

Best practices for using Jira in a client company

Case: Finnish IT and system development company



Bachelor thesis

Degree Program in Business Information Technology
fall, 2023

Piia Halttunen

Tietojenkäsittelyn koulutus / Business information technology	Tiivistelmä
Tekijä Piia Halttunen	Vuosi 2023
Työn nimi Best practises for using Jira in a client company	
Ohjaaja Lasse Seppänen	

TIIVISTELMÄ

Opinnäytetyön tarkoituksena oli selvittää, mitkä ovat parhaat tavat käyttää Jira Softwarea (Jira) asiakasyrityksen erään liiketoimintayksikön implementointiprojekteissa. Opinnäytetyössä tutustutaan Jiraan ensin tietokehyksen kautta sekä selvitetään kyseisen liiketoimintayksikön työntekijöiden kokemuksia Jiran käytöstä. Näiden perusteella, osana opinnäytetyötä, luotiin ohjeet ja käytännöt siihen, miten Jiraa tullaan käyttämään Yritys X:n valitussa business yksikössä. Opinnäytetyön toimeksiantajana oli suomalainen teollisuuden ja logistiikan sovelluksia kehittävä yritys.

Opinnäytetyön tietopohja perustuu Atlassianin (Jiran kehittäjä) ohjeisiin Jiran käytöstä. Tämän lisäksi tietopohjassa käytettiin myös muita Internetin lähteitä. Tutkimusaineiston tärkein tavoite oli ymmärtää valitun business yksikön työntekijöiden Jiran käyttökokemukset, jotta niiden avulla voitiin luoda parhaat mahdolliset käytännöt tulevaisuuden projekteille. Tutkimusaineisto kerättiin kyselytutkimuksella ja aineisto analysoitiin sisällönanalyysillä.

Johtopäätöksenä opinnäytetyöstä voidaan todeta, että yhteiset ohjeet Jiran käytölle olivat tarpeen. Lopputuloksena opinnäytetyöstä syntyi konkreettiset ohjeet, joiden avulla business yksikkö hyödyntää Jiraa parhaiten tulevissa implementointi projekteissaan. Kehittämistyön perusteella suositellaan, että parhaita käytäntöjä noudatetaan jatkossa kaikissa implementointi projekteissa, jotta luodaan johdonmukaisuutta projektien välillä.

Avainsanat Jira, Projekti, Ohjeet, Käytännöt

Sivut 33 sivua ja liitteitä 22 sivua

Author Piia Halttunen

Year 2023

Subject Best practices for using Jira in a client company

Supervisors Lasse Seppänen

ABSTRACT

The purpose of this thesis was to create the best practices for client company's one business area for using Jira Software (Jira) in implementation projects. Jira is first studied in the thesis through theoretical framework and by asking the employees of the business area their experiences on using Jira. Based on the gathered information, the best practices for using Jira were created as a part of the thesis. The thesis was commissioned by a Finnish industry and logistics software development company.

The theoretical framework of the thesis is based on Atlassian's (Jira's developer) guidelines on how Jira is planned to be used. In addition to Atlassian's instructions, other Internet references were used. The main goal for the research was to understand the current user experience of the employees in the business area to be able to create the best practices for future implementation projects. The research was conducted by a survey and the results were analyzed by content analysis.

The research demonstrates that the need for common best practices was there. As a result of the thesis, the commissioner has concrete instructions and guidelines on how to use Jira in their future implementation projects. Based on the analysis, it can be suggested that the best practices are followed in every implementation project, to create consistency between the different projects.

Keywords Jira, Project, Best Practices, Instructions

Pages 33 pages and appendices 22 pages

Glossary

Agile	Project methodology which Jira is based on
Atlassian	Developer of Jira Software
Bitbucket	Version control tool, created by Atlassian
Board	A table of Issues
Confluence	Documentation tool, created by Atlassian
Epic	Issue type used in Jira
Gadget	A reporting tool, where different configurations can be made to follow the project
Integration	When two or more systems are communicating together
Issue	A sort of a card that are followed in Jira, includes the details of a tasks that needs to be done
Jira	Project management tool
QA Review	Quality assurance review, a.k.a. internal review for a task
Status	A describing word or words to tell what the situation of an issue or issues is, for example "in progress"
Task	Issue type used in Jira
Use case	A description for a user's need for a system
Workflow	The flow of different status which an Issue can travel through the project

Contents

1	Introduction	1
2	Jira software.....	2
2.1	History of Jira.....	2
2.1.1	Development of Jira.....	2
2.1.2	The differences of the three main aspects of Jira	4
2.2	Using Jira.....	5
2.2.1	Projects	5
2.2.2	Boards	6
2.2.3	Issues and issue types.....	7
2.2.4	Workflows.....	8
2.3	Reports and dashboards.....	9
2.3.1	Reports.....	9
2.3.2	Dashboards	9
2.4	Atlassian’s other applications and Jira integrations.....	10
2.4.1	Confluence	10
2.4.2	BitBucket cloud	11
2.4.3	Jira integrations	12
3	Conducting the best practices	14
4	User experience survey.....	16
4.1	User experience – background.....	16
4.2	Open text answers.....	19
4.3	Conclusions.....	21
5	Best practices for using Jira in client company.....	23
5.1	Using Jira.....	23
5.1.1	Boards	24
5.1.2	Issues and issue types.....	25
5.1.3	Workflows.....	26
5.2	Reports and dashboards.....	27
5.2.1	Reports.....	27
5.2.2	Dashboards	28
5.3	Atlassian’s other applications and Jira integrations.....	28
5.3.1	Confluence	28
5.3.2	BitBucket cloud	29

5.3.3 Jira integrations	29
6 Summary	30
References.....	31

Figures, program codes, commands, and tables

Figure 1 Development points of Jira (Saleh, 2021)	3
Figure 2 Maturity of the respondents.....	17
Figure 3 Respondent roles	18
Figure 4 Respondent role and maturity	19
Figure 5 Client company’s project best practices	24
Figure 6 Issue statuses	25
Figure 7 Jira workflow	27
Figure 8 Standard reports	28
Program code 1 An example of an advanced query.....	6
Quatation 1 Open text comment A	20
Quatation 2 Open text comment B.....	20
Quatation 3 Open text comment C.....	20
Quatation 4 Open text comment D	20
Quatation 5 Open text comment E	20
Quatation 6 Open text comment F	21
Quatation 7 Open text comment G	21

Annexes

- Annex 1 Material management plan
- Annex 2 User experience responses
- Annex 3 Instructions for using Jira in the chosen business area

1 Introduction

This thesis will focus on the project management and development tool Jira Software (“Jira”) and creating the best practices of using Jira in the client company. The thesis will study the best ways of using Jira functionalities and the result of the thesis will give the client company clear guidelines on how to use Jira most efficiently in the company’s IT and system development projects.

The thesis is done for the author’s employer company, where the author has been employed since 2020. The company is a Finnish based IT and system development organization, it has offices in five different countries, mainly in the Nordics, and it has over 350 employees. The company’s yearly revenue is more than 44 million euros and they operate in three different key areas: discrete manufacturing, process industries and logistic and retail.

In order to keep the scope of the thesis more manageable, the thesis will focus on creating the best practices for a specific business area and the Jira usage of the other two business areas of the client company will not be analyzed. As the client company is using Jira in both implementation projects and as a service desk tool, it will cover the best practices for both ways, but focusing more on implementation projects as requested by the client. The thesis will also include the add-ons and extensions that Jira has to offer and evaluate the need and usefulness for them for this specific company.

The thesis will answer the following questions:

- What are the business area’s employee’s current experiences on using Jira?
- What is the best way to use Jira in the business area?
- In what ways should Jira be used in the business area to help lower project costs?
- In what ways should Jira be used in the business area to make the project life cycle more efficient?
- What are the differences between using Jira in an implementation project versus as a service desk tool?

By answering these questions, the outcome of the thesis will be guidelines on using Jira in the client company’s chosen business area.

2 Jira software

This chapter will give a glance at the history of Jira Software and explain the scenario for which the tool was created. The chapter will also describe the best practices for using Jira, by explaining the tools it holds, as it is meant by the creators of Jira. The last subchapter will cover the extensions of Jira.

Jira is an agile project management tool which supports, in addition to the widely known Scrum and Kanban, any agile methodology the user would prefer. Agile as a word means adaptiveness and response to change. As a project management methodology, it encourages to accept and adapt to change. With Agile the team creates their own set of conventions that they all agree to follow and focuses on team members doing the work and how they work together. (AgileAlliance, n.d.; Atlassian, n.d.-a)

2.1 History of Jira

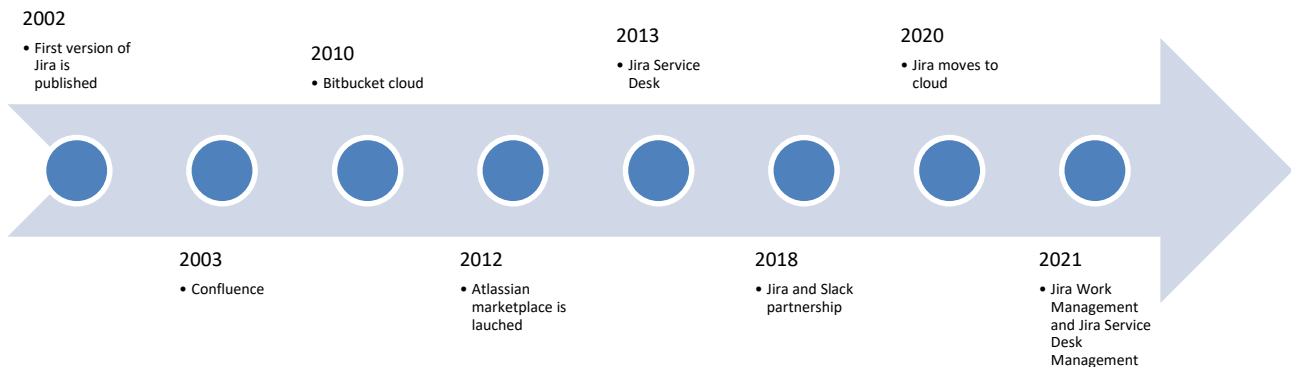
The first version of Jira software is published by an Australian software company called Atlassian. The co-founders, Mike Cannon-Brooks and Scott Farquhar started Atlassian in 2001 as a tech support service and the tool, Jira, was originally designed for their own software developers. The co-founders decided to start selling the tool in addition to their support service in 2002. Since its beginning, the usage of Jira has spread to companies from various fields. Today it is used in 190 countries by over 180 000 users. More than 400 of the Fortune 500 companies use Atlassian's products. Jira has many purposes, the most common one is to use Jira as a project, task, and product management tool. It is widely used also as a bug tracking, software development, DevOps, and Agile teams' tool. (Atlassian, n.d.-m; Saleh, 2021)

2.1.1 Development of Jira

As the tool has been used already for more than 20 years, it has had constant development and crucial milestones over the years. Today, Jira consists of three main aspects. Jira Work Management is the framework on which other functionalities are built on top and it is the tool for project management in general. Jira Software brings agile project management features on top of Jira Work Management. Jira Service Management is the tool to use in service desk or IT

operations. *Figure 1* shows the most important milestones from Atlassian's history that have the biggest impact to modern day Jira. (Saleh, 2021)

Figure 1 Development points of Jira (Saleh, 2021)



Confluence is the first extension of Jira. It was published in 2003 and it was created for a development collaboration platform. Subchapter 2.3.1. will describe Confluence in more detail. In 2010 Bitbucket was introduced as a code hosting platform to provide development teams a place to collaborate with their code and guide the team through the development process. Atlassian marketplace was launched in 2012 which shifted Jira from developer tool to project management tool by enabling the third-party developers to build project management plugins for Jira. 2013 the first version of Jira Service Desk is published, today the Jira Service Desk is known as Jira Service Management. (Bitbucket, n.d., 2018; Saleh, 2021)

Jira starts a partnership with Slack (communication application) in 2018 to simplify the development process and communication of development teams. The integration between the two apps can be used for example to connect Slack channel to a specific Jira project, to get personal Jira notifications in Slack instead of email, to interact with Jira issues through Slack, to accessing Jira issues without leaving Slack and to inviting users to Jira through Slack channels. In 2020 Atlassian moves Jira to cloud by completely decommissioning Jira Servers and a year later, in

2021, Jira Work Management and Jira Service Desk Management are created. (Atlassian, n.d.-k; Saleh, 2021; Slack, 2017)

2.1.2 The differences of the three main aspects of Jira

The three main aspects of Jira are Jira Core or Jira Work Management, Jira Software and Jira Service Desk. All the three aspects have the same look and feel, providing a consistent experience, and they can be purchased either together or separately. (Akeles, n.d.)

