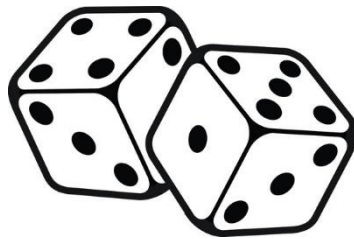


Jira Serendipity*: A Randomized Task Granting Plugin

By,
Neil Duraiswami.

Instructor: Dr. Bhriugu Celly



***Serendipity:**

Noun: "the occurrence and development of events by chance happily or beneficially."

Table of Contents

1.Introduction.....	4
2.Market Research	5
Research	5
i. Recurring Tasks for Jira Cloud by.....	5
ii. Smart Assignments and Rotations for Jira by Trundl Inc.....	6
iii. Competence-Driven Work Assignments and Skills Management by 67 talents.	7
Areas for Improvement.....	8
3.Plugin Proposal.....	9
Purpose	9
Scope.....	9
i. Random Task Assignment.....	9
ii. Integration with Competence-Based Assignment	9
iii. User-Friendly Interface	9
iv. Customization and Flexibility	10
Unique Selling Points	10
Target audience.....	10
4.Product Design	11
Use-Case Diagram	11
Wireframe.....	11
5.Development Pipeline Design Plan.....	15
Version Control.....	15
Continuous Integration (CI).....	15
Continuous Deployment (CD)	15
Monitoring and Alerting	15
6.Testing Pipeline Design Plan	16
Test Environment Setup.....	16
Test Types and Strategies	16
Test Automation.....	16
Test Data Management.....	16
Performance and Load Testing	16
Security Testing	16
User Acceptance Testing (UAT).....	16

Regression Testing.....	17
Reporting and Analysis.....	17
7.Product Roadmap.....	18
Quarter 1: Project Initiation and Foundation.....	18
Quarter 2: Core Feature Development.....	19
Quarter 3: Testing and Optimization.....	19
Quarter 4: Release and Deployment	19
8.Product Management Strategies	21
Pricing Models and Licensing Options.....	21
Go-to-Market Strategies	21
Feedback Mechanisms	21
Customer Support, Training, and Documentation	21
References	23

1.Introduction

In today's fast-paced and dynamic work environments, effective task management and collaboration are essential for driving productivity and achieving project success. Jira, a widely used project management tool, provides robust features for task tracking and project coordination. However, the availability of plugins can further enhance Jira's capabilities, offering tailored solutions to meet specific project needs.

The objective of this project is to conduct a thorough market analysis of existing plugins for Jira, identifying strengths, weaknesses, and areas for improvement. By assessing the current landscape, we aim to pinpoint gaps and opportunities where a new plugin could provide innovative solutions to enhance task management, collaboration, and productivity within Jira projects.

With the insights gained from market research, the project will focus on conceptualizing and developing a new plugin that addresses the identified needs and challenges. This new plugin will aim to offer unique features and functionalities that complement and extend Jira's existing capabilities, providing users with enhanced flexibility, efficiency, and effectiveness in managing their projects.

Furthermore, the project will involve designing a comprehensive product management strategy for the new plugin. This strategy will encompass pricing models, licensing options, go-to-market strategies, feedback mechanisms, and customer support strategies. By formulating a well-rounded product management plan, we aim to ensure the successful launch, adoption, and ongoing success of the new plugin in the competitive landscape of Jira plugins.

