my version history and work	Low	High	GitHub	Restart Project	Keep a secondary backup of most recent working project build	1
My Glasses Break	Harder to focus on code	Low	Low	Me	Get a new pair	Maintain safe usage and storage	2
API Breaks	Front and Backend failures	Low	High	Me	Return to working version	Maintain MVP at all times when tinkering	3
Real life problem requires attention	Stop coding to sort issue	High	Medium	Me	Sort issue and return to coding when able	Sort life issue after work and make sure no commitments are set during work times	4
Project MVP requirements change	Previous work is useless	Low	Medium	Customer	Salvage existing code that can be used and replan	Keep updated with the Customer product scope page	5
Keyboard and Mouse faults	Can't use desktop to code	Medium	Medium	Manufacturer	Go buy new equipment or borrow laptop	Maintain safe peripheral usage	6
Internet goes down	Can't connect to JIRA or gitHub	Medium	Low	Virgin Media	Use Hotspot from Phone	Ethernet connect into router to avoid wifi drops	7
Testing Fails and results in under standard coverage	API is not valid	Low	High	Me	Reverse engineer API from testing failures	Continuous testing to excessive issues at the end	8
Low Productivity	Incomplete or below standard project	Low	High	Me	centre and re-focus	Every morning have a reasonable idea of what can be achieved	9
Main PC breaks	Can't code on my preferred device	Low	Low	Me	Continue on second desktop or laptop	Maintain safe usage and monitor temperatures at application launches	10

LIKELIHOOD	IMPACT LEVEL		
	LOW	MEDIUM	HIGH
LOW	2,10	5	1,3,8,9
MEDIUM	7	6	
HIGH		4	

Link to Risk assessment pdf found in the documentation section of the Repo.

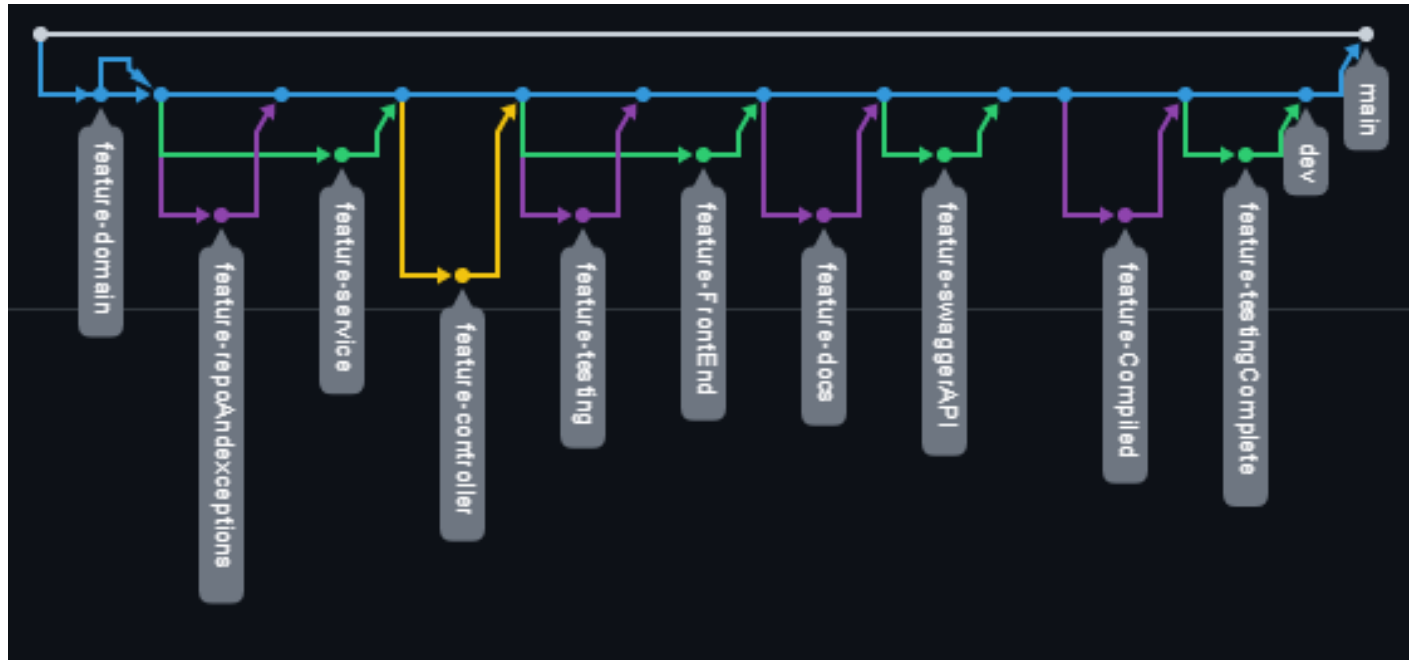
TECHNOLOGY LEARNED

Logo	Technology	Description
	JavaScript	Allows for the creation of interactive web pages.
	CSS	Cascading Style Sheets allows for the styling of elements on HTML docs.
	HTML	Is the building block of webpages that describe their structure & meaning.
	Spring Boot	Application used to fast track creation of services with their frameworks.
	VS Code	Code editor.
	Swagger	Creates documentation for API's.



