Vision and Scope Document
for
AI4Agile

Prepared by Emily Cawlfield
Team Katara
Submitted 12/13/20

Table of Contents

1. Business Requirements

1. Create an app that both saves Agile developers time and aids in consistency by assisting in the process to make stories and tasks from an epic.
2. Be able to generate a relationship graph to save time in determining activity dependencies and assignments.

1.1. Background

The AI4Agile product's inception began with a prompt for the ICSE SCORE 2021 competition. It was intended to be a stepping stone for a product with a grander scope, one which would include improved risk management, sprint planning aid, and a range of analytics aspects not yet commonly automated. This project falls under the risk management portion of that scope, assisting in that aspect by providing a more consistent method for going between epics, stories, and tasks, in contrast with the current differing strategies of humans doing these tasks. With a more consistent breakdown, it becomes potentially easier later to provide estimations of effort per activity.

1.2. Business Opportunity

The agile development process involves describing a project in portions called epics, which get broken down further into user stories, and from those stories the work is separated into individual workpieces called tasks. One issue with this process is the time it can take when there are an abundance of epics or stories, and a user isn't necessarily familiar with the text of all of them. A typical customer for this app primarily needs to efficiently refine user stories. This product, for example, would assist with that need by using Artificial Intelligence to generate suggestions for splitting up the user story type. In addition, it would be improving the efficiency of the process by generating graphs for the customer to give them a clearer picture of the dependencies and assignments of already existing issues as they are going through the decomposition process. To achieve these goals, it is imperative that the suggestion processes have sufficiently low waiting times, so as not to slow the overall process but rather accelerate it. For the interface itself, it is necessary for it not to be overly complicated, in order to not inflict unnecessary learning curves to potential users.

1.3. Success Metrics

To determine the success of this project, a list of metrics and their target values has been compiled. With the highest complexity and value being in the AI components, it is those pieces that have the greatest impact on achieving that success. The categories of metrics in the table can be summarized as those concerning time, and those concerning options the user has. These can be traced back to the main nonfunctional requirements for ease-of-use, speed, and more generally the goal of adding value to the user's experience.

No.	Metrics	Target
1	AI understands what is an epic	Minimum 3 Clusters
2	AI understands what is a user story	Minimum 2 prediction
3	AI understands what is a task	Minimum 3 Slices
4	Graphical component	Tree and cluster charts
5	Time to navigate the decomposition	Maximum 2 minutes
6	NLP components	Word2Vec, Doc2Vec
7	Prediction slider	Minimum 2 option levels
8	Options to select or discard suggestion	Yes, no
9	Options to to select different relationship graph	Hierarchy relationships, assignee relationships
10	Bounds of time to decompose an input	From 1 - 5 seconds
11	Variety of types of input that can be decomposed	Paragraph, sentence
12	Response time	From 1 - 2 seconds
13	Percentage of information retained	From 90% - 100%
14	Authentication system	Minimum 1 type of encryption
15	Amount of new output AI can generate	Minimum 1 output

1.4. Vision Statement

The AI4Agile application aims to lessen the administrative time associated with the creation and assignment of stories and tasks in Jira, through use of artificial intelligence and visualization aids.

1.5. Business Risks

The primary business risks that concern this project are the possibility of failing user acceptance testing and failure to complete AI component implementations. The table below describes the severity and mitigation strategies the team will use to manage each of these risks.

Risk	Severity	Mitigation Strategy
Lack of user acceptance	Low	Go through continuous iterations of acceptance testing with a domain expert in UI/UX creation.
Failure to fully implement AI components	Medium	Use an iterative, Agile workflow for development and assess feasibility of the methods used at each step.

1.6. Business Assumptions and Dependencies

Assumptions:

● Jira will be a continually supported and adopted tool for facilitating Agile Development.

Dependencies:

● Jira Platform
● Atlassian Connect Express (ACE)
● Flask

2. Scope and Limitations

2.1. Major Features

1. <u>Epic Decomposition</u>: The process of decomposing an epic, defined here as a compilation of specifications and requirements, into user stories based on related content.
2. <u>Story Optimization</u>: The refinement of existing user stories into new user stories of similar sizes for further assignment to sprints.
3. <u>Task Generation</u>: The creation of tasks from their corresponding user story, with clear deliverables.
4. <u>Dependency Visualization</u>: A graphical component to represent epics, stories, tasks, and the explicit and implicit relationships between them.