2. Market Research

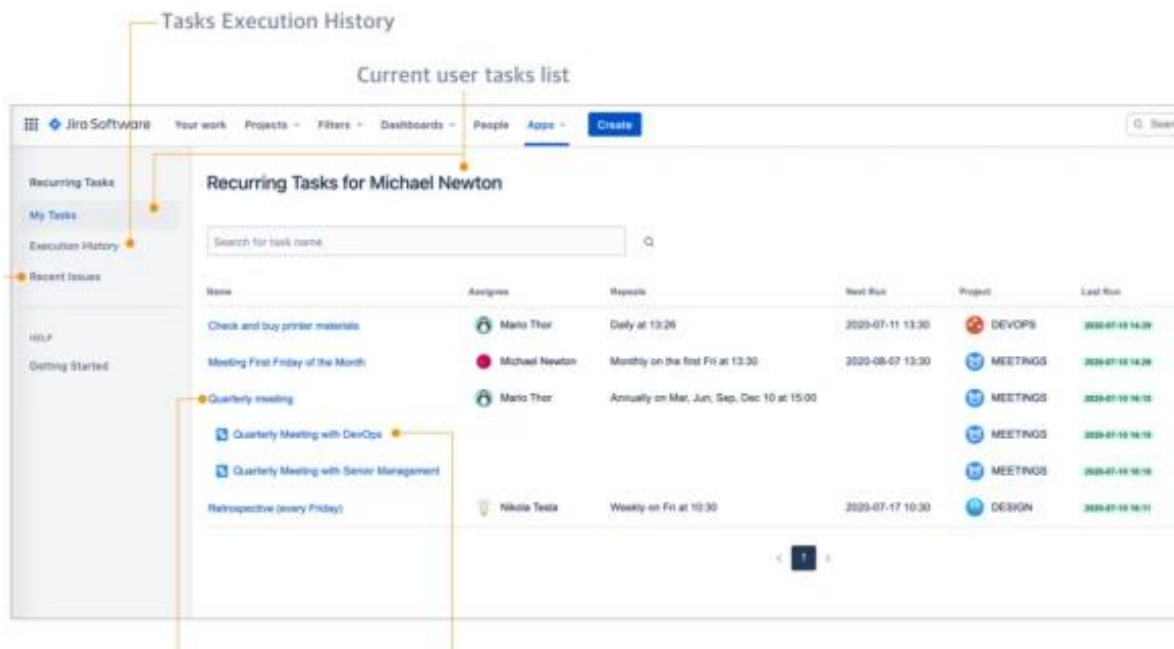
Research

i. Recurring Tasks for Jira Cloud by [Gebsun Software](#).

Automatic Task Creation: Users can set up tasks to be automatically created in Jira at specified intervals or dates in the future.

Support for Cron Format: By supporting cron format, the plugin caters to users who are accustomed to using this standard for scheduling tasks. It offers flexibility for users who may have specific scheduling requirements that are best expressed using cron expressions.

Calendar View: The calendar view provides users with a visual representation of their task schedule. This allows users to quickly see when tasks are scheduled to occur and monitor task executions. It enhances the overall user experience by offering a familiar and intuitive way to interact with task schedules.



ii. Smart Assignments and Rotations for Jira by [Trundl Inc.](#)

Schedule and Rotation Creation: Users can easily create schedules and rotations for their teams within Jira. This likely involves setting up recurring schedules for team members, defining shift timings, and establishing rotation patterns.

Rule-based Ticket Assignment: The plugin allows users to write rules for auto-assigning tickets based on the schedules they've set up. This automation helps streamline ticket management by ensuring that incoming tickets are assigned to team members according to predefined criteria.

Customizable Working Days and Shift Timings: Users can configure their working days, shift timings, and mid-shift breaks according to their team's specific requirements. This customization ensures that the scheduling system aligns with the team's operational needs.

The screenshot displays the 'Schedule Configuration' interface for the Trundl Inc. plugin. It is divided into two main sections: 'Basic Rotation' and 'Advanced Rotation', with a 'Switch to Advanced' button highlighted by an orange arrow.

Schedule Configuration:

- Name:** Schedule for Bugs
- Issue Type:** Bug
- Issue Status:** To Do
- Buttons: Next, Save Draft, Cancel

Basic Rotation:

- Name:** Rotation for Bugs
- Color:** Yellow
- Time Zone:** Asia/Kolkata
- Type:** Fixed Time
- Duration:** Enter interval count
- Interval Unit:** Select Interval Unit
- Start Date:** Select Start Date
- Perpetual Date:** ☒
- End Date:** Select End Date
- 24 Hours:** ☒
- Team Members:** Select...

