**Azure DevOps Boards Lab: Building a Docker Image with Jenkins CI/CD & Implementing Agile Processes**

**Author**

**Course**

**Instructor**

**Affiliation**

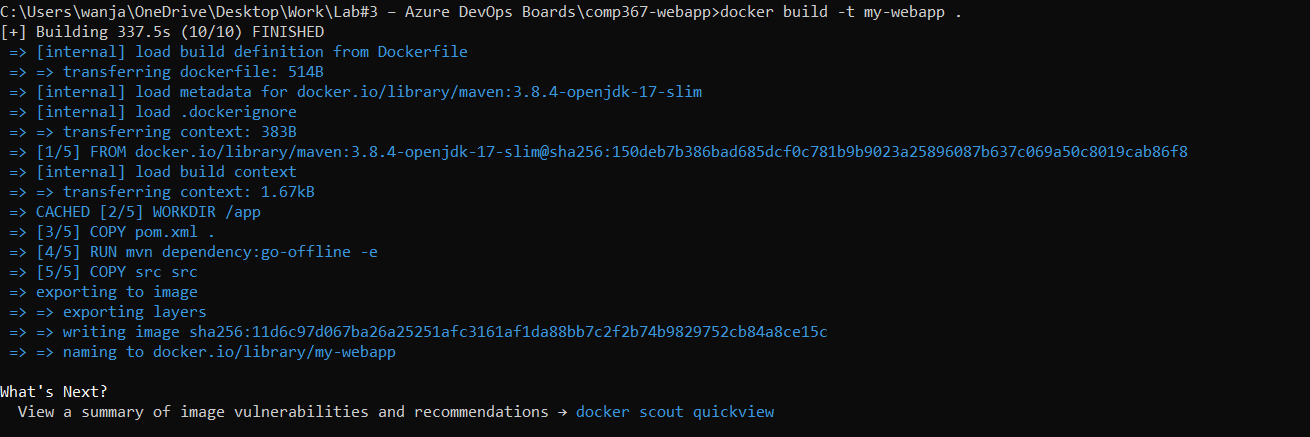
**Due Date**

**Question 1: Create Jenkin CI/CD Pipeline to Build Docker Image For A Maven Project**

In this project, we will use Jenkins to automate the process of constructing a Docker image for a Java web application that uses Maven, as part of a continuous integration and delivery (CI/CD) pipeline. With every change to the codebase being automatically produced into a new Docker image and submitted to a Docker registry for simple deployment, this project aims to optimize the development workflow.

