Generative AI Agents for Minecraft: Automating Creative World Tasks with Amazon Bedrock

Sprint 1 Deliverable

Team Members

Taeesh Azal Assadi
Sanjna Kumari
Faraz Ulhaq Shah

March 22, 2025

1 Demo working product system

Current Implementation Video: Sprint 1 Demo - Local Mineflayer Integration
        Password: `m7d^6.Hh`

Core Command System:
    - Functional command handler in src/main.ts:

```typescript
case 'dance': // New custom command
  mcBot.chat('Let me show you my moves!');

  // Dance sequence
  for (let i = 0; i < 3; i++) {
    // Jump
    mcBot.setControlState('jump', true);
    await new Promise(resolve => setTimeout(resolve, 200)); // Jump for 200ms
    mcBot.setControlState('jump', false);

    // Move Left
    mcBot.setControlState('left', true);
    await new Promise(resolve => setTimeout(resolve, 300)); // Move Left for 300ms
    mcBot.setControlState('left', false);

    // Move right
    mcBot.setControlState('right', true);
    await new Promise(resolve => setTimeout(resolve, 300)); // Move right for 300ms
    mcBot.setControlState('right', false);

    // Look around
    mcBot.look(mcBot.entity.yaw + 45, 0); // Look 45 degrees to the right
    await new Promise(resolve => setTimeout(resolve, 200)); // Wait 200ms
    mcBot.look(mcBot.entity.yaw - 90, 0); // Look 90 degrees to the left
    await new Promise(resolve => setTimeout(resolve, 200)); // Wait 200ms
    mcBot.look(mcBot.entity.yaw + 45, 0); // Look back to the center
  }

  mcBot.chat('That was fun!');
  break;
```

Local Testing Framework:
    - Validation of basic AI commands without AWS dependencies:

2 Scrum

2.1 Sprint 1 backlog and burndown chart

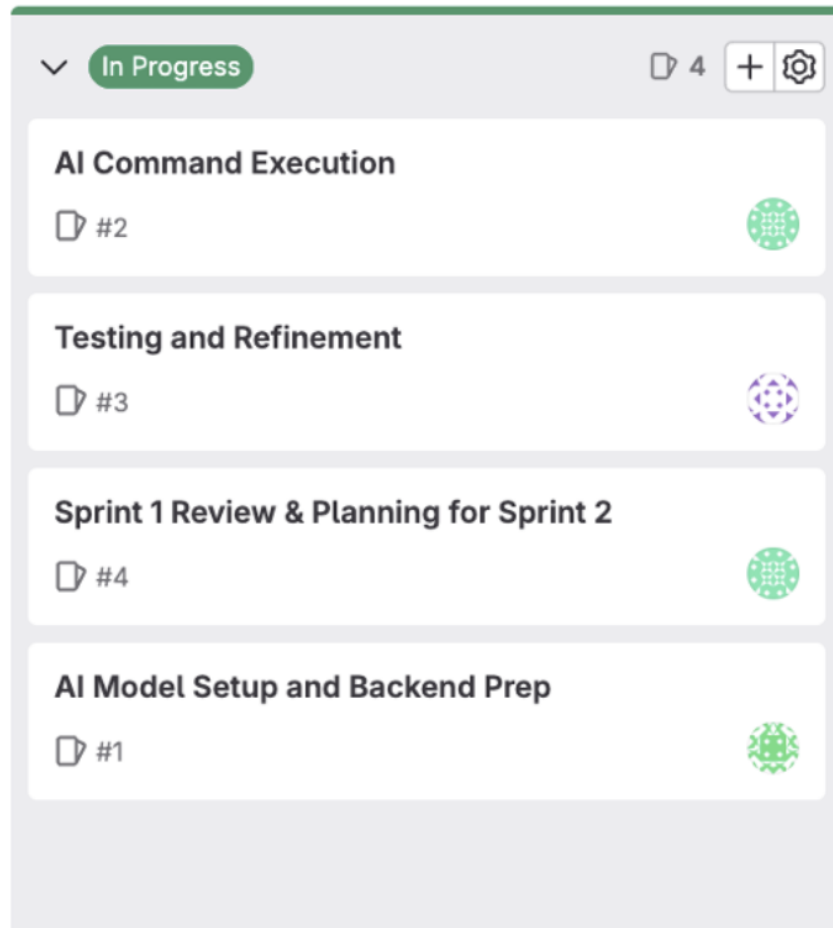**Sprint 1 Duration: Weeks 5–8 (Feb 24 – Mar 23, 2025)**

Figure 1: Key Focus: Foundational AI Setup & Basic Feature Implementation

Objectives and Tasks:

- **Week 5** – AI Model Setup & Backend Prep (AWS Bedrock, API, CloudFormation Setup).