VERSION CONTROL

- **Version control System:** Git & managed through Github
- **FBM:** Feature branch model was used to ensure MVP was always functional to users when development was occurring.
- **Protection:** Branch protection was in place for Main and Dev branch requiring review before merging.
- **Naming Syntax:** Branches were named with the concept they alter example “feature-Testing”.



TESTING

INTEGRATION

```
@Test
public void getByRatingLessThan() throws Exception {
    //Create an anime to check the output of readById
    List<Anime> result = new ArrayList<>();
    result.add(new Anime(1L, "english", "japanese", 1, 10, 5, true));

    //converts the list to a JSON (As API responds in JSON)
    String resultAsJSON = mapper.writeValueAsString(result);
    mvc.perform(get("/Anime/getByRatingLessThan/10")
        .contentType(MediaType.APPLICATION_JSON))
        .andExpect(content().json(resultAsJSON));
}
```




UNIT

```
@Test
void getAllTest() throws Exception {
    //Create a list to check the output of readAll
    List<Anime> result = new ArrayList<>();
    //Add the single entry to the list
    result.add(new Anime(1L, "english", "japanese", 1, 10, 5, true));

    Mockito.when(service.getAll()).thenReturn(result);

    //converts the list to a JSON (As api responds in JSON)
    String resultAsJSON = mapper.writeValueAsString(result);
    mvc.perform(get("/Anime/getAll")
        .contentType(MediaType.APPLICATION_JSON))
        .andExpect(content().json(resultAsJSON));
}
```

COVERAGE

Element	Coverage	Covered Instructions
▼ Anime_1608_1415	 94.3 %	1,114
> src/main/java	 85.0 %	363
> src/test/java	 99.6 %	751

Analysis:

- A key issue I ran into that was resolved through testing was services returned a list of animes for results that only required one due to their uniqueness.
- Therefore, when this test returned Red, I refactor and optimized until they were Green and coverage was acceptable.



DEMONSTRATION



<https://github.com/CSCairney/HobbyWebApplication-Project>



KANBAN BOARD

SPRINT 1

Initial sprint with, idea generation, risk assessment, user stories created for all users and creation of repo with full FBM & branch protection.

Example of tasks:

As a Developer, I want to have a sensible package structure in my IDE, so that my code is readable by all users

Attach Add a child issue Link issue

To Do

Description

Add a description...

Story point estimate 40

Priority Highest

Child issues Order by 0% Done

HWAP-21	Create Spring Boot API Starter Project	10	TO DO
HWAP-22	Create Packages for Controller	6	TO DO
HWAP-23	Create Package For Domain	6	TO DO
HWAP-24	Create Package For Exceptions	6	TO DO

Projects / Hobby Web Application - Project

HWAP Sprint 1

Initial project setup and clearly outline goals for future to create a smooth production line.

Search C + Epic

TO DO 2 ISSUES

As a Developer, I want a complete risk assesement, so that I can estimate all the potential issue that may arise

AGILE MANAGEMENT

HWAP-1 2

As a Developer, I want to determine my MVP requirements, so that I have a clear goal to achieve

AGILE MANAGEMENT

HWAP-4 18

IN PROGRESS 1 ISSUE

As a Developer, I want to have a sensible package structure in my IDE, so that my code is readable by all users

BACKEND

HWAP-6 40

DONE 3 ISSUES

As a Developer, I want to formulate a web app idea, so that we have a project to work on

AGILE MANAGEMENT

HWAP-2 6

As a Developer, I want to create a story point system for tasks, so that effort can be determined

AGILE MANAGEMENT

HWAP-10 4

As a Developer, I want to create the start of my FBM, so that version control is ensured

AGILE MANAGEMENT

HWAP-3 14



KANBAN BOARD

SPRINT 2

Creation of the API consisting of the rest controller, service, repo and domain. Also, creation of the basic FrontEnd.

Example of tasks:

As a User, I want a website where I can keep a personal watchlist, so I can keep track of my Anime

Attach Add a child issue Link issue

To Do

Description

Add a description...

Story point estimate

36

Priority

High

Child issues

Order by

...