Jira Work Management is the core of Jira, and it is directed to business projects and non-technical teams such as business users, legal, HR, Marketing, Finance and Operations Teams. Jira Work Management could be used for example to track advertising campaigns, on-boarding new employees, task management, workflow approvals, change requests and in general project team collaborations. It has pre-made templates for project, process, and task management and it is a stripped down version of Jira software to make it less confusing to business users. (Akeles, n.d.)

Jira software is intended for developers by adding the integration of developer tools such as Bitbucket. The targeted users for Jira software are software developers, Scrum Masters, Project Managers and Quality Assurance Engineers. Jira software could be used in Agile and basic software development projects, bug tracking, and in product management. In addition to the templates in Jira Work Management, Jira software includes Scrum, Kanban, and basic software development templates. (Akeles, n.d.)

Jira Service Desk, on the other hand, is targeted to customers, help desk agents, call center managers, non-technical users and for IT helpdesk, helpdesk ticketing, incident, change, and problem management. Jira Service Desk is recommended to be integrated with Atlassian Confluence for integrated ticketing tool and knowledge base. It is also good to note that some features are not accessible by the End users, such as editing tickets, workflow status transit, searching, and dashboards. Jira Service Desk includes IT and Basic Service Desk templates and the same templates as Jira Work Management. (Akeles, n.d.)

2.2 Using Jira

The fundamentals of using Jira are projects, boards, issues, and workflows. The following subchapters will dive deeper into all the fundamentals to get the understanding of how Jira software is meant to be used. (Saleh, 2021)

2.2.1 Projects

A project is defined as a combination of agreed objectives that are set to be accomplished in a fixed time. In Jira Software a project is described as a container that organizes and tracks all the tasks that are required to finish the project. A project in Jira software is customizable so it can fit the needs for all kinds of teams and projects. Issues, people, and workflows are the key elements of a project. Issues describe what needs there are for the project. People to describe who will do the needed tasks and workflows to help the project management to follow the status of the project. Each Jira project includes, by default, at least one board. Jira software offers multiple project templates from the most popular Kanban and Scrum boards to more tailored templates for specific teams like HR or marketing. Project boards are opened in more detail in subchapter 2.2.2. (Atlassian, n.d.-o; Simplilearn, 2023)

Jira has two types of projects, a Team-Managed project and a Company-Managed project and they differ in the way they are managed. The Team-Managed project is for simplified projects and for teams who want to be in more control of their processes and practices. The Company-Managed projects are directed to teams who want to work across multiple teams and multiple projects. Company-Managed projects should be used when company management wants the company to follow the same best practices throughout the company. Team-Managed projects are all individual projects, and the settings can be changed without affecting other projects under the company's Jira. When updating the Company-Managed project settings, all the projects opened under the same company's Jira will have the same settings. The Company-Managed projects are managed by the company's named admins when the Team-Managed projects are managed by anyone in the team. (Atlassian, n.d.-o)

2.2.2 Boards

Jira software boards are used to visualize the project teams' tasks on cards that can move between columns. The columns represent the status of the task and make it easier for the team to follow the tasks and project progress. Task cards are called issues, which will be processed in more detail in subchapter 2.2.3. Each project board will be created with default columns of "To Do", "In Progress" and "Done", but statuses can be modified, deleted, or added new once. The pre-configured Jira boards are Kanban and Scrum, but besides those, Jira also offers pre-made templates, which anyone can use. The pre-made templates go from Project management to recruitment to tracking budget creation. The templates can be found by simply searching the Internet with keywords "Jira templates". (Atlassian, n.d.-aa, n.d.-n)

Kanban template visualizes the work with simple but powerful board. Kanban template is created for teams who focus on creating continuous workflow. Scrum board on the other hand is created to break down large pieces of work into smaller, manageable pieces. The basics of Scrum is to divide the work into sprints that are a fixed period where agreed number of issues are delivered. The Scrum template also includes a board where the team can organize the workload based on priority. Project teams that work with large projects or multiple workstreams, can use cross-project boards to track and manage different types of work more easily. (Atlassian, n.d.-o, n.d.-aa)

Jira boards allows the creation of filters with JQL to only see issues that the user would like to analyze. JQL or Jira Query Language is made to ask complex questions by creating queries. The queries can be very basic e.g., JiraKey-001 (to search for a specific issue, from a specific project) or more advanced queries where Jira editor helps the user by giving hints while the query is typed. An example of an advanced query is shown in *Program code 1*.

Program code 1 An example of an advanced query

```
Priority in (Medium) AND project in (ProjectX) AND assignee is EMPTY
```

The *Program code 1* will filter out all issues with priority medium from Project X that has no assignee. The JQL and Structured Query Language (SQL) are very similar and those who have used SQL are able to easily understand and execute JQL queries. The filters can be saved to make them easier to access again. (Amoeboids, n.d.; Atlassian, n.d.-x, n.d.-p; AWS, n.d.)

2.2.3 Issues and issue types

Jira issues are used to manage the project tasks. Issues can help the project team to manage code and estimate and track workload. Issue types are story, bug, and task. Projects created in Jira will by default come with one parent issue type, Epic. Each issue will have their own unique key, which consists of a project key and a sequence number. The project key is created during the project creation and consists of alphanumerical characters. The project key needs to be at least two characters and start with an uppercase letter. By default, each of the three main aspect of Jira consists from different issues and issue types. (Atlassian, n.d.-ac, n.d.-y)

Issues can be divided hierarchically in three levels: Epic issues, which are the parent issues, standard issues that include bugs, tasks, and stories, and lastly, subtask issues. Subtasks are used to break down the other types of issues, which can be helpful for example in cases where one issue needs to be worked on by multiple team members. Subtasks are not included in time tracking and issue views are limited to show only 500 subtasks at a time. Subtask can be converted to an issue if there is a need for more closer follow up or the subtask is holding the issue resolution. An issue can also be converted as a subtask, if it is in the end seen as a part of bigger task. Subtasks can only be used if they have been enabled by the company administrators. A subtask will inherit project, security level and sprint value (if used) from the parent task. A parent issue can be either a bug, a story, or a task type of issue. (Atlassian, n.d.-y, n.d.-h)

Epic is a large structure of work that can be broken down to smaller pieces of work, for example to user stories or tasks. Epics can be tracked on multiple boards and are targeted to guide multiple teams towards the same goal. Epics are usually used flexibly over the project sprints and updated according to the needs of the project team and customer feedback. An Epic can be created multiple ways, but with each way there are two fields that are mandatory: Epic name and Epic summary. Epic name will be shown in as a label in each issue that belongs to a certain Epic. Epic

summary will be shown as an identifier for the Epic, whenever Jira shows the specific Epic. (Drozd, n.d.; Rehkopf, n.d.-a)

Epics can be used to either break down a project to manageable pieces or to collect and store smaller pieces of tasks together. It is though recommended to use Epics only one way per project, to make it less confusing for the team. Epics should also only be created when they are planned to be finished, to make the Epic report and Epic panel more readable. (Idalko, n.d.)

A story or user story is used specially in development projects to describe a software feature from the end user perspective. User stories are often formed by answering the three questions of: “To whom?, What? and Why?”. A user story should be informal general explanation of a feature that should be built during the project. User stories have multiple benefits that include keeping the focus on the end-user, they enable collaboration, create momentum, and drive creative solutions. When working with user stories, the team breaks the story down to smaller tasks, estimates the tasks and by the created estimates, the team can now decide which stories can be included in the next sprint. (Rehkopf, n.d.-b)

When software development team member or a user finds that something is not working in their system, they can record it by creating a bug or a task type issue. Tasks are meant to represent work that needs to be done and bugs are meant to report a problem that impairs or prevents the functions of the system. Bugs are mainly used in Jira Service Desk type of projects. (Atlassian, n.d.-e, n.d.-y)

2.2.4 Workflows

A workflow is the path that an issue takes when it moves from the creation of the issue to the point that there is no more work left for that specific issue. The workflow consists of a set of statuses and transitions that the issue moves through during the process of working the issue. With a Team-Managed project the project admins can create, edit, or delete the workflows inside that Team-Managed project. In Company-Managed project the company’s Jira Administrator with global permissions can edit or create a more advanced workflow. (Atlassian, n.d.-z)

2.3 Reports and dashboards

This chapter will dive into the reports and dashboards that Jira offers. The chapter will give basic information on what kind of reports and dashboards Jira has and how to create them. The subchapter 2.3.1 dives deeper into Jira reports and subchapter 2.3.2 opens up the usefulness of Jira dashboards.

2.3.1 Reports

Each Jira project includes in total 12 standard reports from time tracking, user workload report, resolution time report to created vs resolved issue report, to mention a few. These 12 reports are available without any need for user configuration. In addition to the standard reports, Jira offers the creation of custom gadgets to create more detailed reports and dashboards (dashboards are explained in subchapter 2.3.2). To create a new gadget the user can create query in the issue navigation page (queries are covered in subchapter 2.2.2. Boards). The query will be shared with a reporting gadget which will then be transferred to different kinds of charts to show the data more visually. There are more than 20 pre-installed Jira gadgets, and many other gadgets are available that will retrieve data from other Atlassian systems. (Atlassian, n.d.-r)

Jira reports are divided into three categories: Scrum, Kanban, and work management. For Scrum teams there are four pre-made reports to use, sprint report, burndown chart, release burndown and velocity chart. For Kanban teams Jira offers cumulative flow diagram and control chart report. Work management has two categories, issue analysis and forecast & management. Issue analysis reports help identify trends by focusing on following topics: average age report, created vs resolved issues report, pie chart report, recently created issues report, resolution time report, single level group by report and time since issues report. Lastly, the forecast & management has time tracking report, user workload report and version workload reports to choose from. (Atlassian, n.d.-s)

2.3.2 Dashboards

Jira dashboards collect all the created gadgets and pre-installed data displays into one place. A dashboard can hold information from one or multiple projects and they can be either private or shared. Dashboards are modifiable, the administrator can decide on how many columns they want

to show the gadgets on, in which order they are shown and if they want to show only the default gadgets or create gadgets by themselves. (Atlassian, n.d.-ab, n.d.-q)

2.4 Atlassian's other applications and Jira integrations

This chapter will describe Atlassian's other products and how they work with Jira. Atlassian has also made integrations to Jira with other system providers and those will be covered in this chapter as well.

Atlassian has other products besides of Jira and they can be divided in four different categories: Products for all teams, which include Jira (and its different aspects), Opsgenie, and Statuspage. Collaborating tools, that include Confluence, Trello, and Atlas. Tools for development team are: Bitbucket, SourceTree, Compass, Bamboo, Fisheye, and Crucible. Lastly Atlassian has three products for security and identity: AtlassianAccess, Crowd, and Beacon that is still in beta use. The following chapters will dive deeper with Confluence, Bitbucket and lastly give understanding of the extend of the possible integrations offered to connect Jira with. (Atlassian, n.d.-t)

2.4.1 Confluence

Confluence is a workspace created for teams to share knowledge, collaborate, create, and organize their work. It is suitable for teams of all sizes and types. Depending on the needs of the team, Confluence can be located either in the Cloud, Data Center, or your own server. All the work stored in Confluence, lives in Pages and there are templates for almost any kind of content. The pages are stored in workspaces where the team can collaborate and keep their content organized. (Atlassian, n.d.-g)

For integrated use of Jira and Confluence the site admin can create the link between the two tools and there is no longer a need to share files, folders, or drives between team members as everyone has direct access to the documentation. It is also possible to add, view and edit Confluence pages inside Jira, without having the need to even switch to the Confluence side. The connection also works the other way, issues can be created from Confluence by typing `"/jira"` while editing a page in Confluence. It is also possible to track Jira issues in Confluence by pasting a link to a Jira issue and after being published, the confluence page will show the details of the issue. Jira roadmap can also be added a Confluence page to share the team's planned work and to provide status updates.

