

 **amosproj / amos2021ws02-agile-trans-jira** Publicgenerated from [amosproj/amos202Xss0Y-projX-template](#) Code Issues 39 Pull requests Discussions Actions Projects 2

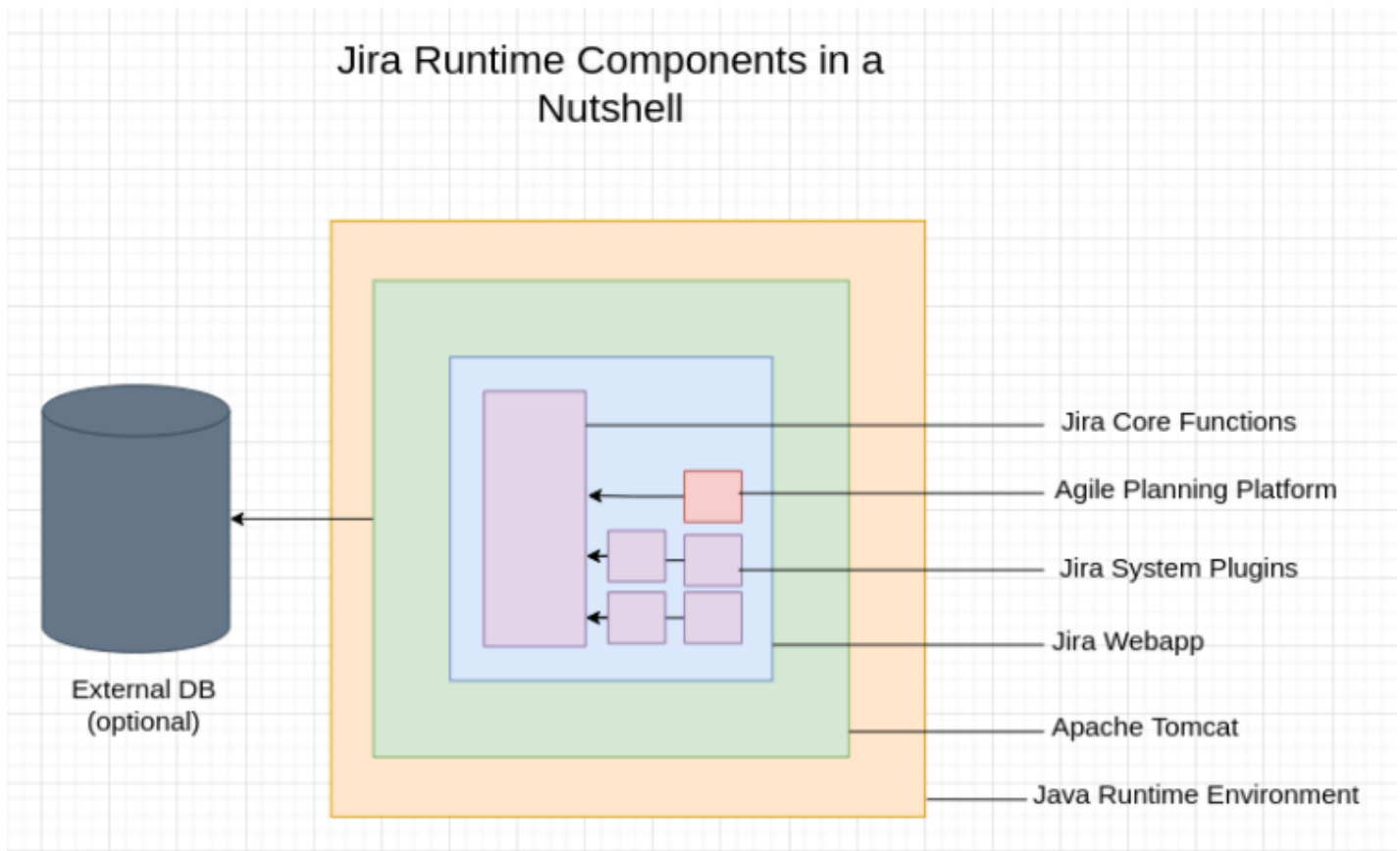
Design Documentation

[Edit](#)[New Page](#)[Jump to bottom](#)

Stefan Pawlowski edited this page yesterday · 7 revisions

Architecture of Jira

From the early analysis of the customer requirements we decided to focus on the on premise variant. Jira on-premise is a Java application, which is bundled in a modified Version of Apache Tomcat and can feature an external database, which will be called through the Apache Tomcat. The application itself consists of some Core functions and a Plugin System with some system plugins on top. It is also possible to deploy your own plugin in this environment, so called “user-installed apps”. This is the piece of code this project aims to produce in order to extend Jira with agile planning features not included in the standard version. The Plugin is pure Java code, but can also feature HTML/Javascript/CSS and JSP code on the presentation layer.