Advanced Rotation:

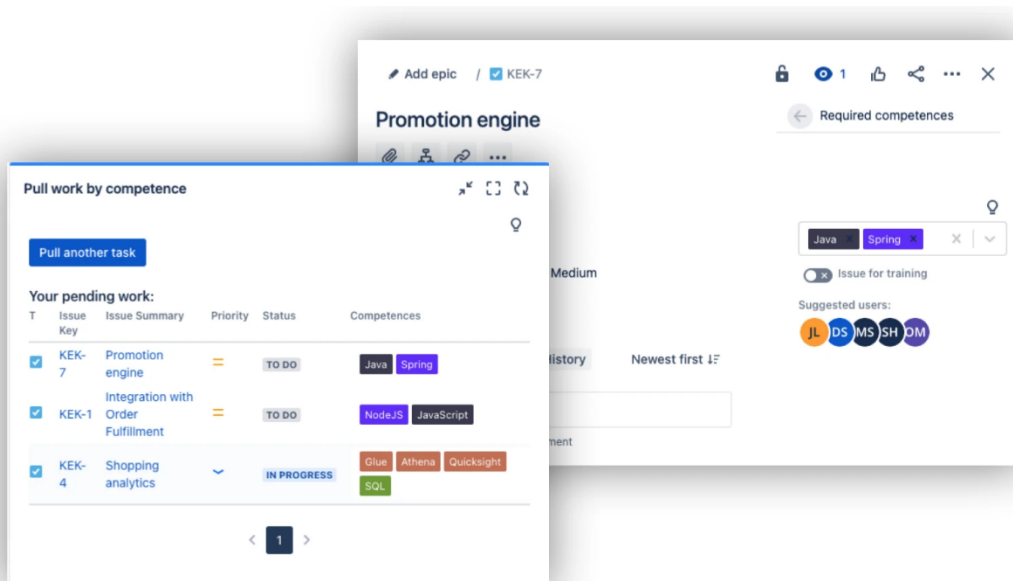
- Name:** U3_Advanced_Rotation
- Color:** Pink
- Type:** Fixed Time
- Duration:** 30
- Start Date:** 2023-04-10
- Perpetual Date:** ☒
- 24 Hours:** ☒
- SQL Set 1:**
 - JQL:** priority = Highest
 - Time Zone:** Asia/Kolkata
 - Team Members:** Select Team Member
- Enable In-Hours Maintenance:** ☐

iii. Competence-Driven Work Assignments and Skills Management by [67 talents](#).

Competence-Based Task Assignment: The plugin automates the assignment of tasks by matching the competencies of users with the skill requirements of the tasks. This ensures that tasks are assigned to individuals who possess the necessary skills and expertise to complete them effectively.

Competence Definition: Administrators or managers can define competence structures within the organization using the plugin. This involves establishing a framework that outlines the various competencies required for different roles or tasks within the organization.

User Skill Assignment: User's competencies can be determined by assigning specific skills to them within the plugin. This allows organizations to accurately assess the capabilities of their workforce and ensure that tasks are assigned to individuals who possess the requisite skills.



Areas for Improvement

Random Task Assignment: None of the existing plugins offer a feature for random task assignment. Introducing this functionality would provide users with a unique approach to task allocation, promoting cross-functional collaboration and skill development within teams. By randomly assigning tasks based on predefined criteria such as availability, workload, skills, or rotation schedules, the new plugin could help teams foster adaptability, versatility, and a more dynamic work environment.

Integration with Competence-Based Assignment: While Competence-Driven Work Assignments and Skills Management by 67 talents offers competence-based task assignment, integrating random task assignment with competency matching could further enhance task allocation efficiency. By considering both competency and randomness in task assignment, the plugin could ensure tasks are assigned not only to qualified individuals but also provide opportunities for skill development and exposure to diverse tasks.

3.Plugin Proposal

Purpose

In response to market research findings revealing gaps in existing Jira plugins, I propose the development of Jira Serendipity—a groundbreaking plugin designed to revolutionize task assignment within Jira Cloud. Jira Serendipity aims to enhance team collaboration, promote skill development, and streamline task allocation by introducing innovative features such as random task assignment and integration with competence-based assignment.

Scope

Jira Serendipity will be a comprehensive task management plugin for Jira Cloud, offering the following key functionality and features:

- i. **Random Task Assignment**
 - Develop a sophisticated algorithm that randomly assigns tasks to team members based on predefined criteria such as availability, workload, skills, and rotation schedules.
 - Provide users with configurable parameters to adjust the randomness of task assignment according to their team's specific needs.
 - Implement features for tracking and reporting on task assignments, ensuring transparency and accountability in the assignment process.
- ii. **Integration with Competence-Based Assignment**
 - Integrate competency matching into the task assignment algorithm to ensure tasks are assigned to individuals with the requisite skills and expertise.
 - Develop mechanisms for assessing and tagging users' competencies within the plugin, enabling administrators or managers to define competence structures and assign skills to team members.
 - Ensure seamless interoperability between random task assignment and competency-based assignment features, allowing tasks to be allocated based on a combination of randomness and competency matching.
- iii. **User-Friendly Interface**
 - Design an intuitive and user-friendly interface that seamlessly integrates with the existing Jira Cloud environment, ensuring a smooth user experience for task management.
 - Provide comprehensive documentation and support resources to help users maximize the potential of the plugin and leverage its features effectively.

