Hybrid or Custom Agile Lab Experimentation

Soniah N - 192311012

Custom Setup in Jira

1. **Create a Custom Issue Type**

- **Name**: Experiment
- **Description**: For managing Agile lab experiments that mix Scrum, Kanban, and Lean principles to maximize flexibility.

Custom Fields:

1. **Experiment Type**:
    - Dropdown options: New Feature, Marketing Test, Technical Prototype, A/B Test, etc.
2. **Hypothesis**:
    - Text field for stating the experiment's hypothesis (e.g., "We believe that shortening onboarding will increase user retention by 15%.").
3. **Goals**:
    - Text field for defining the objectives of the experiment (e.g., "Reduce user churn during onboarding by simplifying the process.").
4. **Execution Method**:
    - Dropdown options: Scrum, Kanban, Hybrid.
5. **Metrics to Measure**:
    - Multi-select field for metrics (e.g., User Feedback, Conversion Rate, Technical Performance).
6. **Status**:
    - Options: Define Hypothesis, Testing, Iterating, Analyzing, Completed.
7. **Findings**:
    - Text area to document insights and learnings from the experiment.
8. **Decision**:
    - Options: Implement, Pivot, Discard, Retry.

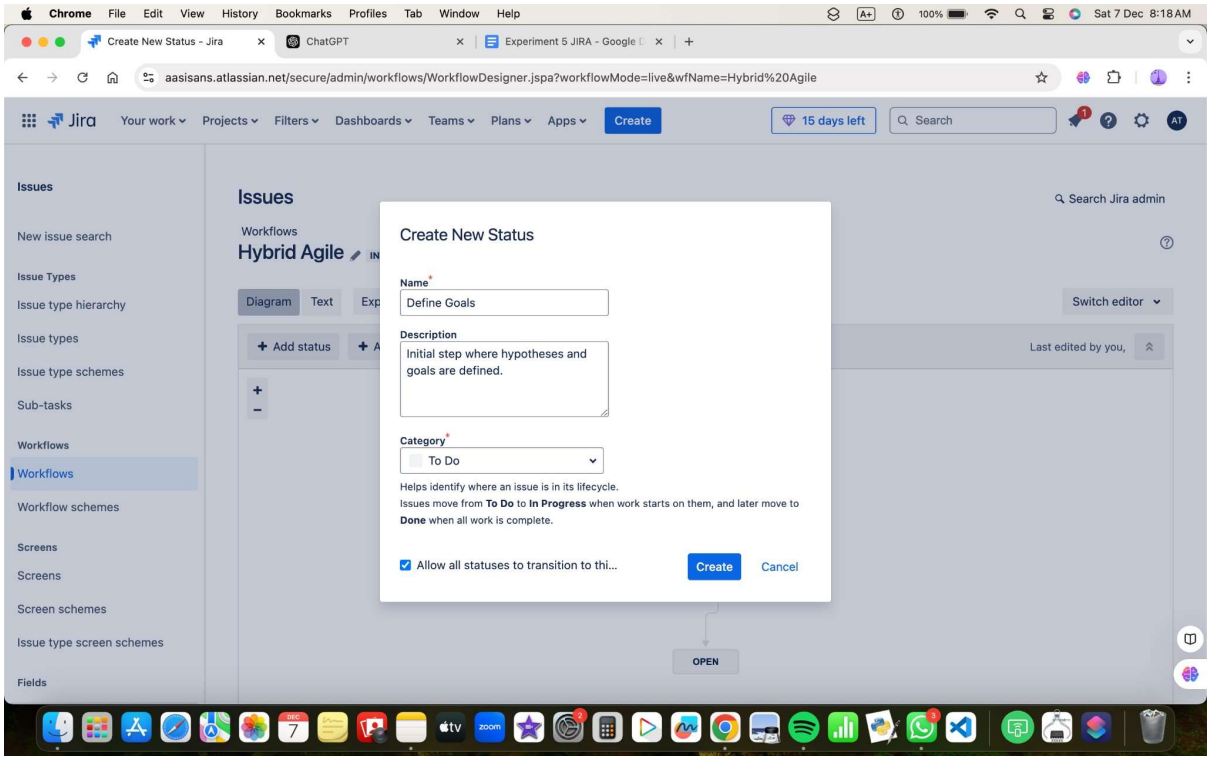
Workflow for Hybrid Agile Experiments

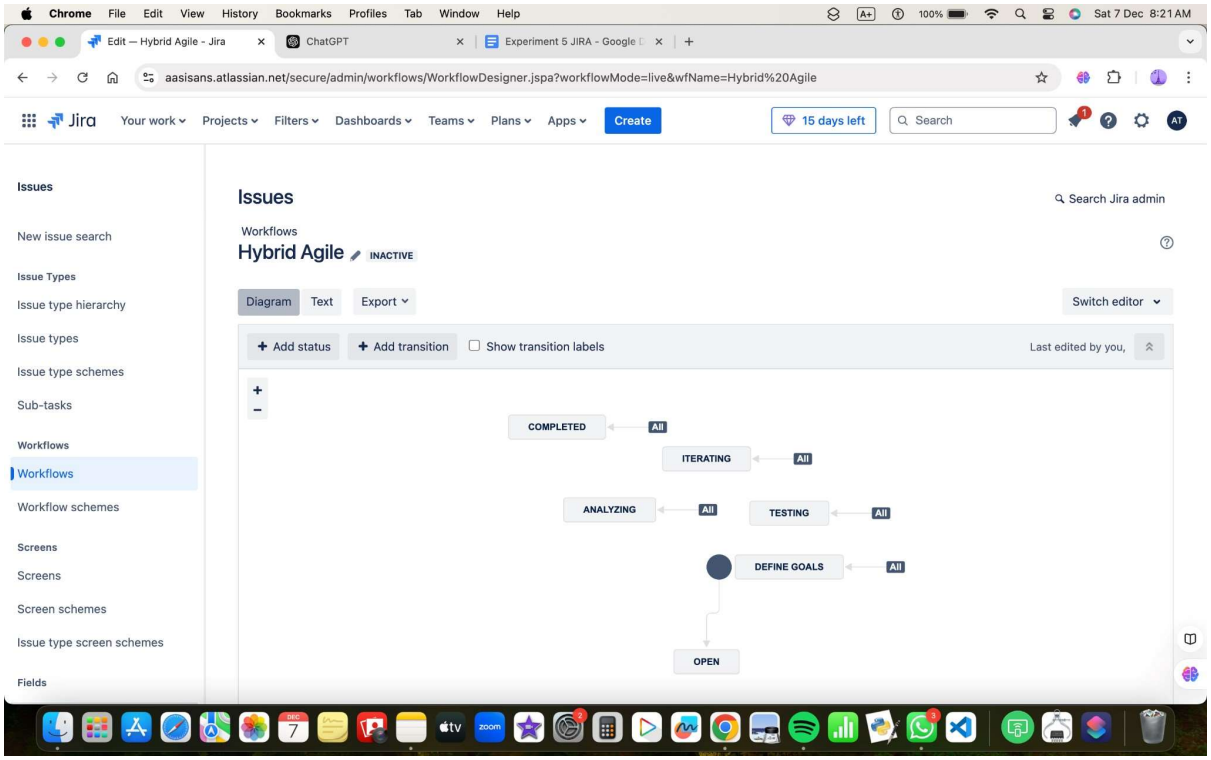
Statuses and Steps:

1. **Define Hypothesis and Goals**:

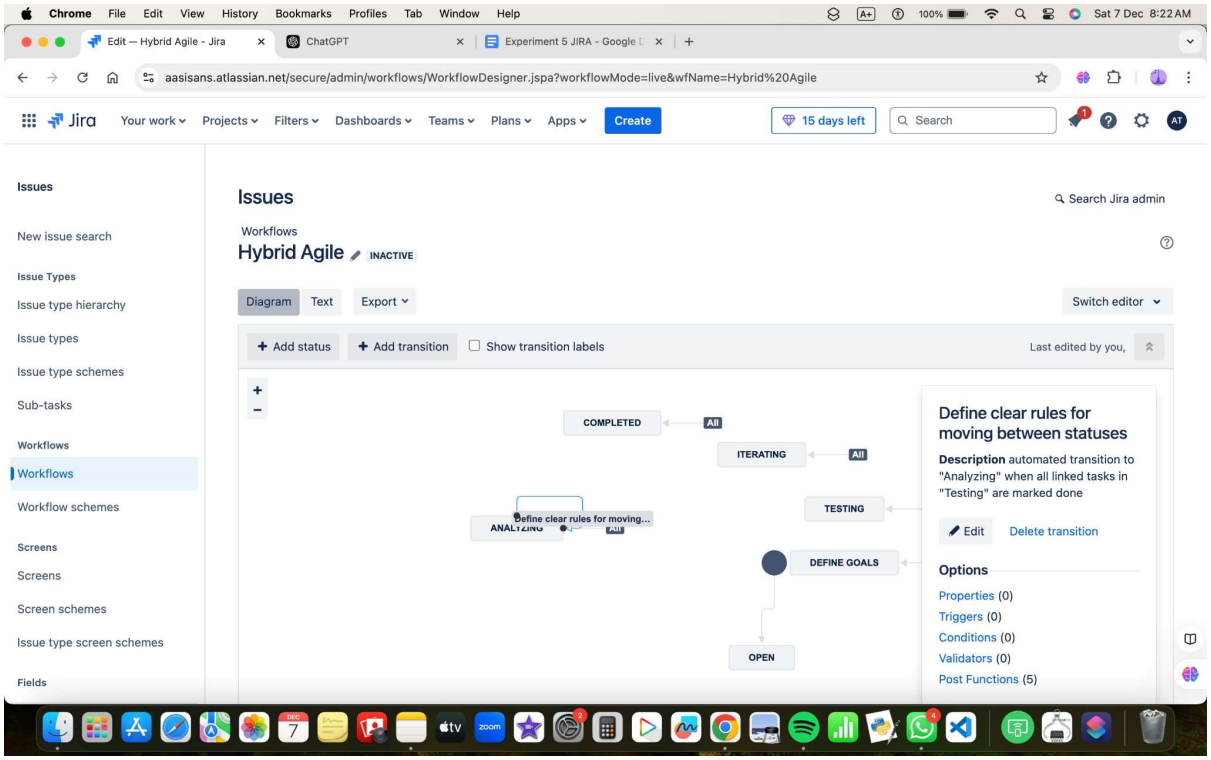
    - Clearly define what you aim to test and achieve.
    - Include the hypothesis and metrics to validate or invalidate it.
    - **Exit Criteria**: Hypothesis and goals documented in the Jira issue.