From the linked Jira project, it is also possible to create Jira template issue reports to a Confluence page to have improved visibility to the whole team and across all stakeholders. All these integrations are created to help find everything in one place without having the need to manually copy and paste content from tool to another. (Atlassian, n.d.-v)

The most common way of using Confluence and Jira together is to use it with Jira Service Management as a knowledge base. The knowledge base gives the service management team a place to store all project-related information and knowledge. Every new team member can easily find the needed information for the specific project. The customer can have access to the help center and it can be used to share information and articles for the customer. By creating a restricted page, the information of the specific page won't appear in the Confluence help center, but is accessible for the team that has been granted access to it. (Atlassian, n.d.-u, n.d.-w)

The Confluence knowledge base can be standardized with templates and customized with project specific logo. As in other Confluence pages, Jira issues can be linked to the knowledge base to track specific issues. By labeling the content in Confluence, it will be easier to find related information. The integration between Confluence knowledge base and Jira Service management can also be utilized by creating a new article directly from a service desk issue. Knowledge base articles can also be recommended in Jira Service Management side, when customers for example starts typing their request by enabling auto-search. Confluence provides an opportunity to also track the success of the knowledge base articles by creating reports that illustrate for example how often an article is shared, viewed, or voted as helpful. A knowledge base article can be so helpful that it solves the request entirely before it is even created. (Atlassian, n.d.-u)

2.4.2 BitBucket cloud

Bitbucket cloud is a cloud hosted tool where the entire software team can collaborate with the project code. BitBucket is integrable with Jira, it has built-in CI/CD, and it is a native Git tool. CI/CD comes from continuous integrations/continuous delivery or deployment. CI/CD is software development and deployment practice that relies on incremental code changes and automation to deliver changes fast and error free to production. Git is a version control system that tracks the changes made to the files stored in Git. As it allows changes made by multiple people to be

merged into one source, it makes the teams collaboration easier. (Atlassian, n.d.-f; NobleDesktop, 2022; Synopsys, n.d.)

Bitbucket and Jira can be integrated for seamless workflow. Jira issues can be created, updated, commented, and explored without leaving Bitbucket. With automated rules, Jira issues can be updated without the developer needing to do anything extra than working the code through Bitbucket. Atlassian offers a complete tool selection for DevOps teams with Bitbucket, Jira Software, Confluence and Opsgenie. DevOps is a software development methodology that combines both development and operations to build and deploy software iteratively and faster. Opsgenie is an Atlassian tool for incident management. (Atlassian, n.d.-d, n.d.-ad; GitLab, n.d.)

2.4.3 Jira integrations

There are over 3 000 different types of software that can integrate with Jira. The integrated systems are called either apps, add-ons, or plugins. All the needed integrable apps can be found from Atlassian Marketplace. Jira is integrable with the most known tools providers such Slack, Microsoft, Google and Zoom. From Marketplace, you can also find more smaller market apps. Most of the integrable tools are easy to set up and free. (Atlassian, n.d.-l)

As an example of tools and apps that can be integrated with Jira for design teams Atlassian recommends software's like AdobeXD, Invision, Figma and Gliffy. These can be used by attaching the created designs to Jira issues and Confluence pages. With the integrations you can include specifications, assets and guides directly with the design file. Communication will be smoother with two-way information sharing and the design team will be able to get feedback on the design directly from the Jira ticket. (Atlassian, n.d.-i)

For IT teams Atlassian recommends tools like Dynatrace, GitHub, Slack. These tools are used for monitoring, supporting, and maintaining the IT infrastructure. By integrating the existing software's and support channels, you will be able to create a single customer portal for streamlined workflows. (Atlassian, n.d.-b)

Business teams could benefit from integrating to apps like Slack, Microsoft office tools, Gmail, and Atlassian's own products Trello and Confluence. With the recommended integrations you would be able to for example forecast release date, get visibility across multiple projects, linked issues,

and relationships between issues. Managing various sizes of projects will be easier by integrating with different tools and having a simple one glance place for all the projects. (Atlassian, n.d.-j)

The software development team's work can be improved by having custom automation, integrated test management and quick code review. For these tasks Atlassian recommends connecting Jira with for example BitBucket Cloud, Optimizely, Jenkins and LaunchDarkly. The add-ons can be used to track the development activities on Jira issues, to access code quickly by syncing the teams work to a project in Jira or by dashboards with real-time data to improve the team's performance. (Atlassian, n.d.-c)

3 Conducting the best practices

The purpose of this thesis is to create best practices for using Jira in client company's business area. The best practices will be used in implementation projects that vary from small projects that last for a few months to big, multisite, and multiyear projects. The best practices will be instructions for project managers and developers to follow, update and manage the project tasks the same way using the same tool in projects throughout the business area. The theory of using Jira is gathered from the internet, mainly from Jira's creators own webpage. As a part of the knowledge base for this thesis, a user experience survey is conducted for the business area's employees. The best practices are based on the user experience and on the theory of Jira.

A participatory research type from the action study method was used for this thesis. Action study method means that the research will simultaneously try to answer the research questions and find a solution for them. This method was chosen to get a picture of the current obstacles and challenges of using Jira in client company's chosen business area. By gathering the comments from current usage, the best practices created during the thesis would in best case help with the adaptation of the best practices and better answer the needs of the users.

The theory of Jira from chapter 2 is the base for the best practices. The responses from the conducted survey gave insights in additions to the theory, on what topics the best practices should focus on, in addition to the basics of Jira. For example, difficulties with combining workload estimation was raised in few of the comments and because of that, it was one the topics that were focused on when creating the best practices.

During the creation of the best practices, the instructions was gone through with the business area country director, two developers and one consultant to get comments from other roles and users besides the author of the thesis. These persons were selected to give comments, because of their experience in system development work and to get feedback from the roles who will be mostly using Jira based on the created best practices. The best practices were updated based on the comments of the other employees of the business area.

The long-term goal of the best practices is to develop the day-to-day working methods in the business area by having a standard way of working in all the business area's projects. As the company is growing steadily, the need for common guidelines is getting more important every

day. The best practices will be tested in the next project starting in the fall and updated if needed after seeing how the best practices work in practices. The best practices are focusing on implementation projects. The company also has service desk projects where Jira is being used, but those type of projects will be covered separately.

4 User experience survey

To gather the user experiences, Microsoft Forms was used. It was selected as the tool, as it was a previously known tool for the author as well as the respondents. As the comments needed to be anonymous, universal link was shared to all the resources in the business area. In total the survey was sent to 30+ people. The survey was open from Thursday 25th of May to Friday 2nd of June. The survey had in total four questions, three with ready-made answer options and the last one with open text answer:

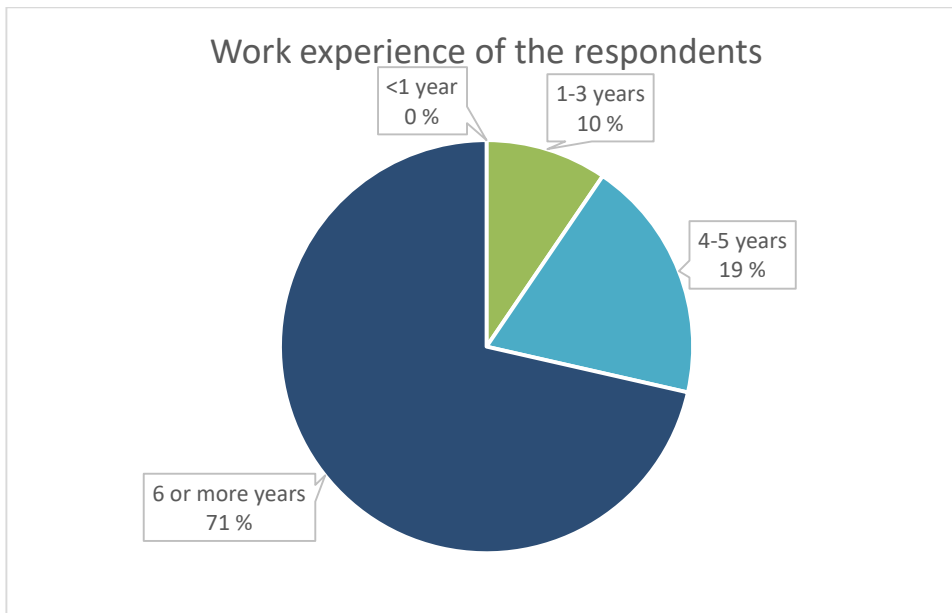
1. How long have you worked in the IT/system development industry? Include experience outside of current employer.
2. In what role are you currently working in?
3. Have you used Jira, either in a project and/or as a service desk tool?
4. What kind of experience of using Jira do you have? What has been good, what could have been improved?

This chapter will analyze the survey results to understand the current situation with using Jira in client company's business area. First subchapter will dive into the respondents and the second subchapter will look at the open text answer received from the survey.

4.1 User experience – background

In total there were 21 responses to the survey and all the respondents have used Jira some way or another. As *Figure 2 Maturity of the respondents* describes, the maturity of the respondents is high, over 70 % of the respondents had more than six years of experience in working IT/system development industry. All the respondents also had more than one year of working experience.

Figure 2 Maturity of the respondents



Highest number of responses, with one third of responses (7 out of 21) came from Project Management/Admin roles. Developers and specialists were the second highest role with only one less response. *The Figure 3 Respondent roles* shows how many responses were received per role. The left axis shows the number of responses per role and right axis shows percentage of the role from all responses.

