RVIT – A REQUIREMENT AND VALUE IDENTIFICATION TOOL FOR PRIORITISATION

Niamh Gillespie – 2549880G

INTRODUCTION & MOTIVATIONS

- Current agile project management tools are often challenging to set up with overly complex user interfaces and a steep learning curve.

- Current tooling also does not tend to place emphasis on the business value that an epic or user story contains. This can lead to the perceived business value differing between stakeholders and developers.

BACKGROUND

- Just-in-time refinement is often used in agile teams to identify requirements and scope epics and user stories.
- The kanban framework can be used to visualise a team's flow of work.
- Visible business values may make it easier for developers to prioritise work and may also change their approach to developing an epic or user story.

- Three existing agile project management tools were reviewed to identify strengths and weaknesses.

Atlassian's Jira

Atlassian's Trello

GitHub's Projects

REQUIREMENTS IDENTIFICATION

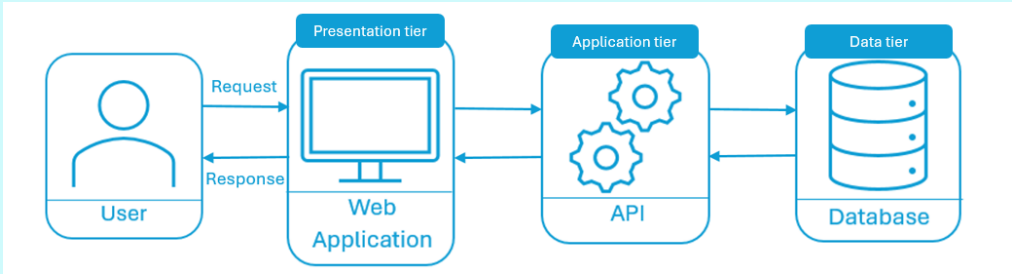
- Requirements identification followed waterfall approach due to time constraints.

Identification process for functional requirements

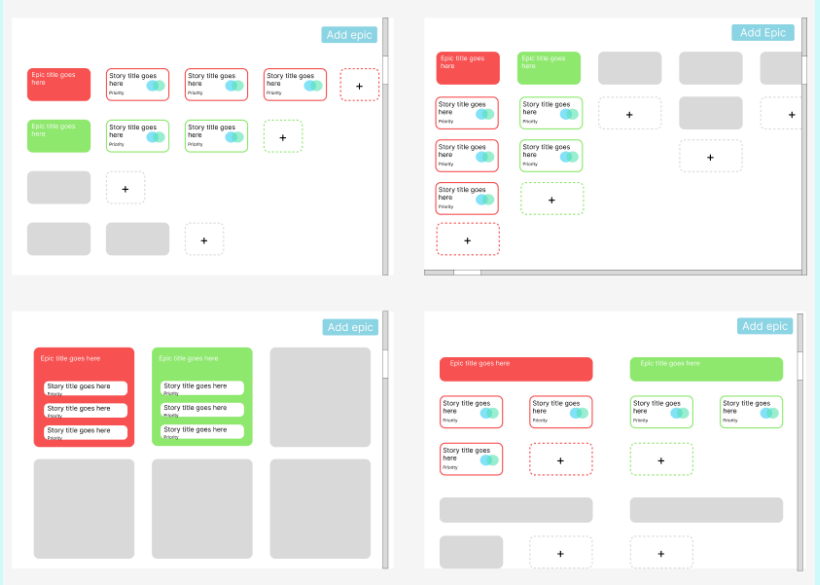
- Non-functional requirements were identified through analysing expected user needs and expectations.

DESIGN

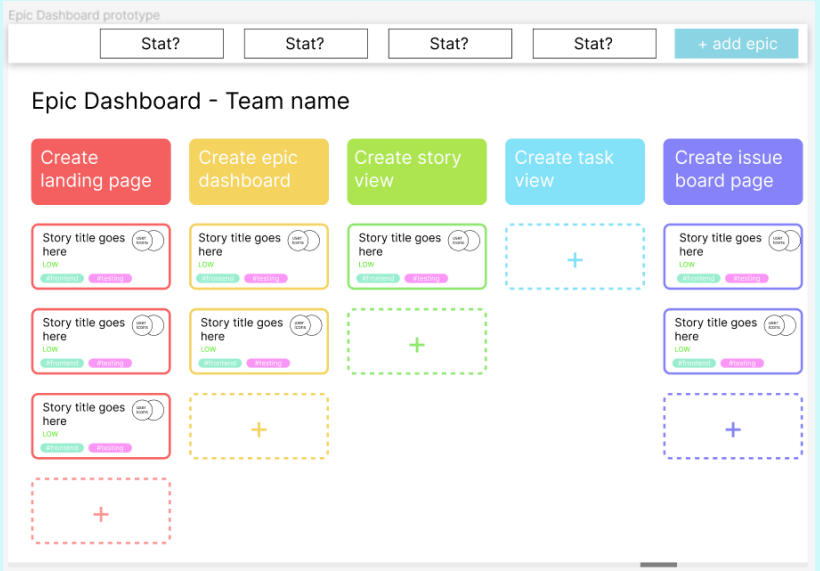
- System architecture based off the three-tier architecture pattern.