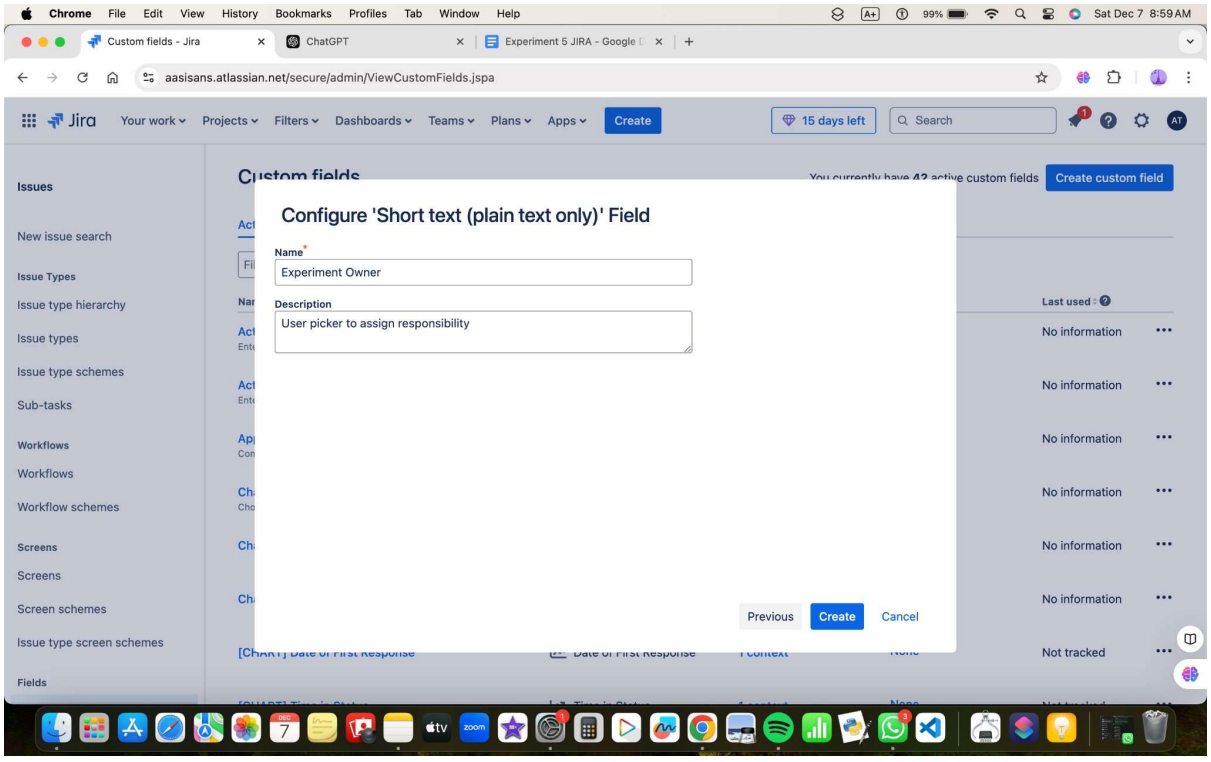
2. **Plan and Execute**:
    - Choose Scrum (time-boxed sprints) or Kanban (continuous flow) based on the nature of the experiment:
        - **Scrum**: Break tasks into stories or subtasks and plan sprints.
        - **Kanban**: Visualize tasks on a Kanban board for a continuous flow of work.
    - Develop the Minimal Viable Experiment (MVE).
    - **Exit Criteria**: Experiment is launched.
3. **Testing**:
    - Run the experiment and collect data continuously.
    - Monitor metrics like user feedback, conversion rates, or system performance.
    - **Exit Criteria**: Data collection is complete.
4. **Analyze and Learn**:
    - Evaluate the results to validate or invalidate the hypothesis.
    - Document findings in the Jira issue.
    - **Exit Criteria**: Analysis and learnings documented.
5. **Iterate or Pivot**:
    - Based on findings, decide the next steps:
        - **Iterate**: Refine the experiment and retest.
        - **Pivot**: Change the approach or hypothesis.
        - **Implement**: Roll out successful experiments.
        - **Discard**: Stop pursuing the idea.
    - **Exit Criteria**: Final decision is documented.


Create New Status - Jira
ChatGPT
Experiment 5 JIRA - Google
aasisans.atlassian.net/secure/admin/workflows/WorkflowDesigner.jspa?workflowMode=live&wfName=Hybrid%20Agile
Jira
Your work
Projects
Filters
Dashboards
Teams
Plans
Apps
Create
15 days left
Search
Issues
New issue search
Issue Types
Issue type hierarchy
Issue types
Issue type schemes
Sub-tasks
Workflows
Workflow schemes
Screens
Screen schemes
Issue type screen schemes
Fields
Issues
Search Jira admin
Workflows
Hybrid Agile
Diagram
Text
Switch editor
Add status
Last edited by you,
Create New Status
Name
Define Goals
Description
Initial step where hypotheses and goals are defined.
Category
To Do
Helps identify where an issue is in its lifecycle.
Issues move from To Do to In Progress when work starts on them, and later move to Done when all work is complete.
Allow all statuses to transition to thi...
Create
Cancel
OPEN