Figure 3 Respondent roles

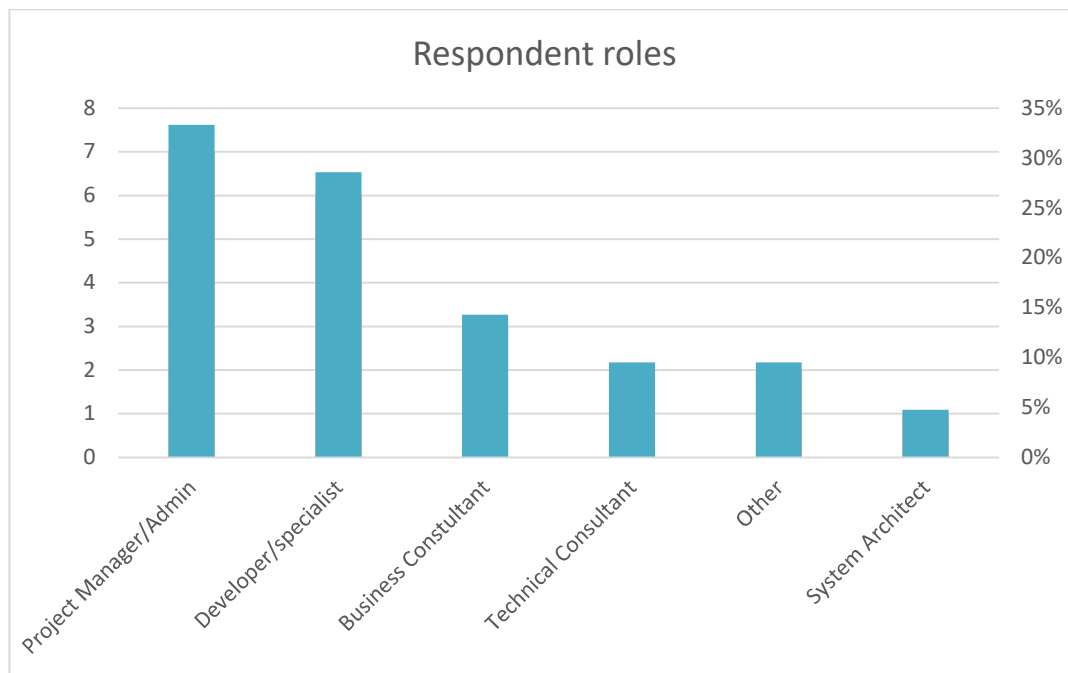
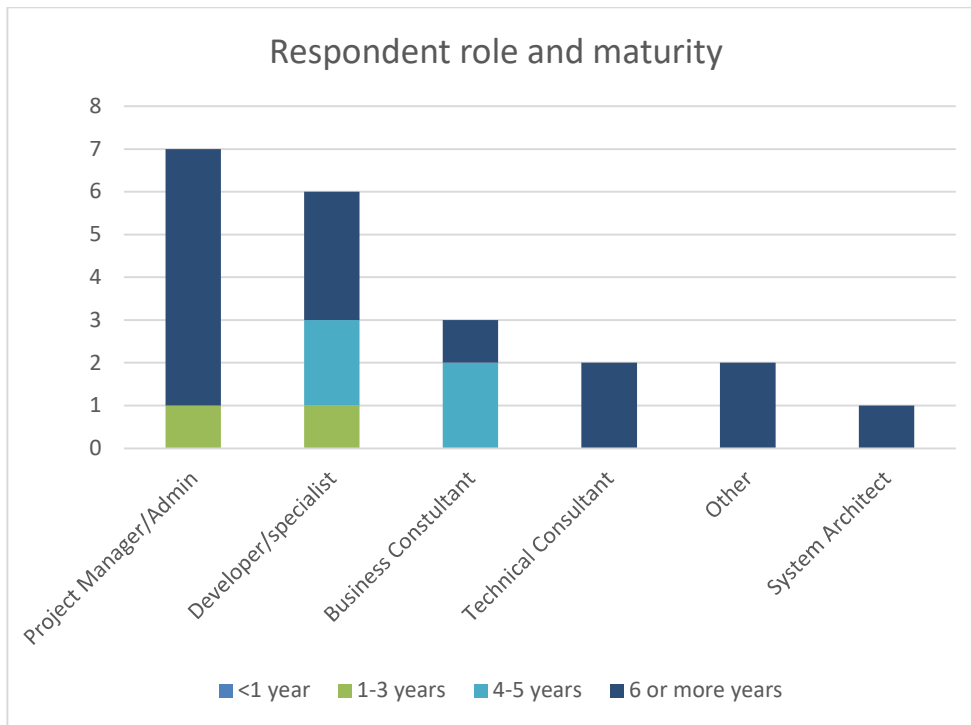


Figure 4 Respondent role and maturity shows the maturity per role. As it was already clearly seen in the *Figure 2 Maturity of the respondents*, it can also be seen here that the maturity of the respondents is high. Only two role groups have resources from 1–3-year category and 4–5-year category.

Figure 4 Respondent role and maturity



The two respondents in the other category described their role to be Business management and Tech Lead/Technical PM, System Architect.

4.2 Open text answers

From the open text answers, it can be clearly seen, that the respondents are happy to use Jira, but are missing the guidelines and best practices for using Jira. Some have mentioned that they would need more training to use Jira the most efficient way. All open text answers are attached as Annex 2 and this chapter will analyze some of the responses.

In *Quotation 1* it is suggested that when using Jira in different kinds of projects, it depends on the project and the tasks for it to be able to use Jira efficiently. The *Quotation 1* sums out the biggest issues in using Jira in the client company. The differences between projects are making it hard to use Jira the same way. The best practices should give guidelines on how to work with Jira in different kinds of projects and not only give strict instructions which needs to be followed in every project.

Quotation 1 Open text comment A

“Used in 2 projects as well as ServiceDesk. In projects it is a good tool but not always easy to understand how to frame or scope the work to be done within an epic, story, task, subtasks etc. We try to work sprint based and use stories as amount of work. this works better and better but there are differences in a pure R&D IT project and a SCADA/MES project which makes it hard sometimes to use Jira in best way. We also try to use it as a way to track the time spent in project and only reference the issue id (e.g., HSL-123) in ERP system and keep all info and documentation in Jira. I think this is the way to go forward.”

From Open text comment B *Quotation 2-6* we can interpret that the respondents are content with using Jira as project management tool, but the users would require more training and the best practices to use Jira efficiently. From *Quotation 2*, it can be interpreting that one of the issues comes when there are multiple resources in one project, and they are updating Jira differently, which creates an issue with following the project.

Quotation 2 Open text comment B

“I think tool itself works fine and it's easy to communicate with people using it. However, I think that the processes around using Jira could be better, I think it's not clear to all exactly how we should update e.g. statuses and ticket owners”

Quotation 3 Open text comment C

“Workflow of working with tickets in sprints is good, you get presented with the most relevant information in an quite good intuitive way. What I think is not that good is how changing of the hours affect the sprint or overall project, you don't get feedback on how the change of tickets in a sprint affects the overall planning of the project in Jira. I'm also missing a good overview of the tickets all-in-all (the Backlog and Roadmap pages are not displaying the whole in an easy way).”

Quotation 4 compares the differences between the client company's Jira and customer Jira. This is a good example of why the best practices are needed, so that all the business area's projects will have the same setup and configuration in the future. The goal for having the same setup for all the projects is to make it easier to create the project and work with it.

Quotation 4 Open text comment D

“I have been using some customer Jiras as developer and project manager, and Roima Jira as team lead/project manager. To my experience Jira is the best tool out there for managing tasks in various types of projects and project methodologies. My experience is that the difficulty is in selecting what kind of setup/configuration should be used in the types of projects that client company or business area is delivering. Business area best practices are needed so that the tool would be used the same way in different projects.”

Quotation 5 Open text comment E

“Jira that The client company uses feels too complicated, or there are choices that the meaning of those have not yet become familiar. Maybe because of that, I am a basic user and not admin/project manager. My first experience of Jira was with XX project that was maintained by customer and it feels easier. Could be that I have just gotten familiar with that version and all the others feel strange.”

This comment was translated from Finnish by the thesis author.

Quotation 7 as well as **Error! Not a valid bookmark self-reference.** notes a problem with time management in the Jira Issues. This is one clear topic that needs to be taken into consideration when creating the best practices. The respondent with **Quotation 7Error! Reference source not found.** sees also a problem with issue overview in Jira and for that as well, the best practices will try to find a solution.

Quotation 6 Open text comment F

"In XX project we started to Use Jira to follow up the time. I, as a PM, had trouble figuring out how to use the reports. Time management was not self-explanatory, and we struggled to follow up sprints and epics. I am sure I can learn to work with this, but I need some education and training. Also, in Lessons Learned, we in the team discussed that we should redo the entire time plan from the original offer to the sprint plan. This is not unnormal, but not in practice in client company right now."

Quotation 7 Open text comment G

"Workflow of working with tickets in sprints is good, you get presented with the most relevant information in an quite good intuitive way. What I think is not that good is how changing of the hours affect the sprint or overall project, you don't get feedback on how the change of tickets in a sprint affects the overall planning of the project in Jira. I'm also missing a good overview of the tickets all-in-all (the Backlog and Roadmap pages are not displaying the whole in an easy way)."

4.3 Conclusions

By looking at the analysis of the responses of the survey and analyzing the comments, it can be clearly seen that even with long experience in the field of IT and system development the tools used in projects are not put in their best usage without having a clear, unified way of working with them. From the comments it can also be concluded that Jira is highly valued tool but using it in different ways in different projects is not efficient. As one comment noted, the client company's Jira should be used instead of customer Jira, to have full control over the configurations of Jira and to be able to follow the best practices created for the business area.

The adaptation of the best practices could be difficult, as there are different kind of users and not all will agree on the created best practices. The best practices will be defined more when they are put to test in real life project. The best practices will be gone through with the whole business area employees and the material will be shared with the users. It is important for the project manager in particular to follow the instructions and make sure the rest of the team will follow them as well.

Few problems, for example time management and the lack of instructions, can be easily identified from the comments. The time management is one of the already before identified topic that need

to be addressed with the best practices to help with the usage of Jira in the business area. In addition to time management, the best practices will make the decision on what issue types and how they are going to be used in the client company's business area. A standard workflow and dashboard will also be created, which can be copied from project to project to help the project manager with the project setup.

5 Best practices for using Jira in client company

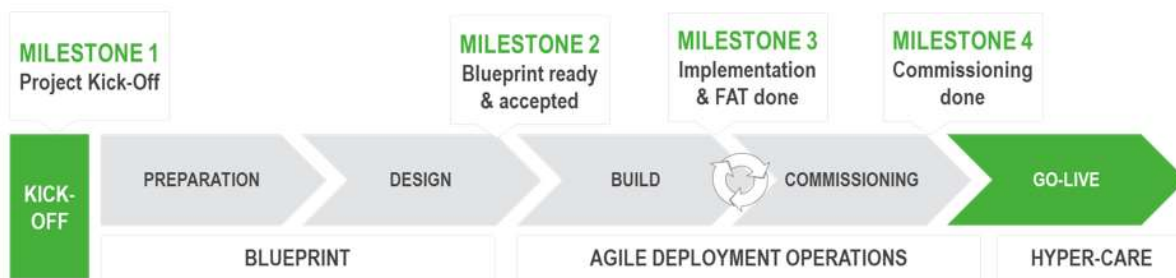
In this chapter the best practices are created by using the same structure as was in chapter 2, the theory of Jira. The best practices of using Jira will be based on the theory of Jira that is covered in chapter 2 and the open text responses from chapter 4, but the best practices will also take the Company's current processes into consideration. The best practices will focus on the implementation projects and cover the service desk side only when they overlap.

5.1 Using Jira

Currently Jira is used in most of the projects in client company's business area. It is used as an implementation and service desk tool, but each project has been created more on the need basis and there have not yet been any guidelines on how to use Jira. The subchapters of this chapter will describe the ways of configuring Jira for projects, boards, issues and issue types, and workflows. The best practices are written here in text format and the instructions for the users are attached as Annex 3. The best practices will give instructions on how to request a new project from the company's Jira admins and how to create new board, what issues to be used, how to configure the needed reports and automation and what kind of workflow will be used.

The projects in client company's business area vary from small projects that last few months to projects that lasts multiple years and cover multiple sites in different countries. The *Figure 5* describes the Company's best practices on running a project regardless of the size of the project. The best practices combine waterfall and agile project methods where agile is used during build and commissioning phase. Jira is used in agile deployment operations.

Figure 5 Client company's project best practices



Best Practices for MES Project Model

- 1 **Preparation**
Share Information, do training and get the team aligned
- 2 **Design**
Design of Minimum Viable Product for MES, with Site Transformation actions
- 3 **Build**
Building the Customer MES Initial and Full Templates with all the functionality
- 4 **Commissioning**
Commissioning and Site Transformation at each site
Testing and Key User trainings
- 5 **Go-Live & Hyper-Care**
Successful Go-Live with 24/7 Hyper-Care for each site

The chosen business area will use Company-Managed Jira project as a standard. In special cases a project can be opened as a Team-managed project, but to have the same settings and ways of working the Company-Managed project is the one which will be used for customer projects.

5.1.1 Boards

By keeping the project best practices (described in *Figure 5*) in mind when using Jira, client company will use Kanban and Scrum boards in their projects. For smaller projects Kanban board is enough but bigger and longer projects will need both. In the business area, there are two types of projects, type 1 and type 2 that are followed in different ways. The usage of Jira will follow the same categorization, type 1 projects will have both Kanban and Scrum boards, and type 2 projects will have only Kanban board.

The Scrum board will be used when the project uses Agile methodology to be able to utilize sprints. These two boards include all the necessary settings that are needed on a basic implementation project in the business area. Although Jira offers the possibility to create and use own templates for project boards, these two boards are commonly known and follow the Agile project methodology that is described to be the methodology for the projects in the business area (especially in build and commissioning phases of the project).

