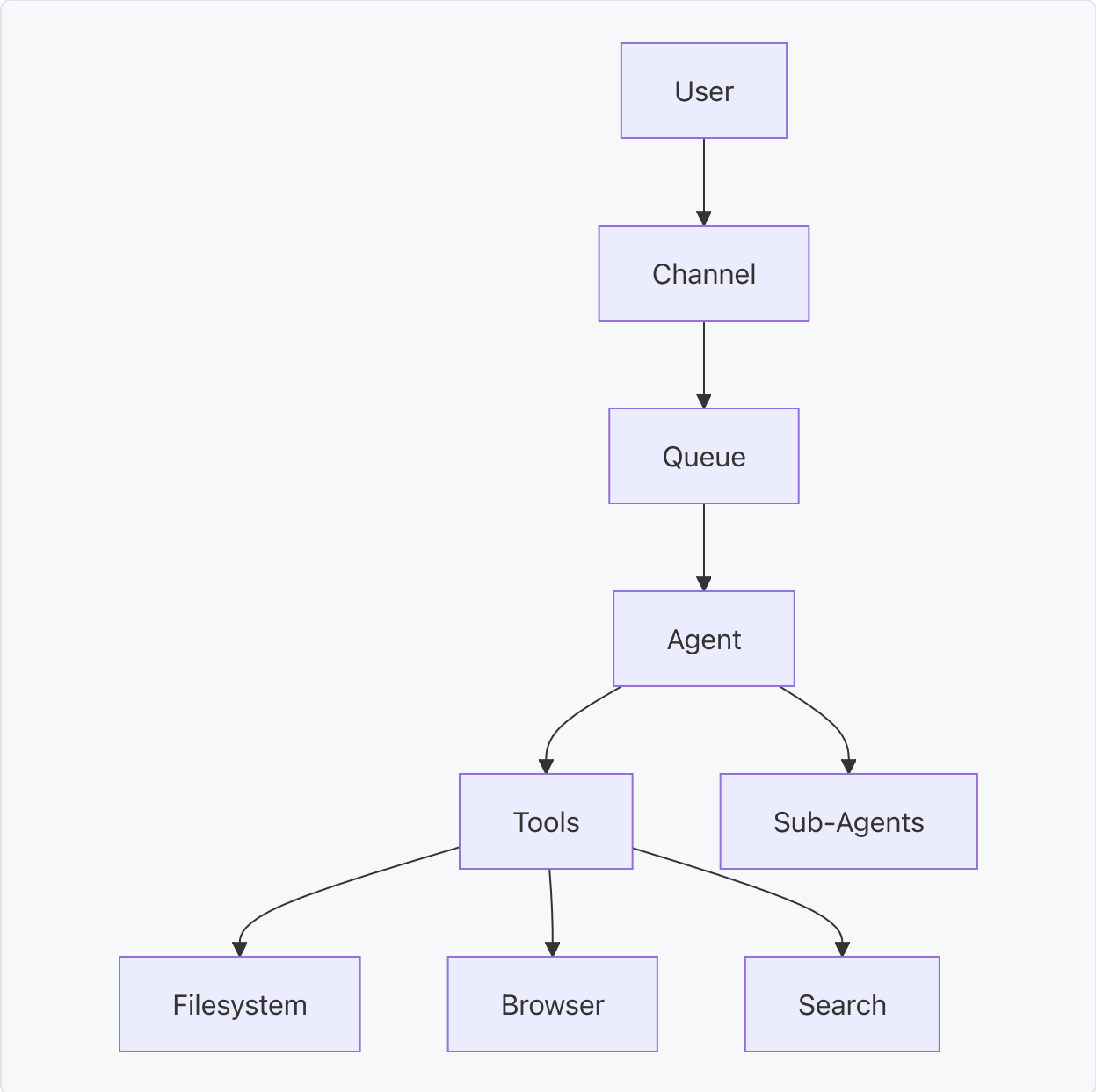
OpenPaw Architecture Report

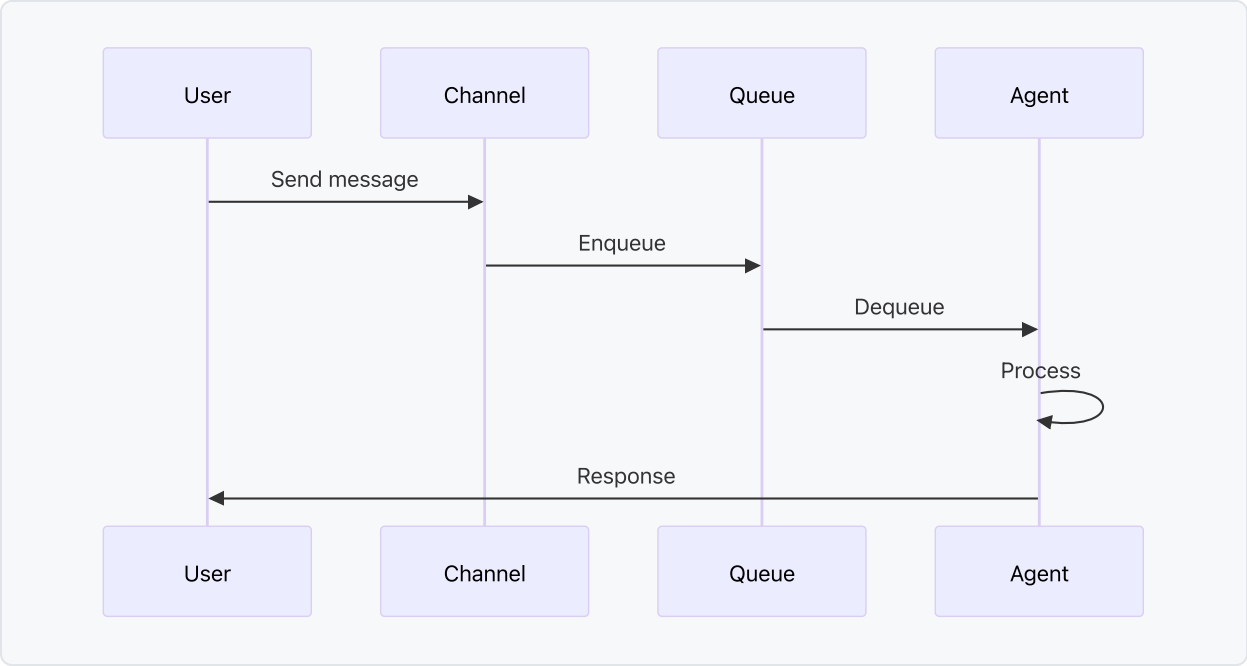
This report demonstrates the md2pdf rendering service with mermaid diagrams, code blocks, and tables.

System Architecture



Message Flow

The following sequence diagram shows a typical message exchange:



Code Example

```python
async def process_message(message: Message) -> str:
    """Process an incoming message through the agent."""
    agent = await create_react_agent(
        model=model,
        tools=tools,
        system_prompt=system_prompt,
    )
    result = await agent.ainvoke({"messages": [message]})
    return result["output"]
```

Component Summary

Component	Purpose	Status
AgentRunner	Wraps LangGraph ReAct agent	Active
QueueManager	Lane-based FIFO queue	Active
SessionManager	Conversation thread tracking	Active
CronScheduler	APScheduler integration	Active
BrowserSession	Playwright automation	Active

Key Metrics

- **870+** tests passing
- **7** completed sprints
- **11** browser automation tools
- **8** max concurrent sub-agents

> *"The framework is designed for autonomous, proactive agent behavior with human-in-the-loop safety gates."*

Conclusion

OpenPaw provides a robust foundation for building autonomous AI agents with channel-agnostic communication, persistent memory, and comprehensive tool access.