- Initial web pages designed on Figma.
- Chose to use Django and React JS for development.



Initial low-fidelity designs for the Epics Dashboard



Finalised high-fidelity design for the Epics Dashboard, focusing on the role of colour

IMPLEMENTATION

- A variety of software engineering techniques were used throughout the development of RViT.

Continuous Integration	Agile Project Management	Sprint-based Development	Code Refactoring

- The frontend is built with a component-based architecture to allow the reuse of components.

- Deployed REST API on Python Anywhere and React JS web application on Netlify.

NON-FUNCTIONAL EVALUATIONS

Unit Testing

96% Django statement coverage.

32% React JavaScript branch coverage.

Performance and Accessibility

Average performance: 98%

Average accessibility: 90%

Average best practices: 96%

Security

9 vulnerable packages.

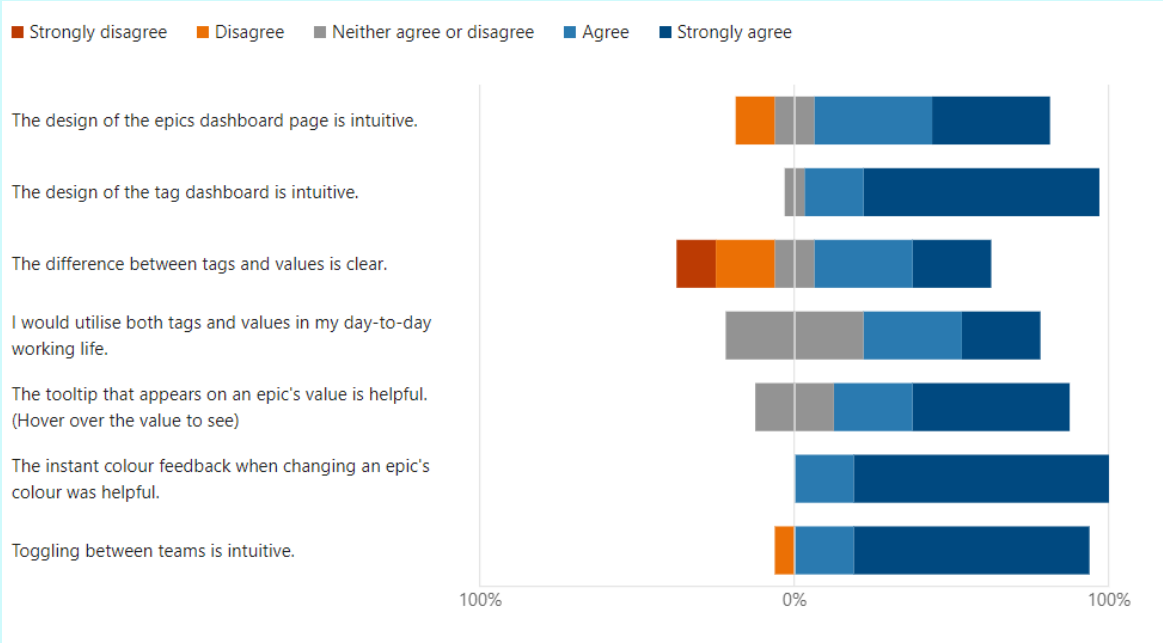
4 cross-site scripting vulnerabilities.

31 hardcoded credentials used.

USABILITY EVALUATION

- User evaluation split into three sets of tasks and one section for final comments.
- 20 full-time developers were recruited for the evaluation, four were asked to complete the pilot study while the others completed the finalised evaluation.

"RViT proved to be far simpler and more user-friendly compared to other tools I've used in the past, such as Jira. … with RViT, I was able to complete tasks swiftly and effectively, as its intuitive design made everything straightforward" ***- quote from a user evaluation participant***



Graph showing how much participants agreed or disagreed with the list of specified statements surrounding the Tag and Epics Dashboards

CONCLUSIONS

- Achieved all the must and should have requirements, with positive outcomes from the various evaluations conducted.
- Many future development opportunities.
- Upon reflection, earlier evaluations would have been extremely helpful.