iv. Customization and Flexibility

- Offer users the flexibility to customize task assignment parameters and criteria according to their team's unique requirements, allowing for adaptability and versatility in task allocation.
- Provide options for integrating with existing Jira workflows and processes, ensuring compatibility and ease of adoption for users.

Unique Selling Points

1. Innovative Random Task Assignment
2. Integration with Competence-Based Assignment
3. Seamless Jira Cloud Integration
4. Customization and Flexibility

Target audience

Project Managers and Team Leads: Project managers and team leads responsible for task allocation and project coordination will benefit from Jira Serendipity's ability to streamline task assignment. The plugin's random task assignment feature alleviates the burden of manually assigning tasks, allowing managers to focus on higher-level project planning and strategic decision-making.

Team Members and Contributors: Team members and contributors involved in project execution will benefit from Jira Serendipity's dynamic task allocation approach. The randomness in task assignment encourages cross-functional collaboration and skill development by exposing team members to diverse tasks and challenges, fostering a more dynamic and engaging work environment.

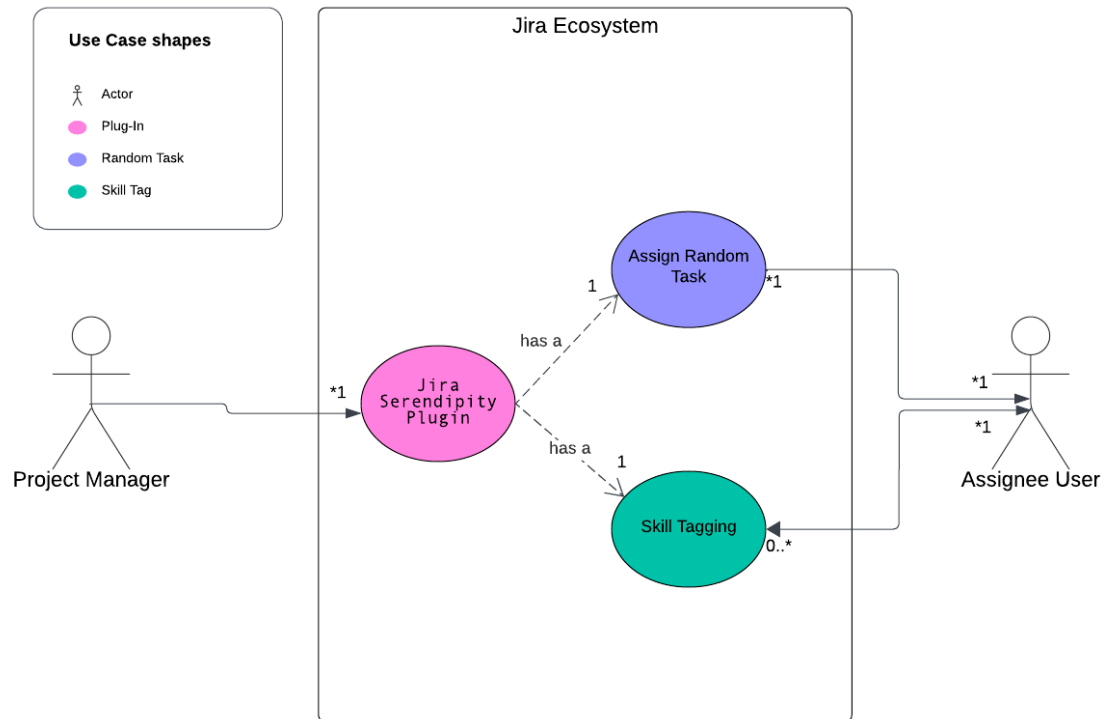
Human Resource Managers: Human resource managers responsible for talent management and skill development initiatives will find Jira Serendipity valuable in facilitating competency-based task assignment. The integration of competency matching ensures tasks are assigned to individuals with the requisite skills, promoting skill development and proficiency within the workforce.