For issue statuses there will be a standard template that will be used in all the projects. The statuses are described in *Figure 6 Issue statuses*. Statuses run from a list of defined issues that are in design to backlog where the development team changes the issue to status in progress when the issue has been taken into development. When the developer is ready with the issue they will move it to QA (quality assurance) review. After QA review the issue status is changed to customer testing. When customer testing is ready the issue moves to status waiting for release. The final step is when issue is ready and deployed, it is moved to status Done. If QA or customer testing fails, the issue moves to status Failed in test, where the developer moves it to In progress when they start working with the issue again and the rest of the workflow will continue with same path.

Figure 6 Issue statuses



With standard issue statuses between the projects, the resources will be able to move more easily between the projects. They will have the same way of working with each project and there is no longer a need to learn a different way of working with each project. The requirement of updating the issues will be the same for each project and will be the normal way of working on a project. With updated issues the project manager will be able to see the project and issue status and intervene with the issues that seem to be lagging on the timeline.

5.1.2 Issues and issue types

Client company's chosen business area will use Epics as the highest level of issue type and Tasks under the Epic to divide the high-level issue into smaller pieces. The workload estimates of certain Epics Tasks will be summed up in the Epics original estimate field by automation rule. The instructions for creating the rule are described in Annex 3. This will give an overall look on the size and workload of each Epic and the planning of the workload will be more straight forward. As

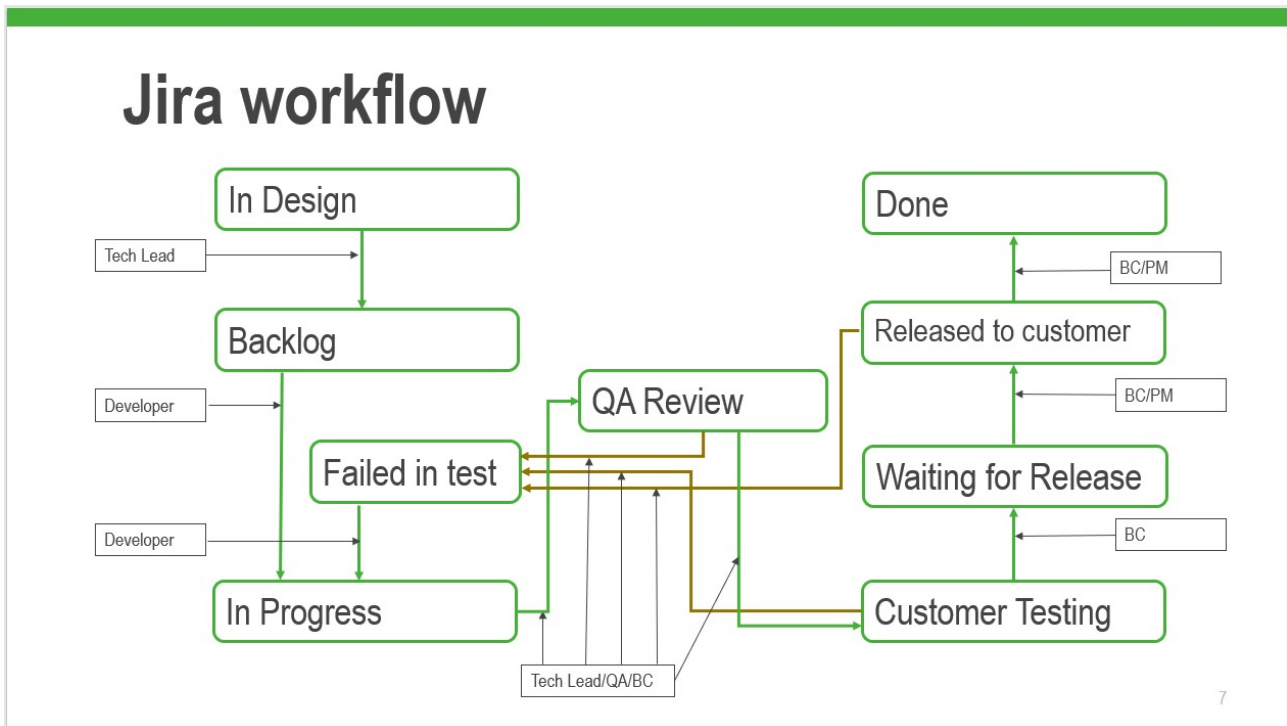
described in theory subchapter 2.2.3. Issues and issue types, Epics are not planned to be used as client company will use them, but it is seen that Stories and Subtasks are not the best way to work with Jira. The Subtask estimation will not be summed in the Story and the overall workload is not seen as clearly as it will be with Epics and Tasks. In the business area's projects one Epic will describe one use case and Tasks below the Epic will break the use case into actual work tasks for the project team.

When opening an Epic, the fields Epic name and summary are needed. These both will need to include the same information, as name is shown in the Tasks as a label and the summary is shown as the Epic identifier. When creating the project Tasks, Fix version and Components need to be filled in to utilize the reports and gadgets that are described in subchapter 5.2.1.

5.1.3 Workflows

Each project will have a standard workflow when the project is created, but that is too simple for the projects the best practices are created for. The workflow is formed from the issue statuses described in subchapter 5.1.2. A separate workflow template will be created by company Jira administrators and will be used for all projects in the business area. The standard workflow and the responsible roles for updating the statuses of the issues is shown in *Figure 7 Jira workflow*.

Figure 7 Jira workflow



5.2 Reports and dashboards

This chapter will describe the best practices for reports and dashboards. Overall goal is to have the same reports and dashboards shown the same way in each project. This will create a unified way of reporting the status of the projects to the team and to higher management level. With standard reports the deviations will be easier to spot out and take needed actions on those.

5.2.1 Reports

For standard reporting from project to another, *Figure 8 Standard reports* describes the reports that every project will use as a standard way of reporting. For project management purposes other reports can also be used on a need basis, but for best practice purposes the reports defined here will be the required reports. The *Figure 8 Standard reports* also describes the need of the report and on what field the report is based.

Figure 8 Standard reports

Filter results	Created vs Resolved chart	Issue statistics (assignee)	Issue statistics (components)	Road map
<ul style="list-style-type: none"> •High-risk items that needs more closer follow up •Report is based on the issue priority 	<ul style="list-style-type: none"> •Work added vs work completed over time •Report is based on the issue statuses and created issues 	<ul style="list-style-type: none"> •Relative workload per person •Report is based on the issue assignee field 	<ul style="list-style-type: none"> •For following tasks for specific component relates issues (eg. UI) •Report is based on the component field 	<ul style="list-style-type: none"> •Progress towards next milestone •Report is based on the fix versions field

Detailed instructions on creating the reports will be provided in the user manual attached as Annex 3.

5.2.2 Dashboards

A template dashboard which contains the gadgets described in subchapter 5.2.1 is created and shared in Jira with all the company users. It will be copied and updated to show data from a specific project. Instructions for updating the gadgets per project will be created and shared with all project managers in the business area (Annex 3). If needed, additional gadgets can be created for a project dashboard, but the template gadgets will be the minimum requirement for each project.

5.3 Atlassian's other applications and Jira integrations

Atlassian has multiple other tools besides of Jira and in addition to their own tools Jira can be integrated with thousands of other applications. This chapter will explain the best practices for integrating Jira in the Client Company's chosen business area.

5.3.1 Confluence

Confluence will not be included in the best practices or used in the client company at this point. There is a risk of confidential information to be shared outside the client company if the Confluence project is opened the wrong way. This could possibly cause liability issues with the client company's customers and Confluence is seen not to be worth the risk.

5.3.2 BitBucket cloud

The client Company uses a few different GIT -based version control tools and one of those is BitBucket. Version control tool is not included in the best practices created for using Jira as it differs too much between projects and systems that are developed. Version control best practices will be part of future development for the client company and the usage of Bitbucket will be analyzed more then. The need for version control best practices has been identified in the company.

5.3.3 Jira integrations

From the list of thousands of other applications that can be linked to Jira, Teams (Microsoft communication app) is decided to be the place where all project documentation will be stored. The issues can also be added as an attachment to meeting invitations and updated through Teams pages. The best practices will include instruction on how to integrate Jira and Teams, to be able to communicate the Jira issues directly from the team's communication tool.

It is also planned and investigated if the Company's self-developed ERP system could be integrated with Jira for automated time tracking. The automated time tracking from Jira to the ERP system would help the development team with logging their hours without needing to manually include the issue keys in the hour bookings, giving the development team more time to focus on the actual development work. The development item is currently on the roadmap of the discrete business area and will not be included in the best practices created during this thesis, but will be included in future, after the development work has been made.

6 Summary

There were five research questions defined in the design phases of the thesis. The first question was answered by conducting the user experience survey and getting feedback from the current and future users. The second question is answered by creating the best practices during the process of the thesis. The created best practices are not written in stone and can change in the future, but for now, the business area has a common way of working with Jira in implementation projects.

The third and fourth questions might not be that easily answered, at least not in short term, but with clear instructions on Jira configurations and issue follow up, the expectation is that project follow up and escalations will become more efficient and by that help lower the costs and life cycle of projects. To answer the fifth and last question, the differences between Jira software and Jira service desk are answered in the theoretical framework section of the thesis. The best practices are directed at using Jira software and the service desk tool is not included in the guidelines but will be covered in separate context.

During writing the thesis I have learned more than expected, firstly about Atlassian as a company and what kind of tools they offer. Secondly, the best ways to use Jira software and how to develop the current usage with add-ons and integrations. And thirdly, the user experience survey responses were very informative on the need for the best practices for the company and the specific business area. Overall, the process of creating the best practices has been helpful for the current role I am working with and will surely bring value in the future as well, not only for me as a project manager but also for the Company and the business area.

After the thesis is published a best practice session will be held for the business area's employees to share the new guidelines. In the future, the best practices will be further developed by new integrations, tools and instructions as seen fit. If requested the ways of using Jira in the Company's other two business areas will be evaluated and the best practices are extended to cover their processes as well.

References

- AgileAlliance. 'What Is Agile?' Accessed 29 May 2023. <https://www.agilealliance.org/agile101/>.
- Akeles. 'What Are the Differences between JIRA Software, JIRA Service Management and JIRA Core'. Accessed 27 May 2023. <https://www.akeles.com/what-are-the-differences-between-jira-software-jira-service-desk-and-jira-core/>.
- Amoeboids. 'A Quickstart Guide to JQL'. Accessed 4 June 2023. <https://amoeboids.com/blog/a-quickstart-guide-to-jira-query-language-jql/>.
- Atlassian. 'Agile Tools for Software Teams', n.d.-a <https://www.atlassian.com/software/jira/agile>.
- . 'Apps for IT Teams'. Accessed 15 June 2023. -b <https://marketplace.atlassian.com/categories/it-teams>.
- . 'Apps for Software Teams'. Accessed 15 June 2023. -c <https://marketplace.atlassian.com/categories/software-teams>.
- . 'Bitbucket Integration'. Accessed 25 June 2023. -d <https://www.atlassian.com/software/jira/bitbucket-integration>.
- . 'Bug Tracking Done Right with Jira Software'. Accessed 28 May 2023. -e <https://www.atlassian.com/software/jira/features/bug-tracking>.
- . 'Code & CI/CD, Built for Teams Using Jira'. Accessed 25 June 2023. -f <https://bitbucket.org/product/>.
- . 'Confluence Basics'. Accessed 4 June 2023. -g <https://www.atlassian.com/software/confluence/resources/guides/get-started/overview#key-terms>.
- . 'Create an Issue and a Subtask'. Accessed 24 June 2023. -h <https://support.atlassian.com/jira-software-cloud/docs/create-an-issue-and-a-sub-task/>.
- . 'Design Tools to Collaborate Seamlessly'. Accessed 15 June 2023. -i <https://marketplace.atlassian.com/categories/design-tools>.
- . 'Get the Best Project Management Apps'. Accessed 15 June 2023. -j <https://marketplace.atlassian.com/categories/project-management>.
- . 'Integrate Jira Cloud and Slack'. Accessed 27 May 2023. -k <https://support.atlassian.com/jira-software-cloud/docs/use-jira-cloud-for-slack/>.
- . 'Integrations with Jira Software Overview'. Accessed 15 June 2023. -l <https://www.atlassian.com/software/jira/guides/integrations/overview#what-is-atlassian-marketplace>.
- . 'Introduction to Jira Family'. Accessed 24 June 2023. -m <https://www.atlassian.com/software/jira/guides/more/jira-family#what-is-the-jira-family>.
- . 'Jira Software Boards Introduction', n.d. -o <https://www.atlassian.com/software/jira/guides/boards/overview#types-of-boards>.