Edit — Hybrid Agile - Jira
ChatGPT
Experiment 5 JIRA - Google
aasisans.atlassian.net/secure/admin/workflows/WorkflowDesigner.jspa?workflowMode=live&wfName=Hybrid%20Agile
Jira
Your work
Projects
Filters
Dashboards
Teams
Plans
Apps
Create
15 days left
Search
Issues
New issue search
Issue Types
Issue type hierarchy
Issue types
Issue type schemes
Sub-tasks
Workflows
Workflow schemes
Screens
Screen schemes
Issue type screen schemes
Fields
Issues
Search Jira admin
Workflows
Hybrid Agile
INACTIVE
Diagram
Text
Export
Switch editor
Add status
Add transition
Show transition labels
Last edited by you,
COMPLETED
All
ITERATING
All
ANALYZING
All
TESTING
All
DEFINE GOALS
All
OPEN

\





Example: A/B Test for App Onboarding

Issue Setup in Jira:

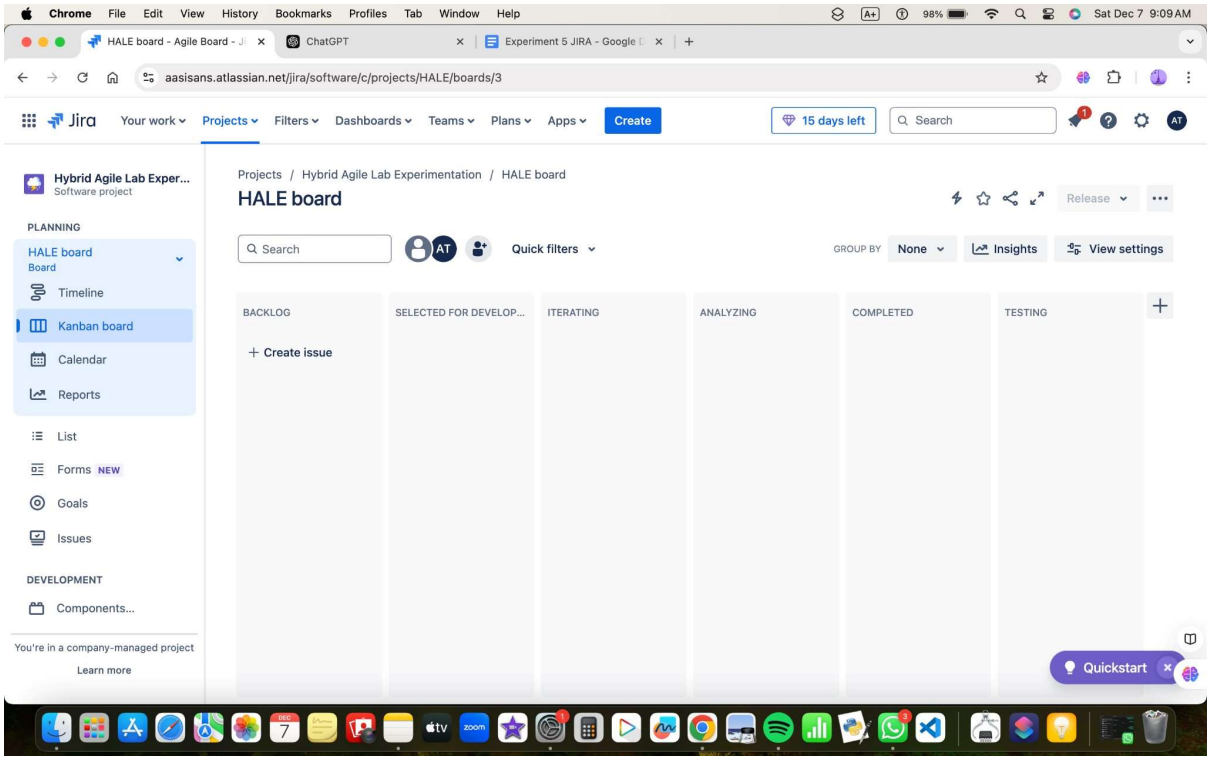
- **Summary**: A/B Test for App Onboarding

- **Experiment Type**: A/B Test
- **Execution Method**: Hybrid (Scrum for development, Kanban for experiment flow).
- **Hypothesis**: "We believe that reducing onboarding steps will increase user retention by 20%."
- **Goals**: Increase user retention by optimizing onboarding. ●**Metrics to Measure**: Retention Rate, Conversion Rate.

