

In Jira or Trello, there are many automatic tasks, those an Agile team used to manage in a day. How is the idea to automate these.

We are going to automate the following four tasks performed in Jira.

- Automatically update Trello boards.
- Generate daily stand-up summaries and send them via Slack.
- Optimize resource allocation based on workload analysis.
- Parse natural language commands like “Move Task X to Done.”

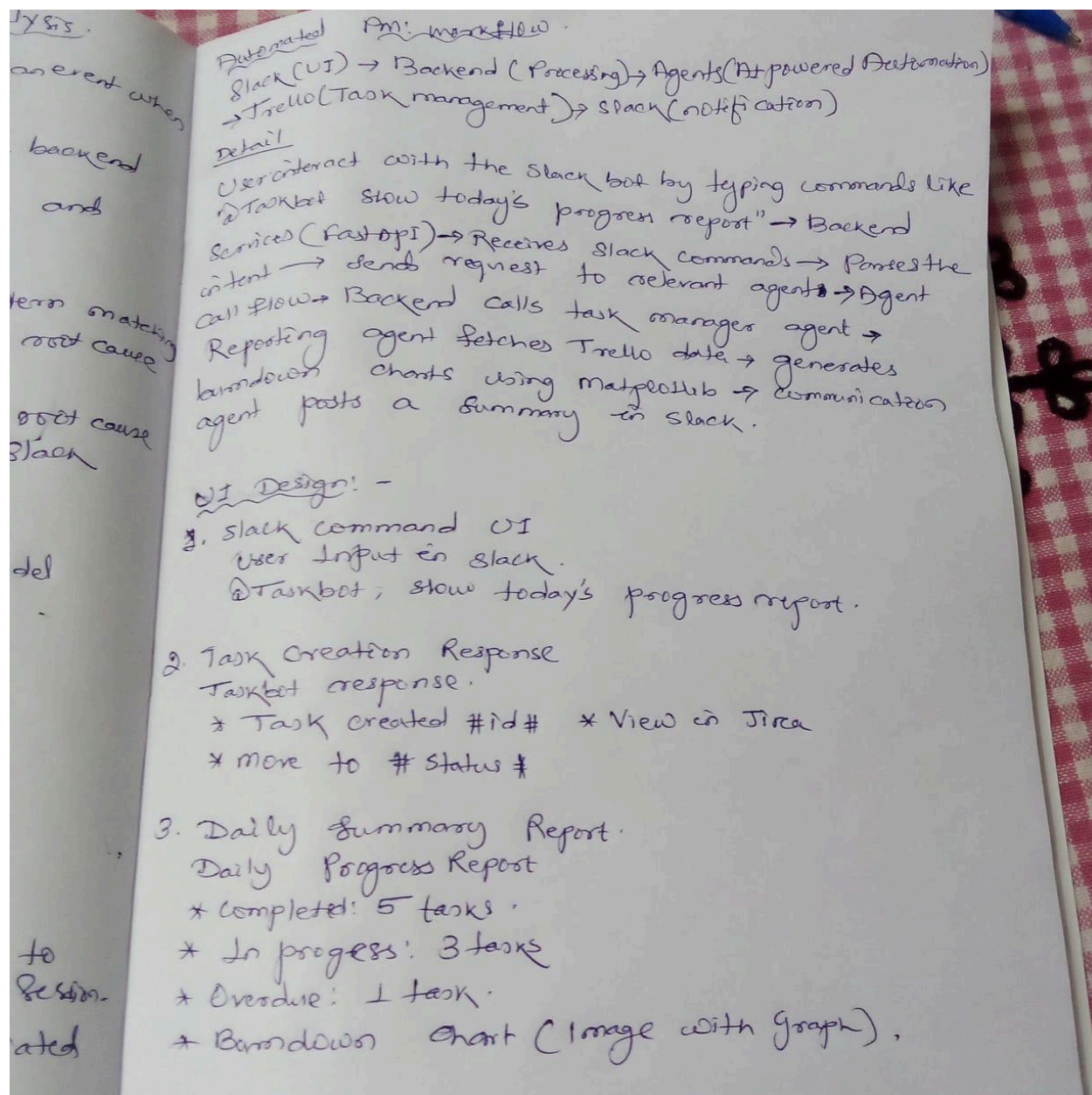
For these, the slack bot will perform the following tasks

1. Input Agent : Parses natural language commands from users (e.g., “Create a new task called ‘Fix Bug #82’”).
2. Task Manager Agent : Updates Trello boards based on parsed commands.
3. Reporting Agent : Generates burndown charts and daily summaries.
4. Communication Agent : Sends updates to Slack channels.

Tools used

1. Python 3.8 or higher.
2. A Trello account and API key/token.
3. A Slack workspace and bot token.
4. CrewAI for orchestrating AI agents while managing complex workflow
5. Flask

Workflow



Code

https://github.com/Suchi-BITS/PM/blob/main/Codes/ProjMan/Automated_PM.ipynb

Setting up AI agents in Slack

Here's a step-by-step guide to creating your own AI agent in Slack:

1. Create your app

- Navigate to "Your Apps" in Slack

- Click "Create New App" and select "From scratch"
- Name your app and select your development workspace
- Click "Create App"

2. Configure the Assistant View

- Find the "App Assistant View" section in the left navigation
- Enable the new container
- Add an overview that will appear in the container
- Choose between fixed or dynamic prompts
- Save your settings

3. Set up Socket Mode

- Enable Socket Mode for testing purposes (Note: This isn't recommended for production environments)
- Generate an app-level token when prompted
- Ensure the connections:write scope is added
- Save your generated token for later use

4. Configure permissions

- Go to "OAuth & Permissions"
- Verify that assistant:write scope is automatically added
- Add additional required scopes:some text
 - im:history
 - chat:write

5. Set up event subscriptions

- Enable Events in the Event Subscriptions section
- Subscribe to necessary bot events:some text
 - assistant_thread_started
 - assistant_thread_context_changed

- message.im
- Save your changes

6. Install and test

- Navigate to the Install App section
- Click "Install to Workspace"
- Grant necessary permissions
- Save the bot token provided after installation
- Check your Slack client to verify the app appears