- . 'Jira Software Project Overview'. Accessed 28 May 2023. -p <https://www.atlassian.com/software/jira/guides/projects/overview#what-is-a-jira-project>.
- . 'JQL: The Most Flexible Way to Search Jira'. Accessed 4 June 2023. -q <https://www.atlassian.com/blog/jira-software/jql-the-most-flexible-way-to-search-jira-14>.
- . 'Learn about Jira Software Reports & Dashboards'. Accessed 4 June 2023. -r <https://www.atlassian.com/software/jira/guides/reports-dashboards/overview#what-are-reports>.
- . 'Reporting in Jira'. Accessed 4 June 2023. -s <https://confluence.atlassian.com/jirakb/reporting-in-jira-461504615.html>.
- . 'Reporting in Jira Software'. Accessed 4 June 2023. -t <https://www.atlassian.com/software/jira/features/reports>.
- . 'Software'. Accessed 25 June 2023. -u <https://www.atlassian.com/software>.
- . 'Stories, Epics, and Initiatives'. Accessed 28 May 2023. -v <https://www.atlassian.com/agile/project-management/epics-stories-themes>.
- . 'Trello'. Accessed 15 June 2023. -w <https://trello.com/tour>.
- . 'Using Confluence and Jira Service Management Together'. Accessed 12 June 2023. -x <https://www.atlassian.com/software/confluence/resources/guides/extend-functionality/confluence-jsm#overview>.
- . 'Using Confluence Cloud and Jira Software Together'. Accessed 12 June 2023. -y <https://www.atlassian.com/software/confluence/resources/guides/extend-functionality/confluence-jira#integrate-confluence-jira>.
- . 'Using the Help Center'. Accessed 12 June 2023. -z <https://confluence.atlassian.com/servicemanagementserver/using-the-help-center-939926561.html>.
- . 'What Are Board Filters?' Accessed 4 June 2023. -aa <https://support.atlassian.com/jira-software-cloud/docs/configure-filters/>.
- . 'What Are Issue Types?' Accessed 28 May 2023. -ab <https://support.atlassian.com/jira-cloud-administration/docs/what-are-issue-types/>.
- . 'What Are Jira Workflows?' Accessed 29 May 2023. -ac <https://support.atlassian.com/jira-work-management/docs/what-are-jira-workflows/>.
- . 'What Is a Board in Jira Software?' Accessed 29 May 2023. -ad <https://support.atlassian.com/jira-software-cloud/docs/what-is-a-jira-software-board/>.
- . 'What Is a Jira Dashboard?' Accessed 4 June 2023. -ae <https://support.atlassian.com/jira-software-cloud/docs/what-is-a-jira-dashboard/>.
- . 'What Is an Issue?' Accessed 28 May 2023. -af <https://support.atlassian.com/jira-software-cloud/docs/what-is-an-issue/>.
- . 'What Is Opsgenie?' Accessed 25 June 2023. -ag <https://www.atlassian.com/software/opsgenie/what-is-opsgenie>.

AWS. 'What Is SQL (Structured Query Language)?' Accessed 4 June 2023.

<https://aws.amazon.com/what-is/sql/>.

Bitbucket. 'A Brief Overview of Bitbucket'. Accessed 27 May 2023. -a

<https://bitbucket.org/product/guides/getting-started/overview#a-brief-overview-of-bitbucket>.

———. 'Bitbucket: What Is Bitbucket?', 7 March 2018. -b

<https://confluence.atlassian.com/confeval/development-tools-evaluator-resources/bitbucket/bitbucket-what-is-bitbucket>.

Drozd, Kelly. 'Jira Epics Tutorial'. Accessed 17 June 2023.

<https://www.atlassian.com/agile/tutorials/epics>.

GitLab. 'What Is DevOps?' Accessed 25 June 2023. <https://about.gitlab.com/topics/devops/>.

Idalko. 'How to Use and Create Epics in Jira: The Ultimate 2023 Guide'. Accessed 17 June 2023.

<https://www.idalko.com/epics-in-jira-2023-guide/>.

NobleDesktop. 'What Is Git and Why Should You Use It?', 19 December 2022.

<https://www.nobledesktop.com/learn/git/what-is-git>.

Rehkopf, Max. 'Epics'. Accessed 28 May 2023. -a <https://www.atlassian.com/agile/project-management/epics>.

———. 'User Stories with Examples and a Template'. Accessed 28 May 2023. -b

<https://www.atlassian.com/agile/project-management/user-stories>.

Saleh, Sarah. 'A Brief History of Atlassian and Jira', 21 December 2021. <https://jexo.io/blog/atlassian-jira-history/>.

Simplilearn. 'What Is a Project in Project Management: Life Cycle, and More', 16 May 2023.

<https://www.simplilearn.com/what-is-a-project-article>.

Slack. 'An Easier Way to Use JIRA with Slack', 28 March 2017. <https://slack.com/blog/productivity/an-easier-way-to-use-jira-with-slack>.

Synopsys. 'CI CD'. Accessed 25 June 2023. <https://www.synopsys.com/glossary/what-is-cicd.html>.

Annex 1: Material management plan

Permission to conduct the survey was sought from the selected business area's country director in advance. The respondents were informed about the survey by email and in a meeting and all the respondents have chosen to respond to the survey. The survey was conducted by Microsoft Forms and the open text answers are processed to anonymize the commissioner and possible customer names. The background questions are processed to get understanding of the seniority and roles of the respondents.

All the material collected and created during the thesis will be stored 1 year after the approval of the thesis, after which those are destroyed. All responses are anonym, and no response can be linked to a certain respondent. No personal information was collected during the thesis. The data was stored in the writer's Google Drive during the thesis process and only the writer had access to the data.

All materials and the results of the thesis are owned by the commissioner.

Annex 2 User experience responses

This annex holds all the open text comments. Some of the comments have been modified to secure the confidentiality of either the Company, the respondent, or the customer.

"Splitting projects into manageable tasks/tickets. Allocating them to a group of developers. And following the progress of the project in completion/time/problems. Giving (programming)tasks time-estimates usually requires some experience. Time to "make it work on my computer" is different than time to "I can prove it works and it is well documented". Also, interaction with previous and other systems is critical when introducing Jira. Version control, document management, billing, what have you, it is paramount to figure out which other systems to keep and what do lose." Developer/Specialist

"Used in 2 projects ... as well as service desk. In projects it is a good tool but not always easy to understand how to frame or scope the work to be done within an epic, story, task, subtasks etc. We try to work sprint based and use stories as amount of work. this works better and better but there are differences in a pure R&D IT project and a SCADA/MES project which makes it hard sometimes to use Jira in best way. We also try to use it as a way to track the time spend in project and only reference the issue id (e.g. HSL-123) in ERP system and keep all info and documentation in Jira. I think this is the way to go forward. I am happy to be involved with feedback and ideas if needed" Other

"Start up training and set up videos. Trainings about how to effectively use JIRA and what features do we have in JIRA" Other

"Briefly, But I heard from others that they are very satisfied." Project Manager/Admin

"I think tool itself works fine and it's easy to communicate with people using it. However I think that the processes around using Jira could be better, I think it's not clear to all exactly how we should update e.g. statuses and ticket owners" Business Consultant

"I worked with jira as part of a project that used agile framework. Jira paired with the agile framework worked well in my opinion. It was easy to use and helped productivity and made planning of your day/week quite easy and simple. The kanban board functionality lets you see what work you have left as well as the other people in the project. This made it simple to follow the state of the project and also going through any possible blockers in daily meetings made progression smooth." Developer/Specialist

"Overall Jira seems to be quite intuitive and easy to use. Jira also has a comprehensive set of configuration tools. But cloud-version has some significant flaws: customer support portal is terrible in multi-project/customer environment and configurability is very limited." Project Manager/Admin

"Good: Very easy to start using, very clear views and visual tool for persons, that are not JIRA users. Improvements: some filters or reports bit tricky to create. And additional tools/plugin-ins are payable." Project Manager/Admin

"I have worked with multiple ticketing system, but none as good as JIRA. Its easy to locate a ticket, very straight forward to raise the ticket. clear status is displayed of the ticket, add comments or attachments and a simple UI to follow" Technical Consultant

"Seems ok when used correctly, heavy when used wrong. Setting the level of detail on the tickets seems essential. Often tickets had a lot of "sub issues", where a ticket is only accepted after all "sub" tickets are accepted. That makes the Kanban board bloated with tickets that were not moving. The statuses also do not always make sense in a project but require discussion which statuses are on us as developers or testers or on the customer. The tool is also missing a bit on the project KPI side, the custom dashboards that can be created are a bit lacking. That caused extra tracking

excels which is double work. The process of creating and managing the tickets is ok, and the interface is easy to learn. A Kanban is not something unique to Jira, though.” Business Consultant

“Workflow of working with tickets in sprints is good, you get presented with the most relevant information in a quite good intuitive way. What I think is not that good is how changing of the hours affect the sprint or overall project, you don't get feedback on how the change of tickets in a sprint affects the overall planning of the project in Jira. I'm also missing a good overview of the tickets all-in-all (the Backlog and Roadmap pages are not displaying the whole in an easy way).” Technical Consultant

*“Project managing, ticketing. Jira is very clear, intuitive and easy to use. Don't have any improvement ideas”
Developer/specialist*

“I have been using some customer Jiras as developer and project manager, and client company Jira as team lead/project manager. To my experience Jira is the best tool out there for managing tasks in various types of projects and project methodologies. My experience is that the difficulty is in selecting what kind of setup/configuration should be used in the types of projects that client company or XXX is delivering. XXX best practices are needed so that the tool would be used the same way in different projects.” Project Manager/Admin

“Jira jota Yritys X käyttää tuntuu liian monimutkaiselta, tai siellä paljon valintoja, joiden tarkoitus ei ole tullut tutuksi. Ehkä sen takia että olen vain peruskäyttäjä, enkä ylläpitäjä/projektinjohtaja. Ensimmäinen kokemukseni Jirasta oli ... projektista (asiakkaan ylläpitämä) ja se tuntuu selkeämmältä. Ehkä ehdin tottua siihen niin muut versiot tuntuu vieraalta.” Developer/Specialist

“This is a really good tool for general project management in an IT project. For actual development with CI/CD, I'd probably prefer Azure DevOps. In future projects, client company's Jira should be the standard and then customers should have restricted access to e.g. Roadmap to see the progress.” Project Manager/Admin

“In ... project we started to Use Jira to follow up the time. I, as a PM, had troubles figuring out how to use the reports. Time management was not self explanatory and we struggled to follow up sprints and epics. I am sure I can learn to work with this, but I need some education and training. Also, in Lessons Learned, we in the team discussed that we should redo the entire timeplan from the original offer to the sprint plan. This is not unnormal, but not in practice in client company ... right now.” Project Manager/Admin

“Correctly configured, it's very effective and easy” Developer/Specialist

*“I have actively used Jira in one project only. I like that we can integrate it to git which automates status updates for tasks. I found trying to find an old task difficult after a sprint was closed, in case I needed to refer to something.”
Developer/Specialist*

“I have used Jira in a big scale IT implementation project as the main issue tracking system. Jira provides easily configurable filters for personal views and a good and easy to use kanban board. Since Jira is a web based software, it can be used easily from anywhere and if required, also by phone. Additionally, in Jira it was easy to get reports and visualizations for the progress and have them combined in one view. Also on the individual tickets it was easy to track the ticket history and set the priority and assign it forward. What could have been improved was the labeling of the tickets, because there seemed to be limited amount of extra fields that can be utilized, but that could have also been due to a poor set up for the system. I don't have much experience from other issue tracking systems besides Jira and Service Now, but I'd pick Jira any day due to the user experience and ease of use it has.” Business Consultant

“Good: boards, dashboards, project templates. Bad: timestamps's format problems, visuality, data format when exporting data” Project Manager/Admin