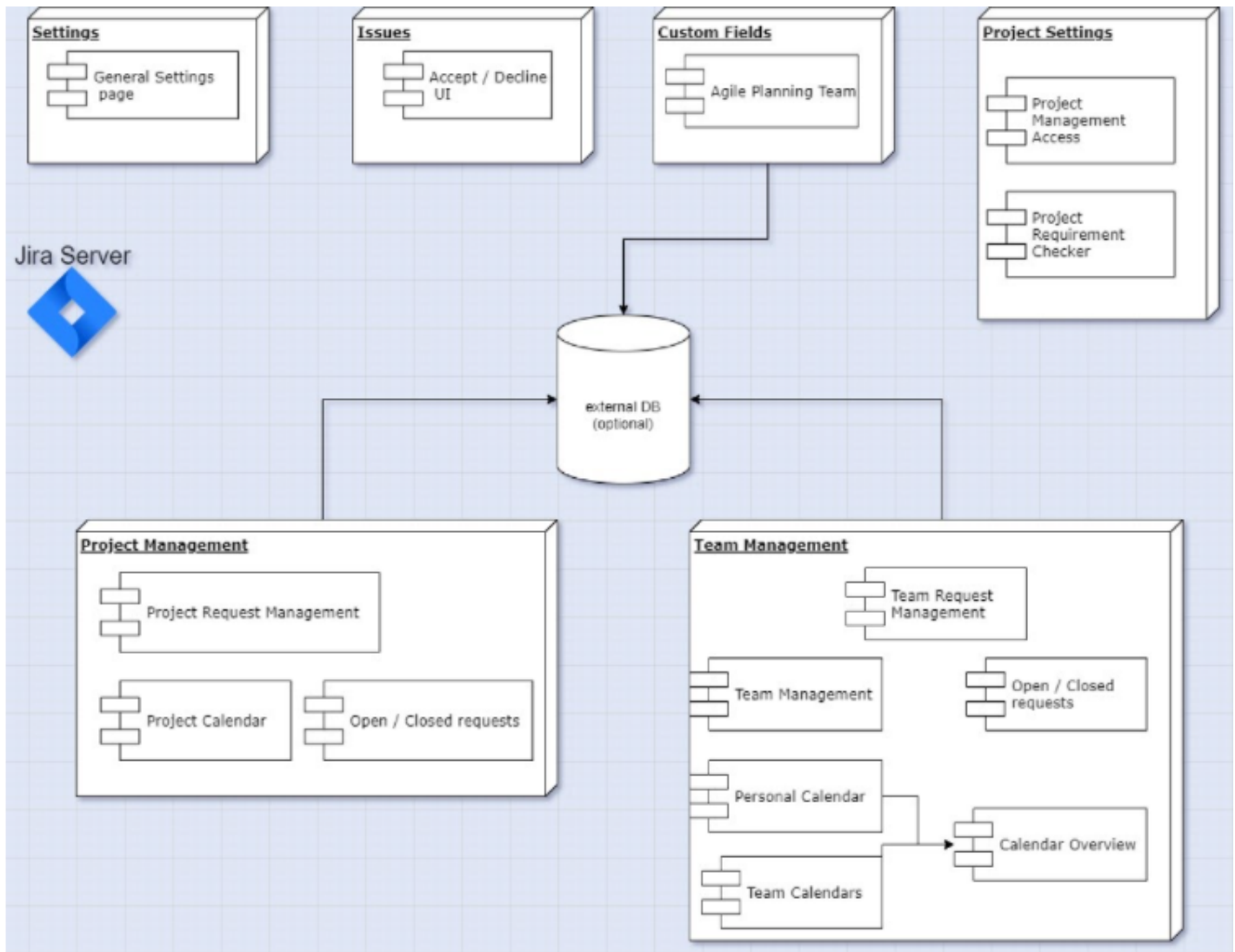


Code Components

As stated by Atlassian, modules are the most important components in a plug-in as they extend the functionality of the Jira platform, e.g. user interface but also other parts such as workflow or permissions.

In the scope of the Jirassic park project we would develop following modules:

- Settings, implementing a general setting page
- Custom fields, enabling option to choose a team, time span and epic link for the request
- Furthermore a module project setting will be implemented which control settings of a project and validates requirements
- Issues, covering the accept and decline functionality within a request
- Project management, which considers users with a project manager role and needed functionalities as: request management, project calendar and overview of open and closed requests
- Team management offers almost the same features as the project management module but from the perspective of a team manager role



Technology Summary

In our application, we strongly differentiate between the backend and frontend technology stack. During the deployment, the frontend technology stack then will be integrated into the backend. Finally, The backend serves the frontend to the user on demand.

Frontend technology in detail

We are using react and other Node.JS components in order to develop modern web components. In the future these components could be improved by the Atlassian Javascript style library.

Later, these components will be deployed using Webpack, which compresses the Javascript code and bundles it together with licenses to the backend. The backend itself features Velocity templates in order to deliver content to the user. These velocity templates are using our React components to deliver the desired design. The Javascript stack is further integrated to Maven in order to ease the handling of the code through the lifecycle of the Java application.

Backend technology in detail

As mentioned above, Jira is using Java as a core technology. As a result, we are using Maven for the deployment of our code. Jira itself is heavily depending on Springboot, which is the basis for many Jira functionalities. The maven file furthermore contains several tools for the plugin deployment in a development environment, but does not introduce further new Java dependencies to the Jira application.

The database connection is maintained by Active Objects, which is a plugin implemented into Atlassian Applications, which enables scalable data access and storage. Active Objects provides the opportunity to create a plugin data storage component and persist your own data in your plugin, it provides developers with access to a real database(in Jira is H2 Database).

+ Add a custom footer

▼ Pages 6

▸ [Home](#)

▸ [Build Documentation](#)

▼ [Design Documentation](#)

[Architecture of Jira](#)[Code Components](#)[Technology Summary](#)[Frontend technology in detail](#)[Backend technology in detail](#)

▸ [Manual Tests](#)

▸ [Project Separation in Frontend and Backend](#)

▸ [User Documentation Feature Overview](#)

+ Add a custom sidebar

Clone this wiki locally

<https://github.com/amosproj/amos2021ws02-agile-trans-jira.wiki.git>