- **Week 6** – AI Command Execution (Basic structure generation & connection to Mineflayer).

- **Week 7** – Testing & Refinement (AI-generated builds optimization).

- **Week 8** – Sprint 1 Review & Planning for Sprint 2.

Features Delivered:

- Quick World Generation for Server Owners.

- Basic AI Command Execution (Build Simple Structures).

- Backend Setup using Mineflayer + YAML CloudFormation Template.

- Local testing of the Minecraft bot with manual commands.

Burndown Chart Overview:

The attached burndown chart shows a healthy and consistent decline in remaining tasks. Although initial progress slightly lagged due to AWS setup delays, we aligned with the planned timeline by week 7 through parallel tasking and efficient testing iterations.
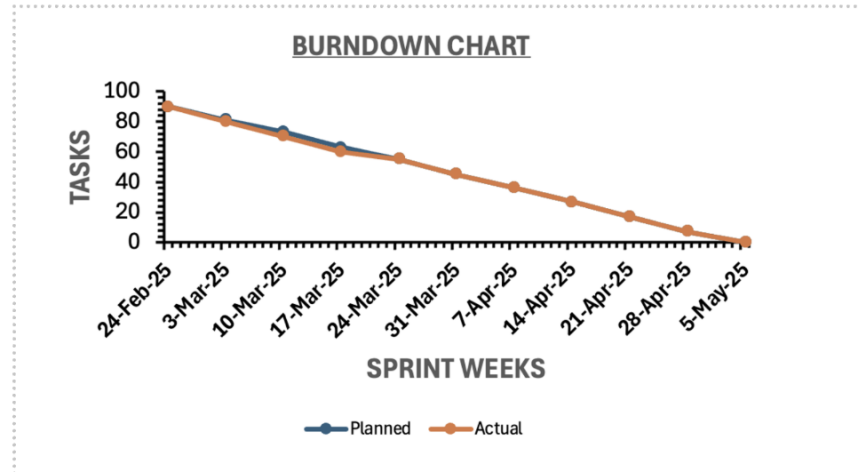

Figure 2: Burndown Chart

SPRINT WEEKS	Planned	Actual
24-Feb-25	90	90
3-Mar-25	81	80
10-Mar-25	73	70
17-Mar-25	63	60
24-Mar-25	55	55
31-Mar-25	45	45
7-Apr-25	36	36
14-Apr-25	27	27
21-Apr-25	17	17
28-Apr-25	7	7
5-May-25	0	0

Figure 3: Effort Estimation for next sprints based on Sprint 1 Progress (Total – 90 hours)

2.2 Kanban screenshots (beginning and end of the sprint)

The screenshots shown below are from Gitlab Kanban Board - **Gitlab Kanban Board**
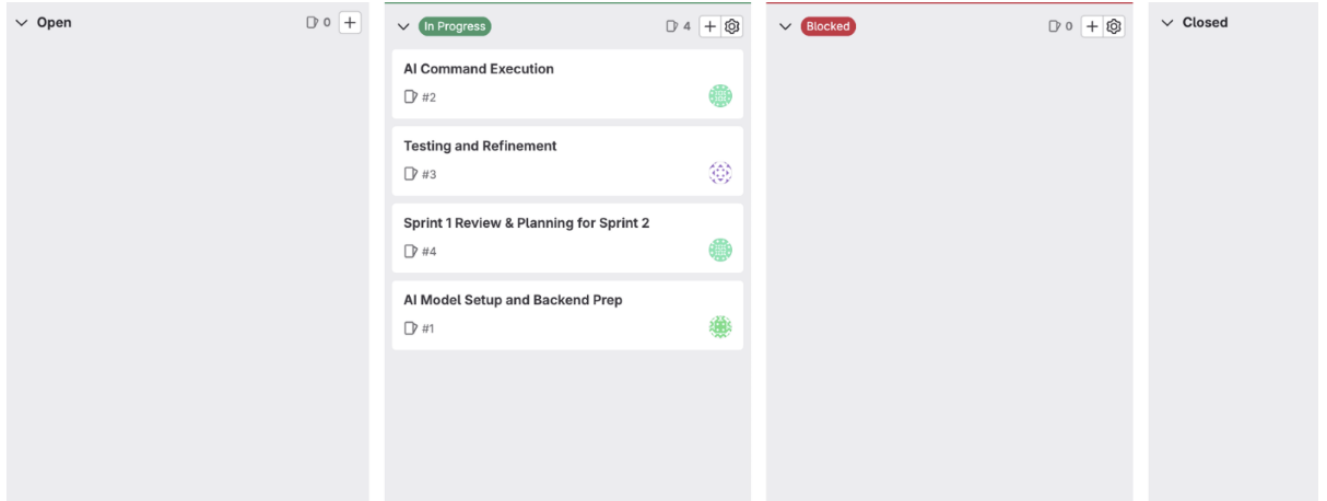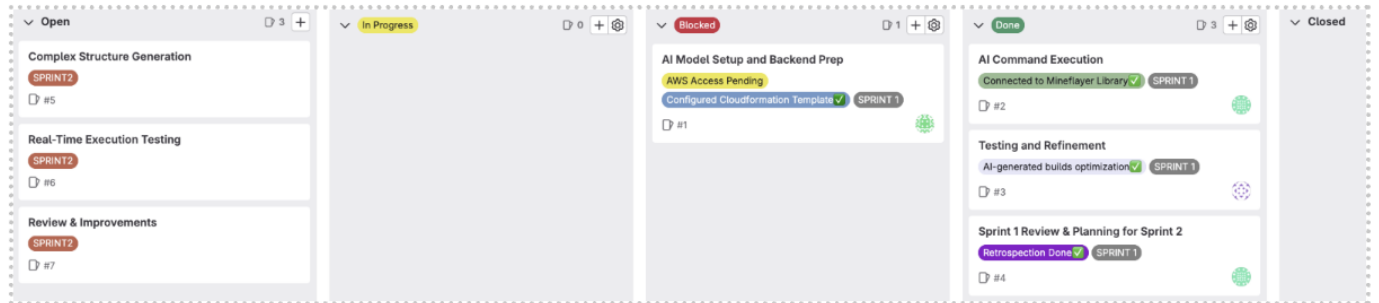