Organizational Leaders and Executives: Organizational leaders and executives overseeing strategic initiatives and project portfolios will appreciate Jira Serendipity's ability to drive innovation and productivity. By promoting cross-functional collaboration, skill development, and adaptability, the plugin contributes to the overall effectiveness and success of projects, aligning with organizational goals and objectives.

Consultants and Freelancers: Consultants and freelancers working on projects for multiple clients will benefit from Jira Serendipity's ability to efficiently manage task assignments. The plugin's random task assignment feature ensures equitable distribution of tasks, preventing workload imbalances and enhancing productivity in a dynamic project environment.

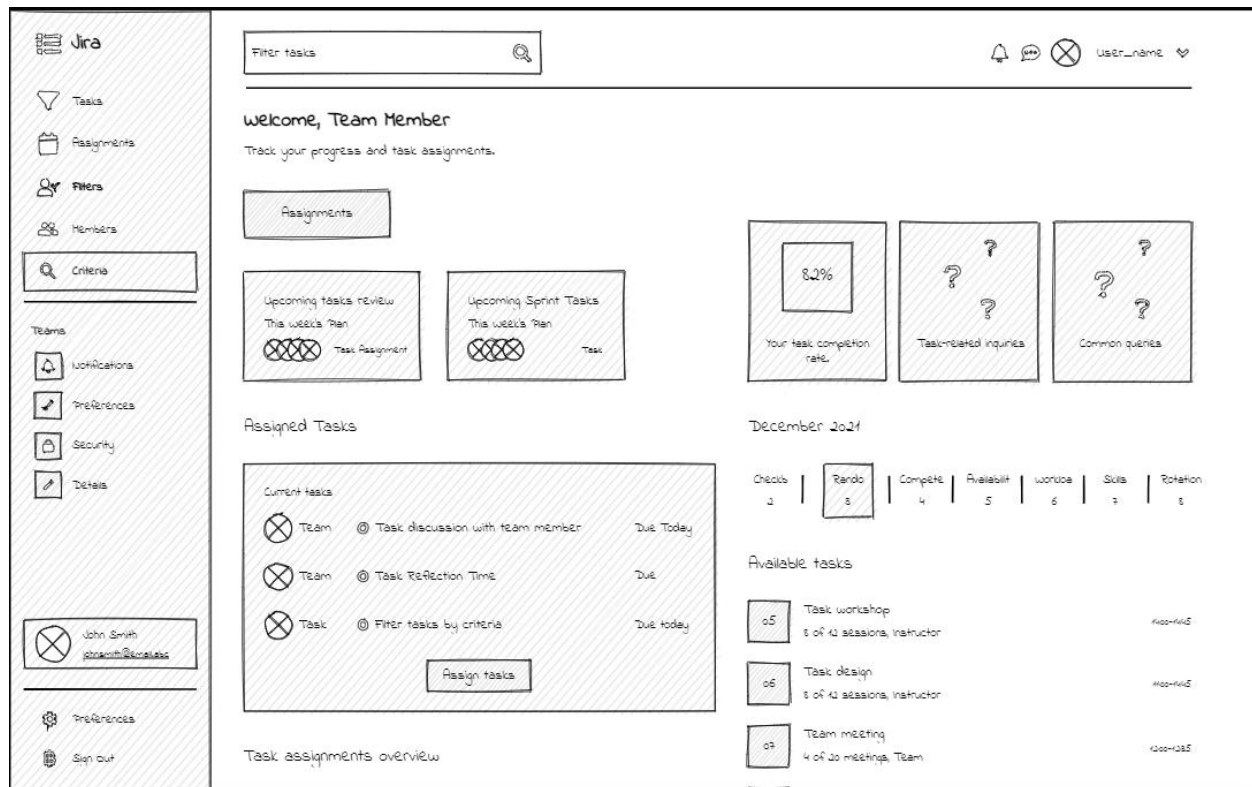
4.Product Design

Use-Case Diagram



Wireframe

Dashboard: A centralized dashboard presents users with task assignments, deadlines, and progress. Task cards display pertinent details like name, description, assigned user, deadline, and status.



Task Assignment: The task assignment interface displays available tasks and allows users to indicate preferences. Users can manually select tasks or opt into the random assignment pool, supported by tooltips for guidance.

