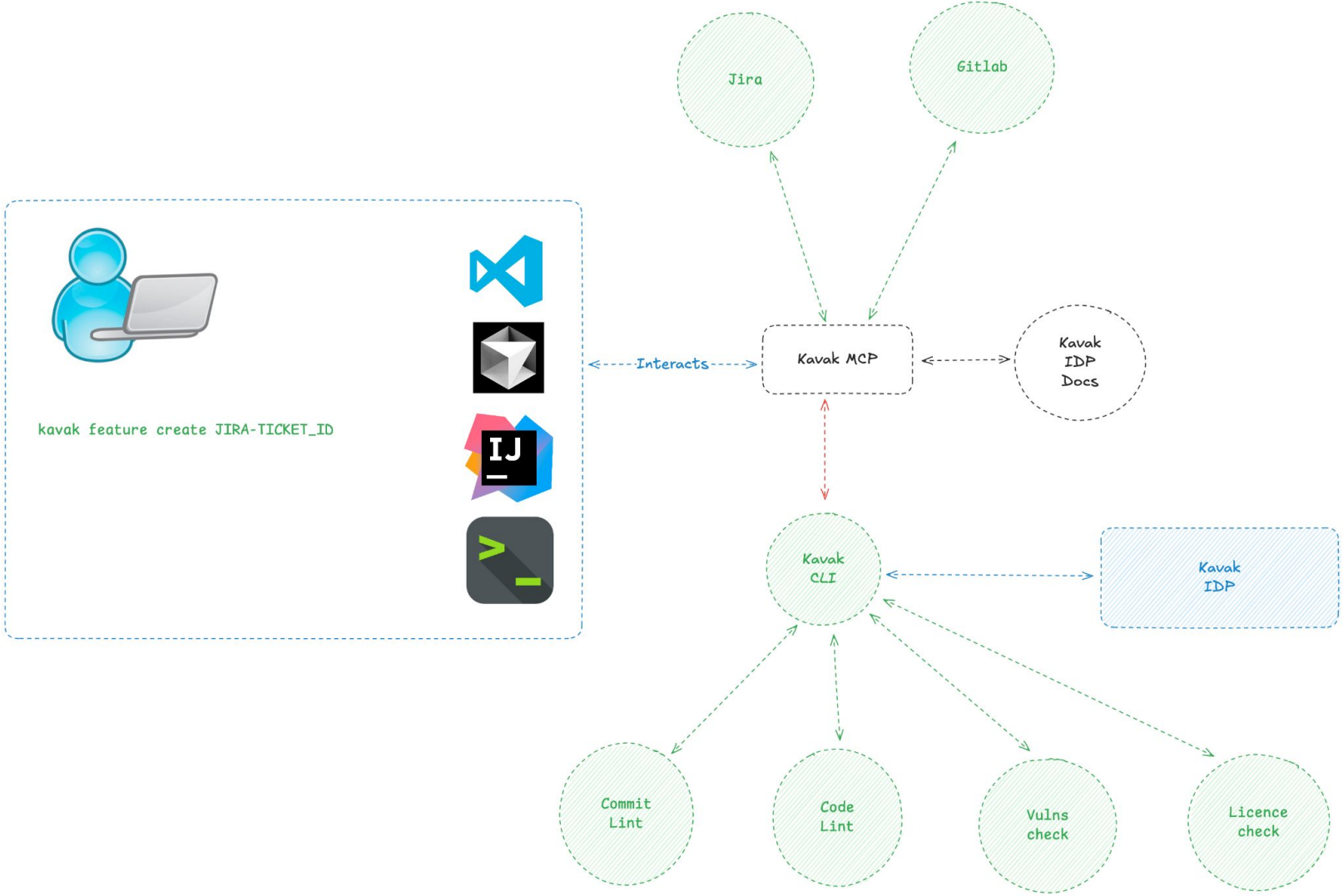
MCP Integration

SDLC

- Design a MCP tool to validate and create a quick security assessment and create a feature definition for MCP servers (similar to .cursor/rules.md)
- Create pre-commit integrations for SDLC inside Kavak
- Create pre-push integrations for SDLC inside Kavak
- Enable MCP functionality cross-organization.