Workflow in Jira:

1. **Define Hypothesis and Goals**:
    - Document hypothesis and goals in the issue.
    - Assign tasks to team members for preparation.
2. **Plan and Execute**:
    - Use **Scrum** to plan development tasks:
        - Story 1: Design onboarding variants (short and long versions).
        - Story 2: Implement A/B testing logic.
        - Story 3: Deploy to staging for QA.
    - Use **Kanban** to track the experiment:
        - Task 1: Launch Variant A.
        - Task 2: Launch Variant B.
3. **Testing**:
    - Run the experiment for a week.
    - Collect data on retention and conversion rates for both variants.
4. **Analyze and Learn**:
    - Variant A shows a 25% increase in retention compared to Variant B. ○ Document findings in the **Findings** field.
5. **Decide**:
    - Decision: Implement Variant A as the new onboarding flow.

- Mark the Jira issue as **Completed**.



Key Features for Jira Integration

1. Custom Dashboards:

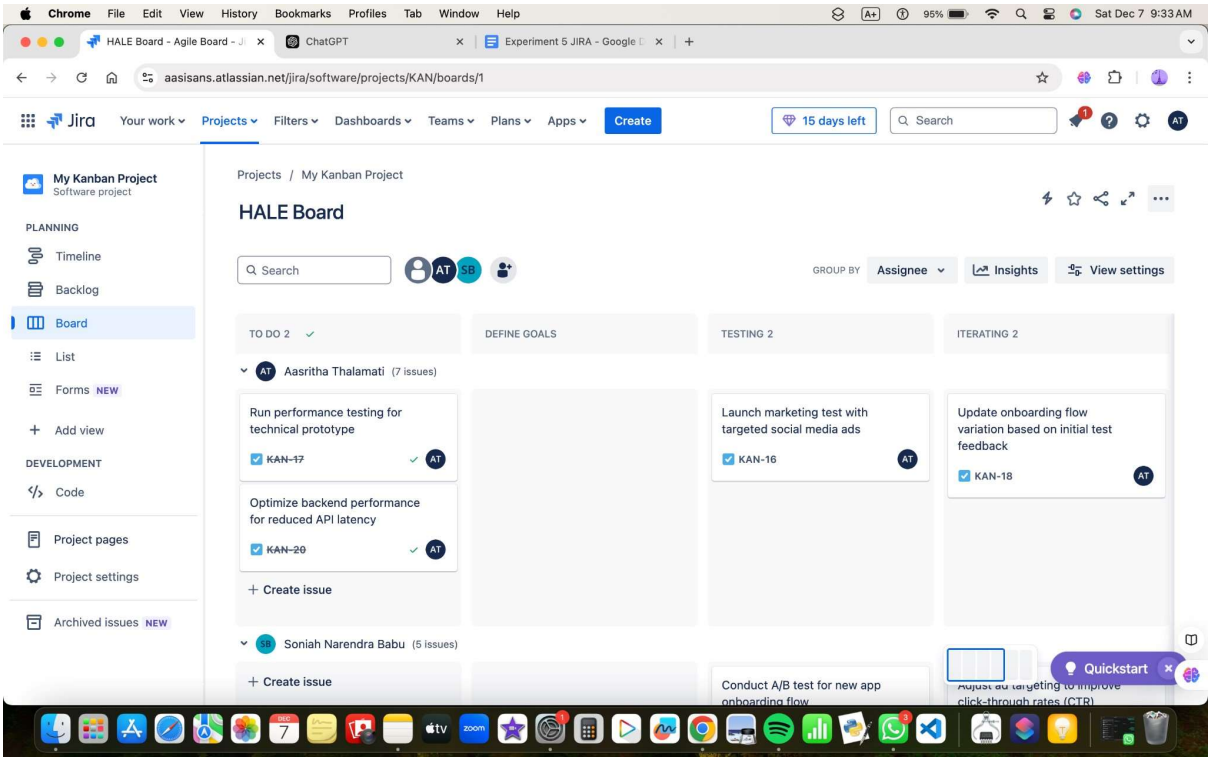
- Use Jira dashboards to track metrics like:
  - Status of experiments.
  - Key metrics over time (retention, feedback).