2.2. Scope of Initial Release

The scope of this project's initial release is highly dependent on the original project prompt and associated research papers. The included features are to forward the goal of saving developers who follow agile development processes time and effort in Jira. With user story refinement assistance that has reasonably low wait times, the product will provide a time savings for the Agile process that needs to be done prior to grouping tasks into sprints. An easily readable dependency visualization graph included in the scope of the initial release will allow users to be able to see issue hierarchies, dependencies, and assignees at a glance and in a more visual format than what Jira already provides. Figure 2.2.1 contains an overview of the features of AI4Agile, from the more general categories of visualization and user story refinement down to sub-features of the individual processes such as the tuning slider feature that allows users to adjust the outputs of the story optimization and epic decomposition processes.
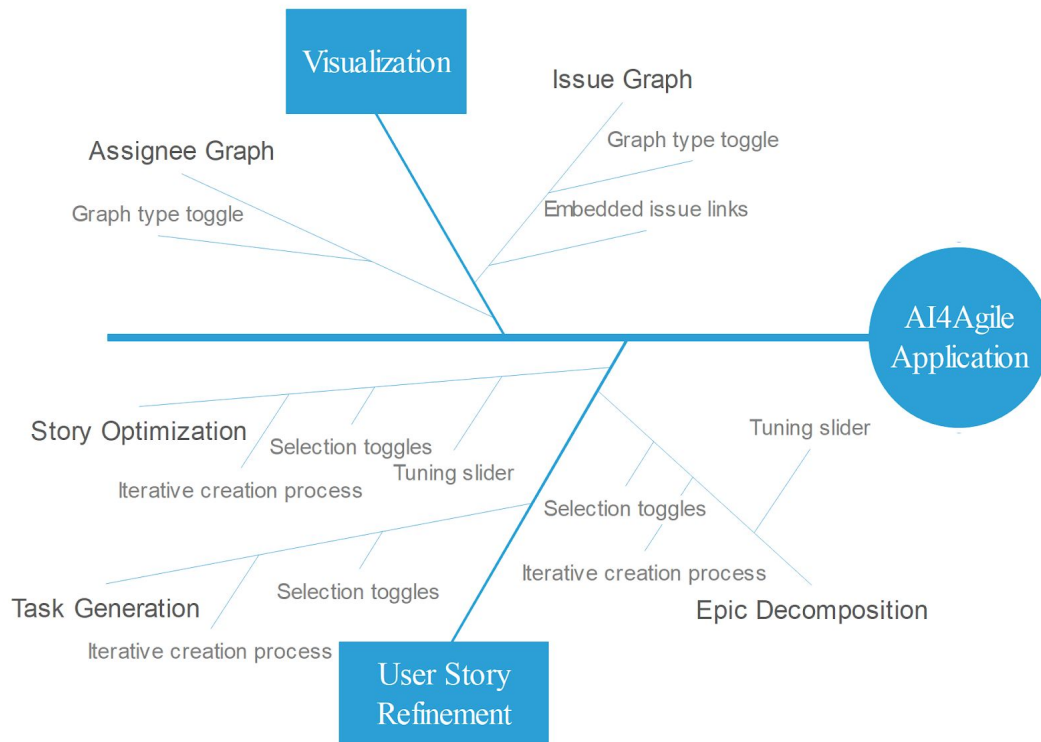
Figure 2.2.1: Feature tree for AI4Agile, grouping the main requirements into two main categories: visualization and user story refinement. Under each category are listed two further levels of the application's features based on significance.

2.3. Scope of Subsequent Releases

2021 Incremental Releases Projected Features

1. Availability in Atlassian Marketplace
2. Dynamic clustering for epic decomposition
3. Improved task generation
4. Improved story optimization speed
5. Suggestion results caching
6. Additional relationship and field type options for graphs
7. Graph filtering options
8. Ease-of-use controls for graphs

The most significant feature for the upcoming releases of this project is its deployment to the Atlassian Marketplace, which will affect the specifics of this feature set due to customer feedback. The other projected features focus on improvements to the initial release major features.

2.4. Limitations and Exclusions

Due to some limitations of the release platform and certain time and scope constraints, the following features which users of stakeholders might anticipate are not planned to be included in upcoming releases.

- Word cloud to reveal backend clustering characteristics
- Undo/redo functionality for created issues
- Directly editable graph relationships
- Automatic refreshing of web panels to display newly created issues

3. Business Context
3.1. Stakeholder Profiles

The table below describes the major value, attitudes, and interests of each major category of stakeholder, as well as any constraints they may impose on the project. Some attitudes and interests have been projected for lack of data and based on assumptions from the nature of certain features and the problems they target.

Stakeholder	Major Value	Attitudes	Major Interests	Constraints
Developers	valuable resume item and experience	see product as a learning opportunity	marketable and demonstrable product	must hit major milestones on time
Mentor	successful product story	highly receptive, but expect high usability	time savings; ease of use	must hit major milestones on time
Research Sponsor	additional proof of research concept	has high hopes for product abilities	epic decomposition; story optimization; task generation; dependency visualization	must have graph and the three stages of user story refinement
Faculty Mentor	academic prowess	see product as competitive academic opportunity	visual presentation	must be demonstrable
Scrum Master users	time savings; improved productivity	interest in the product being able to save time	epic decomposition, story optimization dependency visualization	usage complexity and time to learn must be low

Software Requirements Analyst users	time savings; semi-automation of tasks	doubts towards the accuracy of suggestions due to lack of expertise	epic decomposition	usage complexity and time to learn must be low
Software Project Manager users	time savings; alternate information form	excited for ease of information for decision-making	dependency visualization graph	graph must be easily understood

3.2. Project Priorities

The priorities for this project are laid out below, each in terms of what is driving or constraining them as applicable, and the degree of freedom decided to be allowed for each category within this project. Of these priority dimensions, the least restricted is staff, as changes in this area aren't expected to vary wildly or have significant effects on project success. The most important dimensions are features and schedule, as more of the success metrics mentioned in section 1.3 are affected by the implementation of features rather than their individual quality, and the project is operating within the limits of academic and SCORE 2021 competition timing.

Dimension	Driver	Constraint	Degree of Freedom
Schedule	release 1.0 to be available by 12/11, release 1.1 by 01/10, release 2.0 by 04/30		0
Features			80-90% of high priority features must be included in release 1.0
Quality			70-75% of user acceptance tests must pass for release 1.0, 90-95% for release 1.1
Staff		maximum team size is 1 PM, 2 mentors, and 5 developers	

3.3. Deployment Considerations

The Jira platform has a variety of forms, project types, and an accompanying mobile application. As a result, universal support across the platform will not be supported. Additionally, Jira will be receiving a UI overhaul in March 2021 which may result in further long term maintenance. The

deployment of this application is focused on the browser version of Jira Cloud. User access to the system is dependent on publishing to the Atlassian Marketplace. The architecture of the application poses an additional consideration, as the bulk of the code is written in Python and thus needs to be hosted to run.