Filter tasks

Apply criteria

Random Competency Availability Workload

Task list

- ☒ Task details - 01/23
Urgent task - E-commerce
- ☒ Task details - 02/23
Urgent task - E-commerce
- ☒ Quarterly report task
Monthly company reports

Search options

Jira

Tasks

Assignments

Filters

Members

Criteria

Teams

Notifications

Preferences

Security

Details

John Smith
john.smith@acme.com

Preferences

Sign out

Task Assignment overview

TASK DESCRIPTION This form captures task details for assignment to team members.

ASSIGNMENT Team (all)

DETAILS

Q1 What is the task description and requirements for assignment?

1 10

Q2 Is this task assigned based on competency or availability?

Confirm Cancel

Save assignment

Configuration Settings & Feedback Mechanism(Competency based tasks):

Configuration settings permit users to adjust assignment criteria such as availability, workload, skills, and rotation schedules. Customization options cater to team-specific needs. A feedback mechanism enables users to rate assignments, provide comments, and offer suggestions. Aggregated feedback aids administrators in optimizing task assignment algorithms.

- Tasks
- Assignments
- Filters
- Members
- Criteria**

John Smith

john.smith@acme.com

- Preferences
- Sign out

Filter tasks by

Assign task

Filter

Recent task updates

☒
Random assignment
Task assigned to team member.

☒
Competency-based task
Task completed by team member.

☒
Creative brainstorming session
Task paused by team member.

☒
Visual design exploration
Task pending review by team member.

☒
Annual planning session
Task reviewed and approved by team member.

Competency-based task

Assignee selection

Due date

Ongoing projects

Task priority

File attachments

Upload file

Task references

Include

Move to archive

Remove task

5. Development Pipeline Design Plan

Version Control

- Utilize Git as the version control system.
- Developers work on individual branches for features or fixes.
- Changes merged into the main branch via pull requests.
- Use tags to mark releases and milestones.

Continuous Integration (CI)

- Implement CI with Jenkins.
- Automated build process triggered on each push to the main branch.
- Includes compiling code, running tests, and generating artifacts.
- Ensures immediate validation and integration of changes.

Continuous Deployment (CD)

- Manage CD pipeline with Docker and Kubernetes.
- Artifacts packaged into Docker containers post CI.
- Deployment to staging environment for testing and validation.
- Promote approved containers to production for seamless updates.

Monitoring and Alerting

- Utilize Prometheus for metrics collection.
- Grafana for visualization of metrics.
- Set up alerts for real-time notification of anomalies or issues.
- Enables proactive response and resolution to maintain application health.

6. Testing Pipeline Design Plan

Test Environment Setup

- Establish dedicated testing environments mirroring production.
- Utilize Docker containers for consistent environment configuration.
- Ensure compatibility with CI/CD pipelines for seamless integration.

Test Types and Strategies

- Implement a combination of unit, integration, and end-to-end testing.
- Utilize automated testing frameworks such as JUnit, Selenium, and Jest.
- Prioritize test coverage for critical features and functionalities.

Test Automation

- Integrate testing frameworks with CI pipeline for automated execution.
- Schedule regular test runs triggered by code changes or scheduled intervals.
- Monitor test results and track metrics to identify trends and patterns.

Test Data Management

- Establish procedures for managing test data and environments.
- Utilize tools for data generation, sanitization, and migration.
- Ensure data privacy and compliance with regulatory requirements.

Performance and Load Testing

- Conduct performance and load testing to assess system scalability and reliability.
- Utilize tools like JMeter or Gatling for simulating user traffic and stress testing.
- Analyze performance metrics to identify bottlenecks and optimize system resources.

Security Testing

- Perform regular security assessments to identify vulnerabilities and threats.
- Utilize automated scanning tools for static and dynamic code analysis.
- Implement security best practices to mitigate risks and protect sensitive data.

User Acceptance Testing (UAT)

- Collaborate with stakeholders to define acceptance criteria and test scenarios.
- Conduct UAT in a controlled environment to validate user requirements.
- Gather feedback and iterate on improvements based on user input.