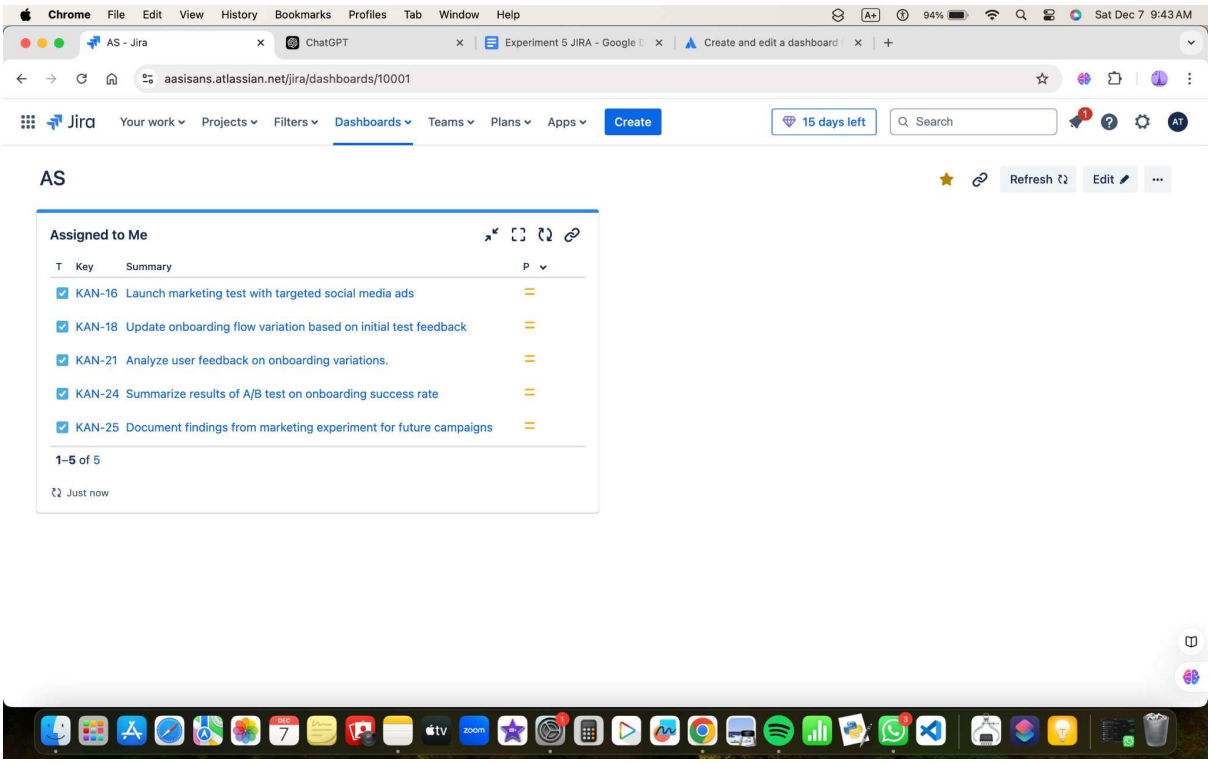
2. Automation:

- Set up Jira automation rules to:
  - Notify stakeholders when experiments move to the "Analyze and Learn" stage.
  - Automatically update the status based on task completion.

3. Reporting:

- Generate reports to summarize findings and decisions for completed experiments.

- 

- 

Benefits of This Hybrid Agile Approach

1. **Flexibility**: Adapts to different experiment types and team workflows.
2. **Focus on Learning**: Ensures quick validation of ideas to minimize waste.
3. **Transparency**: All stakeholders can track experiments through Jira.

4. **Continuous Improvement**: Enables iterative refinement of ideas.