Annex 3 Instructions for using Jira in the chosen business area

BUSINESS AREA

Best practices for using Jira

June 2023

Internal

Background

- The best practices give instructions on
 - opening a project
 - what boards are needed
 - what issue types to use
 - how to add a standard workflow
 - what standard reports and gadgets will be used and how to create them
 - how to create a standard dashboard
 - how to create automation e.g., for time estimation in an Epic
 - Jira and Teams integration
- The best practices are the minimum requirement for using Jira in an implementation project
 - In addition, extra reports and dashboards, automation etc. can be created on a need basis

Opening a Jira project

- IT will open the Jira project and the following information needs to be included:
 - Project name:
 - Project type: Jira software (implementation projects) or Service desk
 - Board: Kanban
 - Project Admin: Project manager/requester
 - Workflow: To be copied from test project
- The request of opening a project should be sent to with the above information

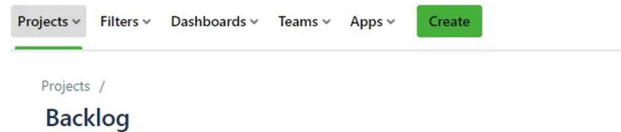
3

Jira boards

- After the project is ready in Jira, the admin needs to add another board to the project. The minimum requirement is that the project includes Kanban (created by default) and Scrum boards
- The Scrum board is needed to be able to use sprints in the project
 - Type 2 projects do not need the Scrum board, but in can be added if seen needed
- Detailed instructions on creating a new board will be found in the end of this presentation

4

Jira Issue types



- In business area's implementation projects, the used Issue types will be Epic and Task
 - One Epic will represent one use case and Tasks opened under a certain Epic will represent the development work needed for that certain use case
- Both types of Issues can be opened in several ways, but the simplest way is to click the green Create button on top of the Jira page.
- When opening an Epic, make sure the Epic Name and Summary has the same information
 - Epic name will be shown in the Tasks created to that Epic and the Summary will be shown as the identifier for the Epic
- When opening a Task, fix version and components field needs to be updated
 - The fix version should be agreed by project, but the main rule is that it should describe the milestones or release versions of the project
 - The components field is used to identify on what part of the use case that specific task is for, e.g., UI etc.

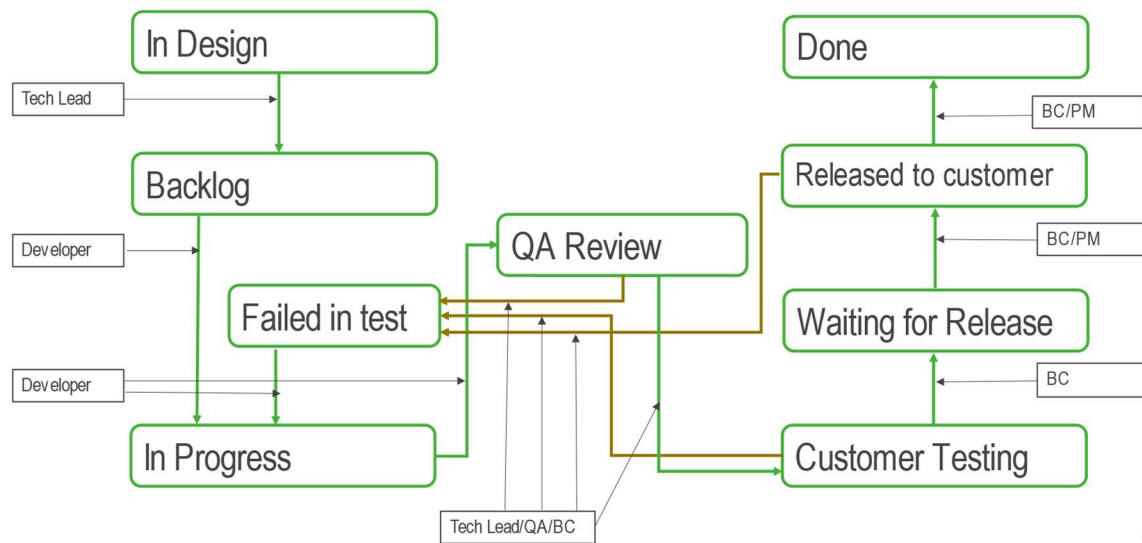
5

Jira workflow and issue statuses

- The same workflow (pictured in the next slide) will be used through out the PB/ , lementation projects
 - All the Issues will start from the In Design status and will be moved by the Tech Lead to Backlog when detailed design is done
 - When the development team member starts to work on the issue, they will move the Issue to status In Progress
 - After the Issue is ready from development, the developer will move it to QA review for QA testing
 - QA can move the Issue to either customer testing or back to development if any problems have been found
 - After customer testing is approved BC is responsible for updating the status to Waiting for Release
 - When the issue is released to customer production environment, the status is changed to Released to customer.
 - After the issue has been tested and approved by the customer, the Issue is moved by BC (or PM) to Done status
 - If QA or customer finds from their tests that something is not working with the issue, the issue is moved to status failed in test. Developer then picks it up from there and starts the process again.
- Issue status are tied to the workflow and the Kanban board needs to be edited to match the workflow statuses
 - The Kanban board will update automatically when the workflow is copied to the project by IT admin

6

Jira workflow



7

Jira reports and gadgets

- will use 5 standard reports in each project
- The project overall status will be easily seen with these reports and a standard way of reporting the projects will make it easier for the readers to see where the bottlenecks are

Filter results	Created vs Resolved chart	Issue statistics (assignee)	Issue statistics (components)	Road map
<ul style="list-style-type: none"> • High-risk items that needs more closer follow up • Report is based on the issue priority 	<ul style="list-style-type: none"> • Work added vs work completed over time • Report is based on the issue statuses and created issues 	<ul style="list-style-type: none"> • Relative workload per person • Report is based on the issue assignee field 	<ul style="list-style-type: none"> • For following tasks for specific component relates issues (eg. UI) • Report is based on the component field 	<ul style="list-style-type: none"> • Progress towards next milestone • Report is based on the fix versions field

- Additional reports can be created per project if needed

8

Jira dashboard

- The reports described in the previous slide can be added to a dashboard to easily show all reports in one page
- A template Dashboard has been created Jira, which holds the needed gadgets
 - The gadgets will need to be configured per project, detailed instructions on configuring the gadgets can be found in the end of this presentation
- The Dashboard layout can be modified, and the changes will only affect on the current Dashboard
 - Options range from 1-3 columns

9

Jira time estimation - automation

- For Jira to be able to sum the workload estimation from Tasks to the parent Epic, an automation needs to be created
- Detailed instructions for creating the automation can be found in the end of this presentation

10

Jira and Teams integration

- Jira and Teams can be integrated, and the benefit is that the Jira Issues can be updated directly from Teams
- After the integrations, the Issues can be added for example to calendar invitations and updated through

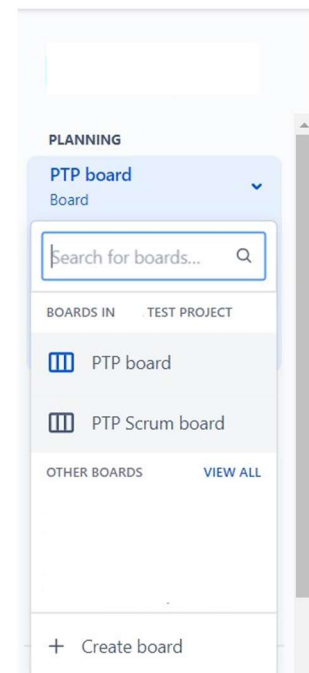
11

Detailed instructions

For boards, gadgets, dashboards, estimation automation and Teams integration

Creating a new board

- A new board can be created from the drop-down menu in the left navigation pane, under the projects name



Creating a new board

- After clicking the + Create board, Jira will ask to choose between Scrum and Kanban boards. Choose Create a Scrum board
- Then click next with the default selection “Board from and existing project



Creating a new board

- Give the board a describing name, from the drop-down menus choose project and location (should be the same)
- Then click Create board

15

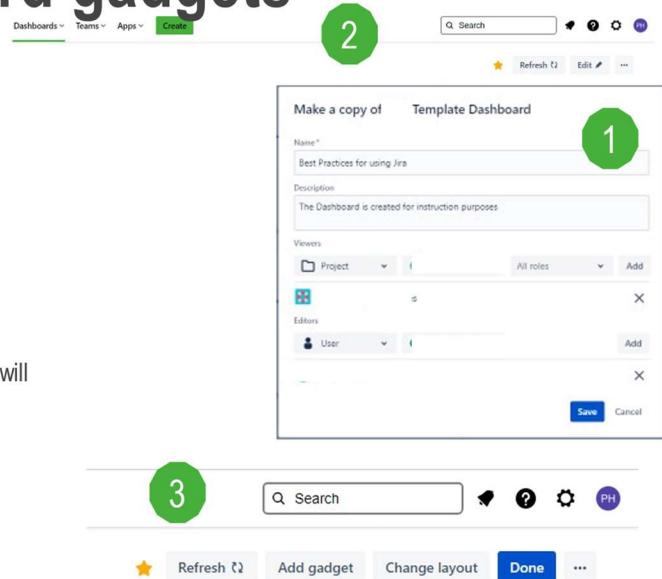
Creating dashboard gadgets

- From the top navigation bar choose option View all dashboards
- Search or scroll to find the Template Dashboard
- Click to open the Template Dashboard
- On the Template Dashboard, click on the three dots in the right upper corner and click Copy

16

Creating dashboard gadgets

- Give the Dashboard a describing name, description and choose viewers (select project) and editors (select user and add the project owner)
- Click Save and the created Dashboard opens with the standard gadgets
- Click Edit from the top right to configure the gadgets
 - After all the gadgets are configured (instructions in coming slides), click Done from the top right corner and the gadgets will automatically update
- Click the star next to refresh button to save the dashboard to your favorites
- At this point, the layout of the Dashboard can be modified, if preferred, from the top right corner



Jira Filter Results -gadget

- For configuring the Created vs. Resolved Chart gadget select first the project the Dashboard is created for and check that the following fields are as
 - Saved Filter: Create a new filter and search it here (see instructions in next slide)
 - Number of Results: 10
 - Columns to display: Issue Type, Key, Summary, Priority, Assignee (not default)
 - Update every 15 minutes unticked
- Click Save



🕒 43 minutes ago

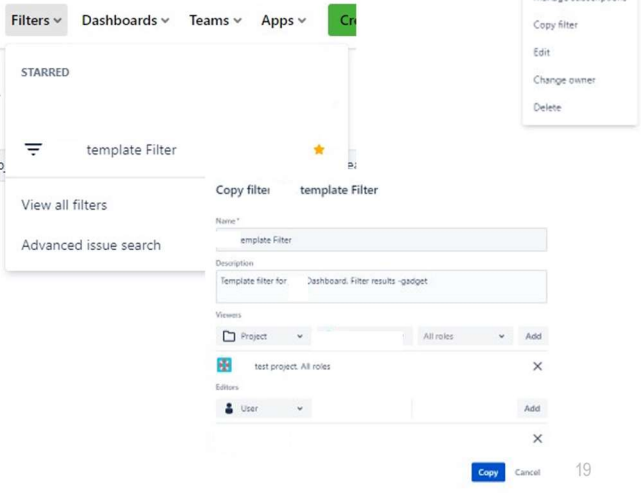
18

Filters **Create a new Filter for Filter Result -gadget** [Create filter](#)

templa X Owner Project Group Clear filters

Name* Owner: Pira Hiltunen Viewers: My organization Editors: All Starred by: 1 person

- Select Filters from the top navigation panel and click View all filters
- Search for the template Filter, click the three dots from the right side of the row and select copy filter
- Give the filter a describing name, description and again choose the viewers (project) and editors (user)
- Click Copy



The screenshot shows the 'Copy filter' dialog box for a filter named 'template Filter'. The dialog includes fields for 'Name*' (template Filter), 'Description' (Template filter for Dashboard, Filter results -gadget), 'Viewers' (Project: test project, All roles), and 'Editors' (User). A 'Copy' button is visible at the bottom right of the dialog.