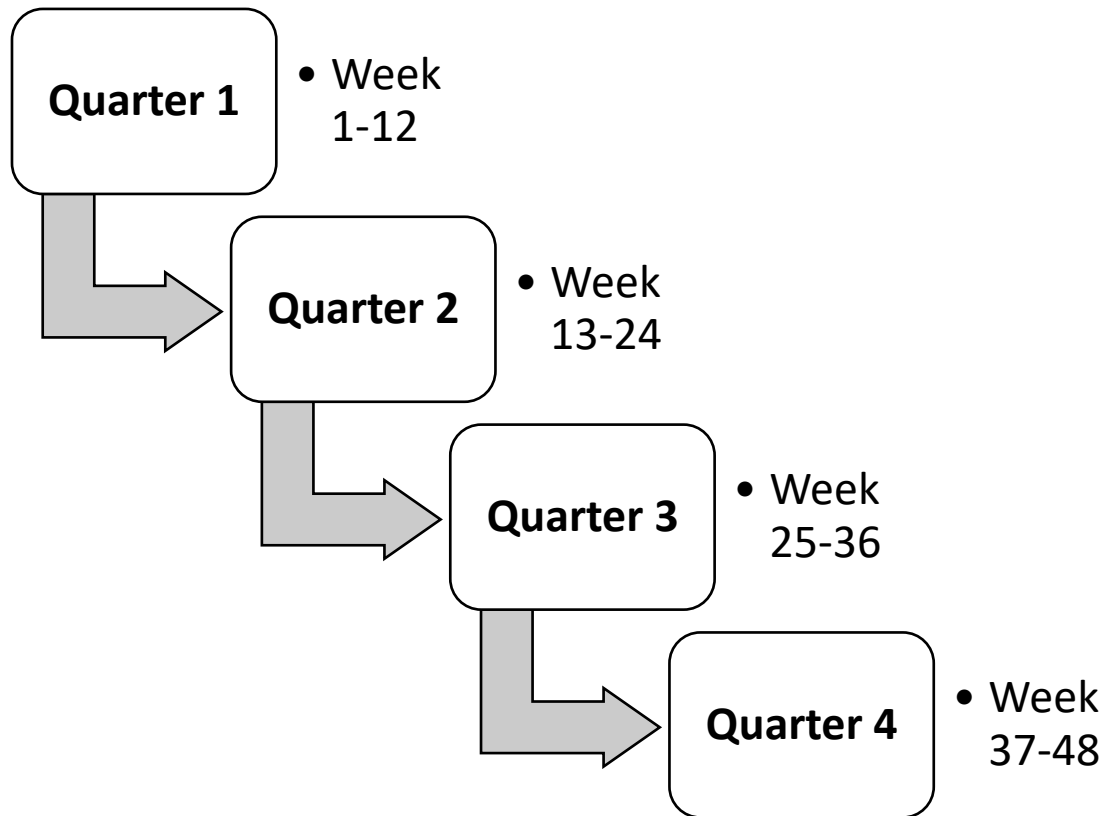
Regression Testing

- Implement automated regression testing to ensure code changes do not introduce defects.
- Maintain a comprehensive regression test suite covering critical workflows and functionalities.
- Execute regression tests as part of the CI/CD pipeline to validate each build.

Reporting and Analysis

- Generate test reports detailing test results, including pass/fail status and defects identified.
- Analyze testing metrics to assess quality trends and identify areas for improvement.
- Share insights with the development team to inform decision-making and drive continuous improvement.

7.Product Roadmap



Quarter 1: Project Initiation and Foundation (Month 1-3)

- i. **Project Kickoff (Week 1)**
 - Conduct project kickoff meeting to align stakeholders on project goals, scope, and timelines.
 - Establish communication channels and collaboration tools for the development team.
- ii. **Requirement Gathering and Analysis (Week 2-4)**
 - Gather detailed requirements for the plugin, including user stories, use cases, and acceptance criteria.
 - Conduct stakeholder interviews and workshops to ensure alignment with user needs and expectations.
- iii. **Architecture Design and Planning (Week 5-8)**
 - Design the architecture of the plugin, considering scalability, performance, and integration requirements.
 - Develop a detailed technical design document outlining components, workflows, and data models.
- iv. **Setup Development Environment (Week 9-12)**
 - Set up development environment with version control, CI/CD pipelines, and testing frameworks.
 - Define coding standards, conventions, and development workflows for the team.