Figure 4: Before the start of Sprint 1

Figure 5: After the end of Sprint 1

2.3 Sprint retrospective

During Sprint 1, we were able to accomplish the majority of what we had planned. Our team successfully completed the backend setup, created the CloudFormation YAML template, and implemented basic AI command execution within Minecraft using the Mineflayer library. We also achieved local testing of the Minecraft bot by manually inputting commands, ensuring that the basic functionality and infrastructure were in place.

The only major task we could not complete was testing with Amazon Bedrock. This was due to a delay in receiving AWS resource access from WISE personnel. The delay was further extended because of the Spring Break period, during which communication slowed down. As a workaround, we proceeded with backend development and configuration in anticipation of receiving access post-break. We are now scheduled to receive access on Monday, March 24th, which will unblock the remaining testing tasks under "AI Model Setup and Backend Prep." (as seen in Kanban Board)

Additionally, while refining our architecture diagram, we decided to remove components like the S3 bucket from the current sprint scope. These were deemed more appropriate for future iterations. This decision was

finalized during an office hours call with WISE personnel, where we walked them through our architectural design and clarified the immediate needs for Sprint 1.

What went well:

- Strong team collaboration and communication.

- Successful implementation of foundational features.

- Valuable hands-on learning with Mineflayer and infrastructure setup.

What could be improved:

- More proactive and synchronous communication (e.g., scheduled calls) with WISE personnel would have helped us get AWS access quicker.

- Better anticipation of institutional delays around academic breaks.

Overall, Sprint 1 was productive despite minor external blockers. We are well-positioned to integrate Amazon Bedrock and continue development in Sprint 2.

2.4   Product Backlog (Updated)

As a result of Sprint 1, we only made one update to our product backlog based on implementation progress and team discussions. The change was the removal of logging user-AI interactions using an Amazon S3 bucket. Initially, the goal was to store these interactions for future analysis and improvement. However, after reviewing our current sprint goals and scope, we concluded that this functionality is not essential at this stage. Introducing S3 would increase the complexity of the system without immediate value, especially as the primary focus remains on building core AI features. We plan to revisit this feature in a future sprint if time and priorities allow.

2.5   Issue Tracking

Sprint 1 completeness and Sprint 2 backlog can be found on Gitlab with relevant labels here - **Gitlab**

2.6   Sprint 2 Backlog

Key Focus: Advanced AI & Automation (Weeks 9–11)

Tasks Breakdown:

- **Week 9** – Complex Structure Generation (AI generates detailed buildings & terrains).

- **Week 10** – Real-Time Execution Testing (Optimizing AI's response handling in AWS Bedrock).

- **Week 11** – Review & Improvements (Testing full structure & terrain generation, optimizing AI accuracy).

Expected Deliverables:

- AI Generating Structures in Minecraft.

- Real-Time AI Execution Testing Completed.

- Sprint 2 Deliverable (Builds & Real-Time Testing).