Create a new Filter for Filter Result -gadget

template Filter — Edited Save Details

project = "PBA test project" AND fixVersion = earliestUnreleasedVersion() AND originalEstimate > 4h Search Switch to basic ID

- Go to the created Filter and update the project to match your project
- Click Search and then Save
- Now the Filter is ready, and you can go back to updating the Filter Result gadget

Jira Created vs. Resolved chart-gadget

- For configuring the Created vs. Resolved Chart gadget select first the project the Dashboard is created for and check that the following fields are as
 - Period: Daily
 - Days Previously: 30
 - Collection Operation: Cumulative
 - Display the Trend of Unresolved: No
 - Display Versions: None
 - Update every 15 minutes ticked
- Click Save

Created vs. Resolved Chart

Project or Saved Filter: **test project**

Search

Project or saved filter to use as the basis for the graph. Advanced Search

Period: **Daily**

The length of periods represented on the graph.

Days Previously: **30**

Days in the past to collect data for the selected period.

Collection Operation: **Cumulative**

Progressively add totals (1, 2, 3) or show individual values (1, 1, 1).

Display the Trend of Unresolved: **No**

Show the number of unresolved issues over time in a subplot.

Display Versions: **None**

Show when versions were released on the chart.

Auto refresh

Update every 15 minutes

Save

23 minutes ago

21

Jira Issue Statistics –gadget (assignee)

- For configuring the first Issue Statistics gadget select first the project the Dashboard is created for and check that the following fields are as
 - To Statistic Type: Assignee
 - To Sort By: Total
 - Number of Results: 15
 - Sort Direction: Descending
 - Show Resolved Issue Statistics: No
 - Update every 15 minutes ticked
- Click Save

Issue Statistics

Project or Saved Filter: **test project**

Search

Project or saved filter to use as the basis for the graph. Advanced Search

Statistic Type: **Assignee**

Select which type of statistic to display for the filter.

Sort By: **Total**

Sort by row total or natural field order.

Number of Results: **15**

Number of results to display (maximum of 50).

Sort Direction: **Descending**

Show Resolved Issue Statistics: **No**

Include resolved issues in the set of issues from which statistics are calculated.

Auto refresh

Update every 15 minutes

Save

13 minutes ago

22

Jira Issue Statistics –gadget (components)

- For configuring the second Issue Statistics gadget select first the project the Dashboard is created for and check that the following fields are as
 - To Statistic Type: Components
 - To Sort By: Total
 - Number of Results: 10
 - Sort Direction: Descending
 - Show Resolved Issue Statistics: No
 - Update every 15 minutes ticked
- Click Save

The screenshot shows the configuration for the 'Issue Statistics' gadget. The 'Project or Saved Filter' is set to 'test project'. The 'Statistic Type' is 'Components'. The 'Sort By' is 'Total'. The 'Sort Direction' is 'Descending'. The 'Number of Results' is '10'. The 'Show Resolved Issue Statistics' is 'No'. The 'Auto refresh' checkbox is checked, and the update interval is 'Update every 15 minutes'. A 'Save' button is visible at the bottom.

23

Jira Road Map -gadget

- For configuring the Road Map gadget, the following selections are to be made:
 - Projects and Categories: Select the project you are creating the gadget for
 - Days: Select the timeline in days on what the gadget will follow, I will use 60 days
 - Number of Results: 10
 - Refresh Interval: Every 2 Hours
- After the selections, click Save
- If there are no Issues or Fix version field updated, the gadget will be empty at this point

The screenshot shows the configuration for the 'Jira Road Map' gadget. The 'Projects and Categories' dropdown is set to 'test project'. The 'Days' field is set to '30'. The 'Number of Results' is '10'. The 'Refresh Interval' is set to 'Every 2 Hours'. A 'Save' button is visible at the bottom.

8 minutes ago

24

Jira Epic estimation automation

Projects / test project / Project settings / Automation

Automation

Rules Audit log Templates

Create rule ...

1

- To create the automation that sums the Task estimations to Epic level go to Project settings, Automations and select Create rule
- From the menu page select Field value changed

New trigger

Triggers start the execution of a rule. Triggers can listen for events or be scheduled to run.

All components Start typing to filter components

Recommended

- Vulnerability found**
Rule executes when a new security vulnerability is found. **NEW**
- Field value changed**
Rule is run when an issue's field value changes. **POPULAR**
- Issue commented**
Rule is run when a new comment is added to an issue. **POPULAR**
- Issue created**
Rule is run when an issue is created. **POPULAR**

Issue triggers

- Field value changed**
Rule is run when an issue's field value changes.
- Issue assigned**
Rule is run when an issue is assigned to a user.

2

25

Jira Epic estimation automation

- Make the following selections:
 - Fields to monitor for changes: Time Tracking, Log Work
 - Change type: Any changes to the field value
 - For: Create issue, Edit issue
- Click Save

Field value changed

This rule will trigger when the value of the fields selected below changes. Learn more.

Fields to monitor for changes*

Time tracking x Log Work x

Change type*

Any changes to the field value

For

Edit issue x Create issue x

Optionally select which operations this trigger will execute for. Leave blank for all operations.

Cancel Save

> How do I access the changed value in my rule?

26

Jira Epic estimation automation

- On the next page select New Condition
- Then select Issue Fields condition

Add component

Components can either restrict execution by testing a condition, perform an action, or control flow by branching on related issues.

New branch
Create a separate section of this rule and perform actions and conditions on other items.

New condition
Actions will only execute if all conditions preceding them pass.

New action
Actions perform changes to a system.

New condition
Actions will only execute if all conditions preceding them pass.

Recommended

Issue fields condition
Check whether an issue's field meets a certain criteria **POPULAR**

All components

Advanced compare condition
Compares two values: `{{issue.status.name}} equals Done`

If / else block
Perform different actions using if, else-if and else to control the flow.

Issue attachments
Checks if issue attachments exist or don't exist

Issue fields condition
Check whether an issue's field meets a certain criteria

27

Jira Epic estimation automation

- Make the following selections
 - Field: Issue Type
 - Condition: Is one of
 - Value: Bug, Task, Story
- Click Save
- On the next page select New branch and then Branch rule/related issues

Issue fields condition

Checks whether an issue's field meets a certain criteria. [Learn more.](#)

Field *

Issue Type

Condition *

is one of

Value Field

Task x Story x Bug x

Cancel Save

Add component

Components can either restrict execution by testing a condition, perform an action, or control flow by branching on related issues.

New branch
Create a separate section of this rule and perform actions and conditions on other items.

New action
Actions perform changes to a system.

New condition
Actions will only execute if all conditions preceding them pass.

28

New branch

Perform actions and conditions on items in a separate section of this rule. Changes in this branch will not reflect in other parts of the rule. [Learn more about rule branches.](#)

Advanced branching
Perform actions and conditions on a smart value object.

Branch rule / related issues
Branch rule and run conditions & actions for these issues.

Cancel

> Get automation tips from the community

Jira Epic estimation automation

- Make the following selections
 - Type of related issues: Epic (parent)
 - Rules restricted to projects: By default, the project where the automation is created to
- Click Save
- On the next page select New Action
- On the next page select Lookup issues

Add component

Components can either restrict execution by testing a condition, perform an action, or control flow by branching on related issues.

New action
Actions perform changes to a system.

New condition
Actions will only execute if all conditions preceding them pass.

2

Branch rule / related issues

Please select which related issue you would like to perform actions against.

Type of related issues*

Epic (parent)

Rule restricted to projects

PBA test project (PTP)

Only issues from the above projects will be considered. You can change this restriction in the 'Rule details' section.

Cancel Save

1

New action

Actions perform changes to a system.

All components look

Recommended

Create lookup table
Create key to value entries that can be accessed through smart values. **NEW**

All components

Create lookup table
Create key to value entries that can be accessed through smart values.

Lookup issues
Search for issues using JQL.

3

29

Jira Epic estimation automation

- Add the following JQL script to the JQL field:
 - "Epic Link" = {{issue.key}}
- Click Save
- On the next page select New Action
- Then select Edit Issue

New action

Actions perform changes to a system.

All components edit issu

Recommended

Edit issue
Update certain issue fields. **POPULAR**

3

Lookup issues

Search for up to 100 issues using JQL and include the results list in other actions by using the **{{lookupIssues}}** smart value. Learn more about the Lookup issues action

JQL*

"Epic Link" = {{issue.key}}

Insert account id Resolve users Validate query

Cancel Save

1

Add component

New action
Actions perform changes to a system.

New condition
Actions will only execute if all conditions preceding them pass.

2

OR

Name your automation..

Who can edit this rule?*

All admins

Turn it on

30

Jira Epic estimation automation

- Then click More Options, deselect the Send Notifications tick

- Add the following script to the Additional Fields field

```
{
  "fields": {
    "timetracking": {
      "originalEstimate" : "{{lookupIssues.OriginalEstimate.sum
.divide(60)})",
      "remainingEstimate" : "{{lookupIssues.RemainingEstimate.sum
.divide(60))}}"
    }
  }
}
```

- Click Save
- Name your automation and select who can edit it
- Click Turn it on

1

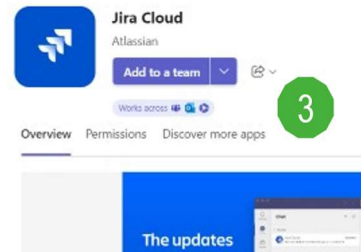
2

Jira Epic estimation automation

- When the configuration of your automation is done, the workflow should look like this
- It might take few minutes/some time for the automation to start working
- Estimates added to Tasks in the Original estimate field will be summed in the same field on the Epic

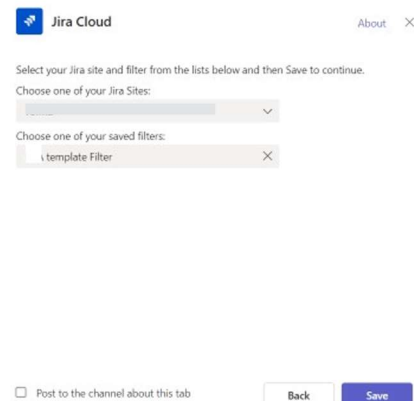
Jira and Teams integration

- Go to Teams and select Apps on the left navigation Panel
- Search Jira Cloud and click the icon
- Then click on the Add to a team button and search for the teams channel you want to connect with Jira Project
- Click Set up



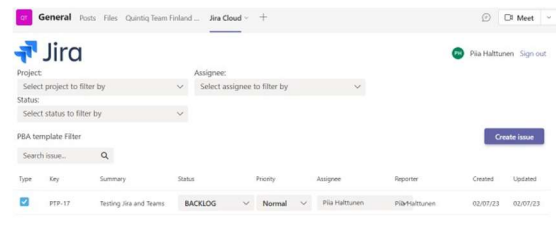
Jira and Teams integration

- Select the Jira Site (you could only have one, I think, you want to connect to Teams with)
- Then select a saved filter and choose if you want to post the channel about the connection or not
 - The above selections can be changed later
- Click Save
- Log in to Jira with your email if it asks for it



Jira and Teams integration

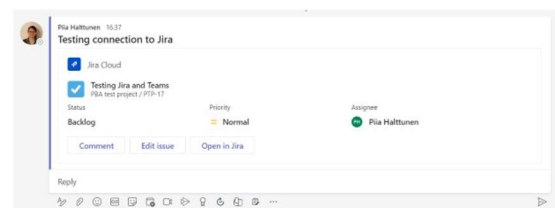
- The integration and issue overview works best with a simple saved filter like the following
 - Priority in (Medium) AND project in (test project) AND assignee is EMPTY
 - The above filter for example would list all the Medium priority issues in F test project with no assignee and show them in the Teams Jira tab
- The connection can be used for example creating issues or updating the issue status, commenting it or adding attachments to the issue



35

Jira and Teams integration

- After the integration, Jira Issues can be linked to Teams channel posts, where the team can discuss about the issue, add comments directly to the Issue or edit it



36

Thank you

More information and help can be asked from:

Piia Halttunen