Quarter 2: Core Feature Development (Month 4-6)

- i. **Random Task Assignment (Week 13-16)**
 - Implement random task assignment algorithm based on predefined criteria such as availability, workload, and skills.
 - Integrate random assignment feature with Jira Cloud API for seamless task allocation.
- ii. **Competence-Based Assignment Integration (Week 17-20)**
 - Develop competency matching algorithm for assigning tasks based on user skills and expertise.
 - Integrate competency-based assignment feature with user profiles and task categorization.
- iii. **User Interface Design and Development (Week 21-24)**
 - Design and develop user-friendly interfaces for task assignment, configuration settings, and feedback mechanisms.
 - Ensure consistency and usability across different devices and screen sizes.

Quarter 3: Testing and Optimization (Month 7-9)

- i. **Unit and Integration Testing (Week 25-28)**
 - Conduct unit and integration testing to validate functionality and identify any issues or bugs.
 - Develop automated test suites for regression testing and continuous validation.
- ii. **User Acceptance Testing (Week 29-32)**
 - Collaborate with stakeholders to conduct user acceptance testing in a controlled environment.
 - Gather feedback and iterate on improvements based on user input and validation results.
- iii. **Performance and Security Testing (Week 33-36)**
 - Perform performance and load testing to assess system scalability and reliability.
 - Conduct security assessments to identify vulnerabilities and ensure compliance with security standards.

Quarter 4: Release and Deployment (Month 10-12)

- i. **Staging Deployment and Validation (Week 37-40)**
 - Deploy the plugin to staging environment for final validation and user acceptance.
 - Conduct end-to-end testing to ensure compatibility and functionality across different environments.
- ii. **Production Deployment (Week 41-44)**
 - Promote approved builds to production environment for release to end users.
 - Monitor deployment process and address any issues or challenges that arise.

iii. Post-Release Support and Maintenance (Week 45-48)

- Provide ongoing support and maintenance for the plugin, including bug fixes and feature enhancements.
- Gather feedback from users and iterate on improvements to ensure continuous product evolution.

8.Product Management Strategies

Pricing Models and Licensing Options

- We will adopt a tiered pricing model, offering different packages tailored to the needs of various customer segments. This model allows for flexibility and scalability, accommodating both small teams and enterprise-level organizations.
- Our licensing options will include both perpetual licenses and subscription-based models. Perpetual licenses provide customers with long-term access to the plugin, while subscription-based models offer the advantage of continuous updates and support.

Go-to-Market Strategies

- Targeted Marketing:** We will employ targeted marketing campaigns to reach our key audience segments, including software development teams, project managers, and Jira users. These campaigns will emphasize the unique value proposition of the plugin in enhancing task management and fostering collaboration.
- Partnerships and Integrations:** Collaboration with Jira consulting firms, software integrators, and technology resellers will help expand our market reach and distribution channels. Integrating our plugin with complementary tools and platforms will enhance its value proposition and increase visibility.
- Customer Education:** Comprehensive customer education initiatives, including tutorials, webinars, documentation, and online forums, will be provided to ensure customers fully understand the plugin's capabilities and how to leverage them effectively.

Feedback Mechanisms

- User Feedback Surveys:** Regular surveys will be conducted to gather user input on their experience with the plugin, including feature preferences, usability issues, and overall satisfaction.
- In-App Feedback:** Built-in feedback mechanisms within the plugin interface will allow users to provide real-time feedback on specific features or functionalities.
- User Forums and Communities:** Online forums and user communities will be established to facilitate discussions, idea sharing, and collaboration among users. This will provide valuable insights into user needs and preferences.

Customer Support, Training, and Documentation

- i. **Customer Support Channels:** Multiple channels, including email, chat support, and a dedicated support portal, will be available to address customer queries, troubleshoot issues, and provide timely assistance.
- ii. **Training Programs:** Comprehensive training programs will be developed to onboard new users and familiarize them with the plugin's features and functionalities. This will include both self-paced learning resources and live training sessions conducted by experienced trainers.
- iii. **Documentation:** Extensive documentation, including user guides, FAQs, and troubleshooting manuals, will be provided to support users in navigating the plugin and resolving common issues. Regular updates to documentation will ensure it remains current and relevant to user needs.

References

1. marketplace.atlassian.com
2. chat.openai.com
3. uizard.io
4. lucid.app
5. Celly, B. (2023). *Software Engineering (PowerPoint slides)*. CSC 581: Software Engineering, Department of Computer Science, California State University, Dominguez Hills.