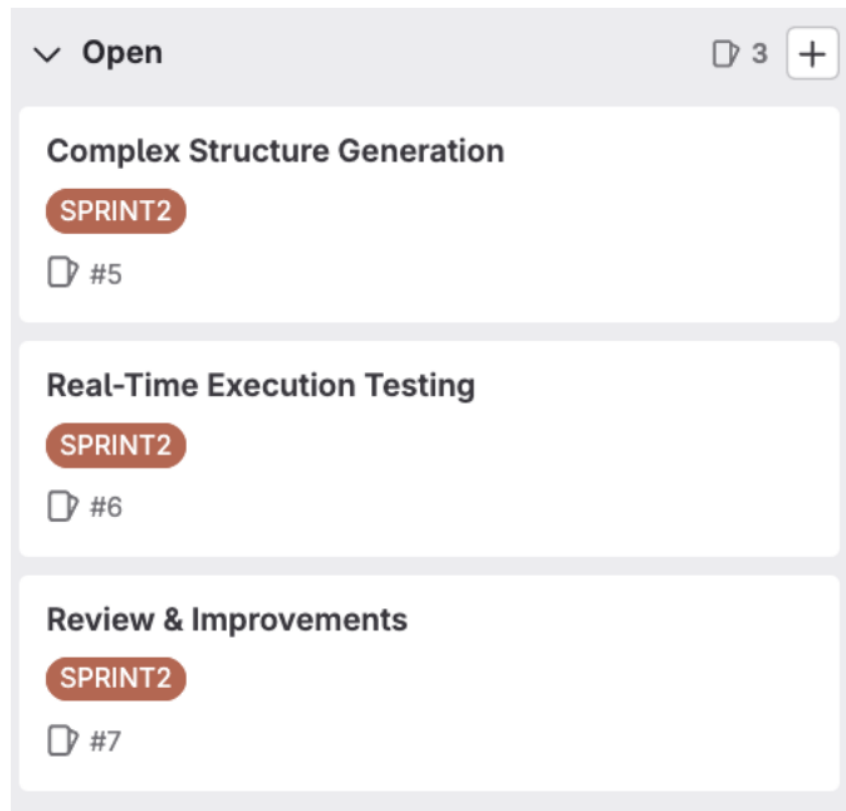**Below is Sprint 2 Backlog Screenshot from Gitlab Kanban Board:**

Figure 6: Sprint 2 Backlog

3 Cloud Deployment

Our project involves cloud development using AWS services, specifically Amazon Bedrock for AI model integration and CloudFormation for infrastructure provisioning. We have completed the necessary preparations, including setting up the backend and creating a CloudFormation YAML template. We recently received approval from WISE personnel, and we are scheduled to receive AWS access to the required resources on Monday, March 24, 2025. Once access is granted, we will proceed with full deployment and testing of our AI agent on the cloud environment.

Attached below is our AWS Architecture showing the resources we require:
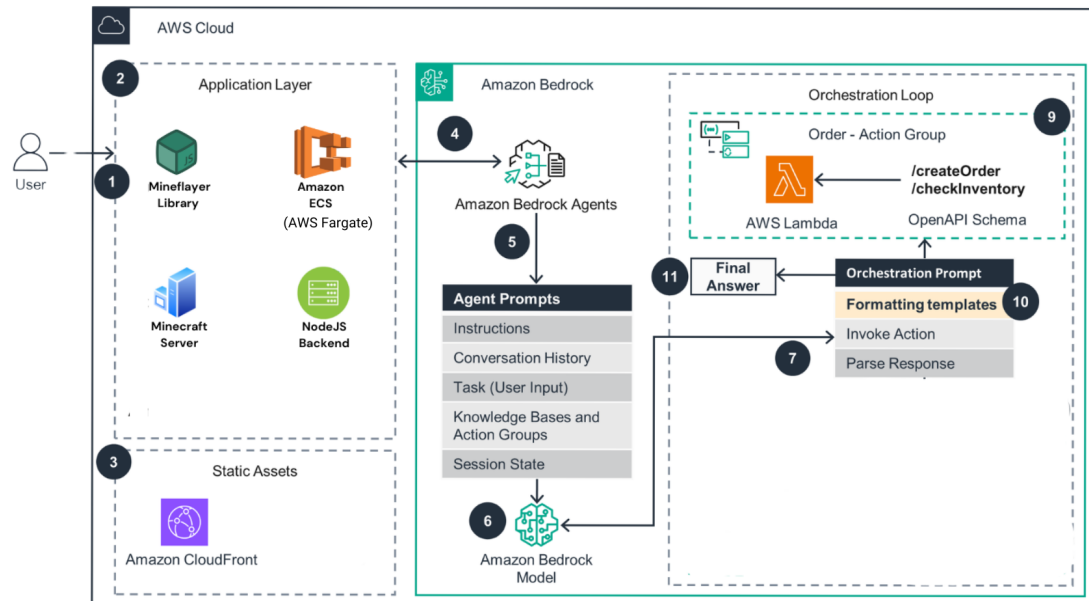

Figure 7: Architecture