0% Done

- HWAP-35 Create Anime Class with full constructors 6 TO DO
- HWAP-36 Create Getters and Setters 2 TO DO
- HWAP-37 Create Repo that Extends JpaRepository<Ani... 4 TO DO
- HWAP-38 Create Service with methods the controller ... 10 TO DO

HWAP Sprint 2

Create basic FrontEnd and API

Q



Epic

TO DO 1 ISSUE

As a User, I want to be able to remove anime from list, so that I can delete shows

BACKEND

HWAP-53

14

IN PROGRESS 2 ISSUES

As a User, I want to be able to update existing animes, so that details stay consistent like current episode

BACKEND

HWAP-30

12

As a User, I want to be able to read from my anime list, so that I can see whats already on there

FRONTEND

HWAP-31

14

DONE 5 ISSUES

As a Developer, I want a simple webpage, so that users can know where their desired access is

FRONTEND

HWAP-28

16

As a User, I want a website where I can keep a personal watchlist, so I can keep track of my Anime

FRONTEND

HWAP-34

36

As a Developer, I want to create a spring boot application to run our API, so that the frontend can talk to the backend

BACKEND

HWAP-32

14

As a User, I want to be able to add new animes to my list, so that I can keep it up to date

BACKEND








KANBAN BOARD

SPRINT 3

Integration and unit testing of the code. Achieve +80% coverage.
Alter CSS to use more external than inline styling.

Example of tasks:

As a Developer, I want to unit test my code, so that I know each area works individually

 Attach  Add a child issue  Link issue  

To Do ▾





Description

Acceptance Criteria - Coverage for Controller +80% with successful tests

Story point estimate 14

Priority  High

Child issues Order by ▾ ... +
0% Done

- | | | | | |
|---|--|---|---|---------|
|  | HWAP-64 Create Unit tests for Controller | 8 |  | TO DO ▾ |
|  | HWAP-65 Create Unit tests for Service | 6 |  | TO DO ▾ |

Projects / Hobby Web Application - Project

HWAP Sprint 3

Testing and CSS

   Epic ▾

TO DO 1 ISSUE

As a Developer, I want to test the packaging of my application with Maven, so that I know the MVP compiles

AGILE MANAGEMENT

 HWAP-61 12 

IN PROGRESS 1 ISSUE

As a Developer, I want to finalise my README with instruction on testing and build, so that anyone can use it



AGILE MANAGEMENT

 HWAP-62 10 

DONE 5 ISSUES ✓

As a Developer, I want to unit test my code, so that I know each area works individually

BACKEND

 HWAP-57 ✓ 14 

As a Developer, I want more styling in my CSS external, so that additions are automatically styled with IDs

FRONTEND

 HWAP-63 ✓ 8 

As User, I want my database to have quick way to delete individual entries, so that its quicker to remove wrong files

FRONTEND

 HWAP-60 ✓ 10 

As a Developer, I want to integration test my code, so that I know it works as a whole

BACKEND

 HWAP-58 ✓ 20 



KANBAN BOARD

SPRINT 4

Full documentation, bug fixing, testing, ReadMe and Compiling final build.

Example of tasks:

As a developer, I want a full presentation, so that the project can be encapsulated for easy intake

Attach Add a child issue Link issue

To Do

Description

Acceptance Criteria - Full presentation covering all areas.

Story point estimate 40

Priority Highest

Child issues

Order by

0% Done

HWAP-75	Create a powerpoint and choose a theme	8	TO DO
HWAP-76	Slide for Concept	4	TO DO
HWAP-77	Slide for Sprint Plan	4	TO DO
HWAP-78	Slide for Consultant Journey	4	TO DO
HWAP-79	Slide for Version Control	4	TO DO
HWAP-80	Slide for Testing	4	TO DO
HWAP-81	Slide for Review	4	TO DO

HWAP Sprint 4

Documentation

Search filters: Epic Type

TO DO 2 ISSUES

As a Developer, I want a working .gitignore, so that it ignores build-generated files and folders

AGILE MANAGEMENT

HWAP-70 6

As a Developer, I want a swagger document generated, so that the API has fully documented copy of the functioning of the API

AGILE MANAGEMENT

HWAP-73 12

IN PROGRESS 2 ISSUES

As a developer, I want a full presentation, so that the project can be encapsulated for easy intake

AGILE MANAGEMENT

HWAP-74 40

As a User, I want a complete README, so that I know how to use the application

AGILE MANAGEMENT

HWAP-66 8

DONE 4 ISSUES

Create does not refresh imitation of live database view after createBtn is used

BACKEND

HWAP-69 8

Spelling Errors in the ReadMe

AGILE MANAGEMENT

HWAP-68 6

As a Developer, I want a documentation folder, so that progress can be seen through the project

AGILE MANAGEMENT

HWAP-71 4

As a Developer, I want to generated an Entity relationship Diagram, so that the relationships can be visuals from the database

BACKEND

HWAP-72 6

SPRINT REVIEW

- Had appropriate epics
- Instead of one day of testing, testing should've been implemented earlier to identify bugs.
- Bugs were highlighted in sprints which was effective.
- All ideas were achieved to create MVP.
- User Stories could've been more specific as some were too broad and spanned many topics.

FUTURE PLANS

- Automated translation feature from English title to Japanese.
- Link your CrunchyRoll account to automatically update your database with your current episode for shows.
- Hosted on the cloud with a share function for friends to compare.
- Once hosted online – A public section could be generated for community discussion.
- Personalized stats page using custom queries for top 5 shows by input rating.



THANKS FOR LISTENING

Charles Cairney - Links



<https://github.com/CSCairney>



<https://www.linkedin.com/in/charlescairney/>