Jira
Gitlab
kavak feature create JIRA-TICKET_ID
Interacts
Kavak MCP
Kavak IDP Docs
Kavak CLI
Kavak IDP
Commit Lint
Code Lint
Vulns check
Licence check

I'd like to create a new feature based from JIRA-TICKET_ID

Checks for Ticket Definition in Jira

Validates Feature Definition from OWASP Perspective

Create security assesment for the user and update the user story

kavak feature create JIRA-TicketID --definition json

```
git checkout main
git checkout -b feat/ticket_id
echo .rules/current_feature.md
```

Software development Life Cycle

kavak feature commit

pre-commit hooks

kavak feature push

pre-push hooks

Idea general

Con base en la definición de un feature, hacer un self-assesment de OWASP para determinar lo siguiente:

- Criticidad del feature
- Vulnerabilidades más conocidas
- Consideraciones de PII o segregación de accesos a tener en cuenta
- Generar un Checklist de tareas para que se complemente al proceso de desarrollo

Empty markdown cell, double-click or press enter to edit.

```python
from openai import OpenAI
from dotenv import load_dotenv
import json
import re

load_dotenv()

client = OpenAI()
```

[12] Python

{} feature_definition.json > {} feature

```json
{
    "project_name": "Todo List",
    "feature": {
        "jira_issue_key": "TODO-123",
        "name": "Multitenant todo list",
        "description": "A todo list that can be used by multiple tenants",
        "type": "backend-api",
        "user_stories": [
            "As a user, I'd like to login using Google OAuth",
            "As a user, I can create/edit/delete/view a todo list",
            "As a user, I can filter todos by tags and due date"
        ],
        "acceptance_criteria": [
            "The todo list is created/edited/deleted/viewed successfully",
            "The todo list is filtered by tags and due date successfully",
            "The todo list is created/edited/deleted/viewed successfully",
            "The login is successful and the user is redirected to the todo list"
        ],
        "non_functional_requirements": {
            "security": [
                "The todo list is protected by OAuth",
                "The todo list is protected by RBAC",
                "The todo list is protected by rate limiting",
                "The todo list is protected by logging"
            ]
        }
    }
}
```

```python
feature_definition = json.loads(open("feature_definition.json", "r").read())

response_feature_definition = client.responses.create(
    model="gpt-4o-mini",
    input=f"""Respond only in JSON format, following the schema provided. ["question1", "question2", "question3", "question4", ...]
    You are a security expert, and you are given a feature description. You need to assess the feature and determine the following:
    * Common security vulnerabilities that could be present in the feature
    * Common security best practices that could be applied to the feature
    * Common security mitigations that could be applied to the feature
    * Common security tools that could be used to assess the feature
    * Common security metrics that could be used to measure the feature
    * Common security standards that could be applied to the feature
    Provide 3 questions to assess the feature, and provide them in the JSON format.
    ```json
    {feature_definition}
    ```
    """
)
```

[3] ✓ 2.9s Python

```
partial_response = re.sub("```json", "", response_feature_definition.output_text)
partial_response = re.sub("```", "", partial_response)

questions = json.loads(partial_response)
```

[6] ✓ 0.0s Python

```
answers = []

for question in questions:
    answer = input(f"Question: {question}\nAnswer: ")
    answers.append(answer)

print(answers)
```

[ ] Python

... ["We're running inside AWS and using Internal developer platform, all DBs are Isolated and Security groups are only available by their respective pods", 'Using TLS, all cc

```
question_answers = [f"{question}\n{answer}" for question, answer in zip(questions, answers)]
print(question_answers)
```

[26] Python

... ["What measures are in place to ensure data isolation between tenants to prevent unauthorized access?\nWe're running inside AWS and using Internal developer platform, all

+ Code + Markdown

```python
response_feature_modification = client.responses.create(
    model="gpt-4o-mini",
    input=f"""Respond only in JSON format, following the schema provided. ["user_story_1", "user_story_2", "user_story_3", ...]
    You are a security expert, and you are given a feature description. You need to assess the feature and determine the following:
    * From the given questions/answers, enrich the user stories with enhanced security considerations based on the user's reponse prov
    ```json
    original_feature_definition:
    {feature_definition}
    question_answers:
    {question_answers}
    ```
    """
)
```

[33] Python

```python
print(response_feature_modification.output_text)
```

[34] Python

```json
[
    {
        "user_story": "As a user, I'd like to login using Google OAuth",
        "security_considerations": [
            "Ensure OAuth tokens are securely stored and transmitted using TLS encryption.",
            "Regularly review and audit OAuth integration and access logs to identify unauthorized attempts.",
            "Implement session expiration and logout mechanisms to protect against token theft."
        ]
    },
    {
        "user_story": "As a user, I can create/edit/delete/view a todo list",
        "security_considerations": [
            "Implement role-based access control (RBAC) to ensure only authorized users can perform create/edit/delete operations.",
            "Validate input on all API endpoints to prevent SQL injection and XSS vulnerabilities.",
            "Use tenant-specific isolation mechanisms to ensure that users cannot access each other's todo lists."
        ]
    },
    {
        "user_story": "As a user, I can filter todos by tags and due date",
        "security_considerations": [
            "Validate filter parameters to prevent injection attacks and ensure they conform to expected formats.",
            "Monitor logging for filtering actions to detect any abnormal patterns that may indicate misuse.",
            "Rate limit filtering actions to mitigate the risk of abuse and protect the backend service."
        ]
    }
]
```