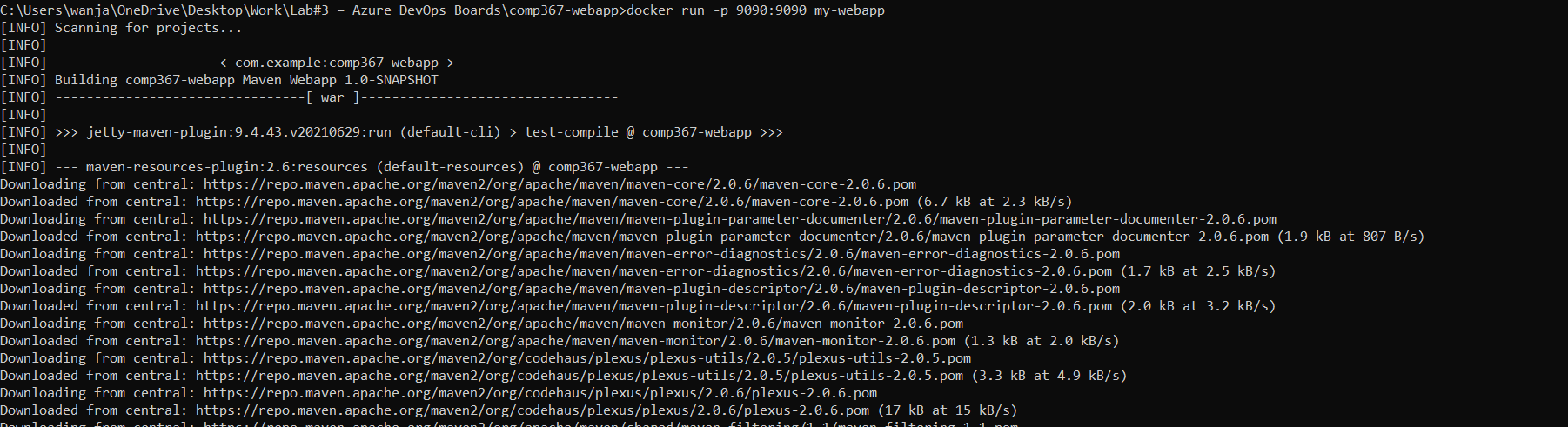
**Figure 1**

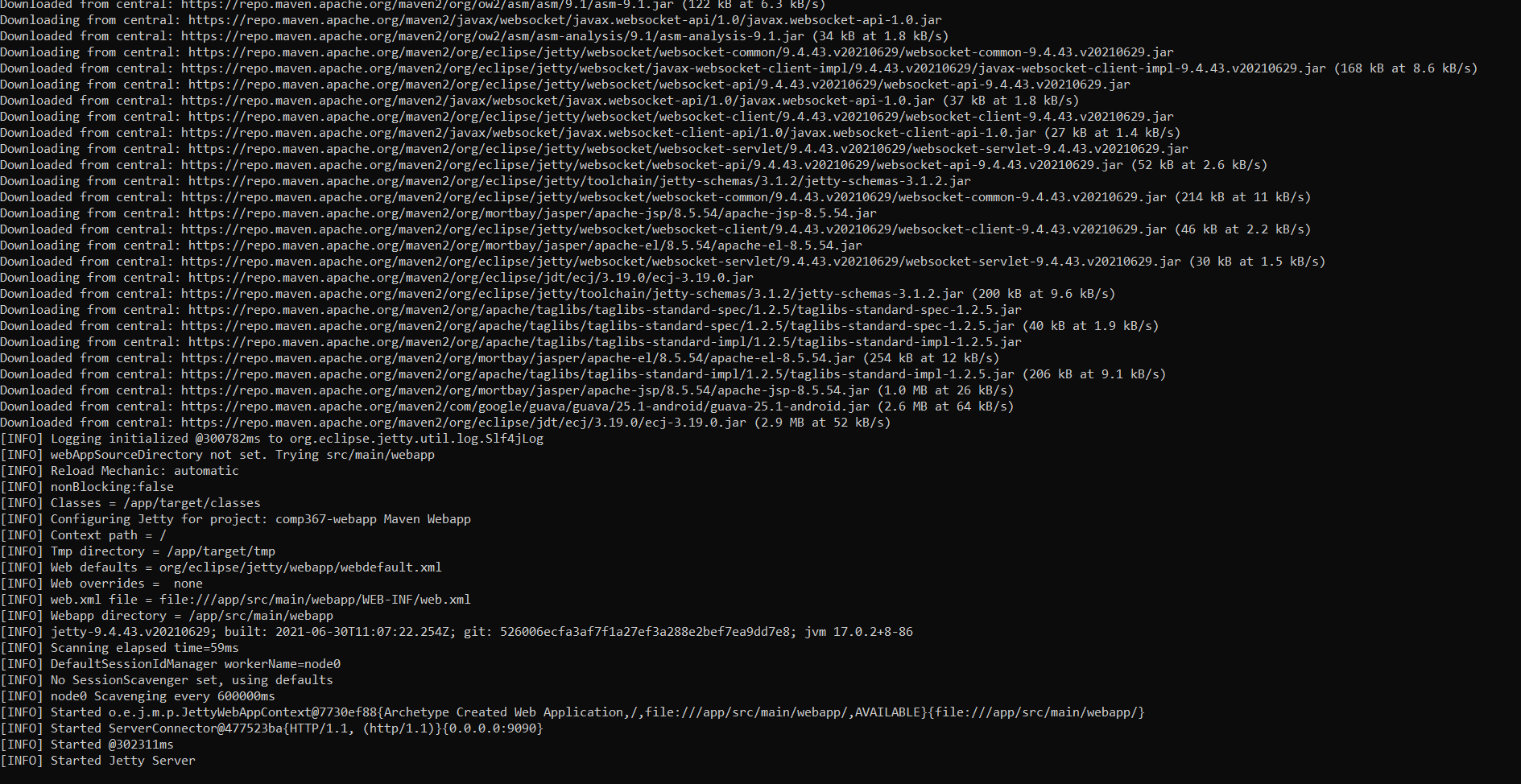
Screenshot to show that manually building docker image is successful



**Figure 2**

Provide screenshot to show that the generated image in step 2 can run properly [1 mark]



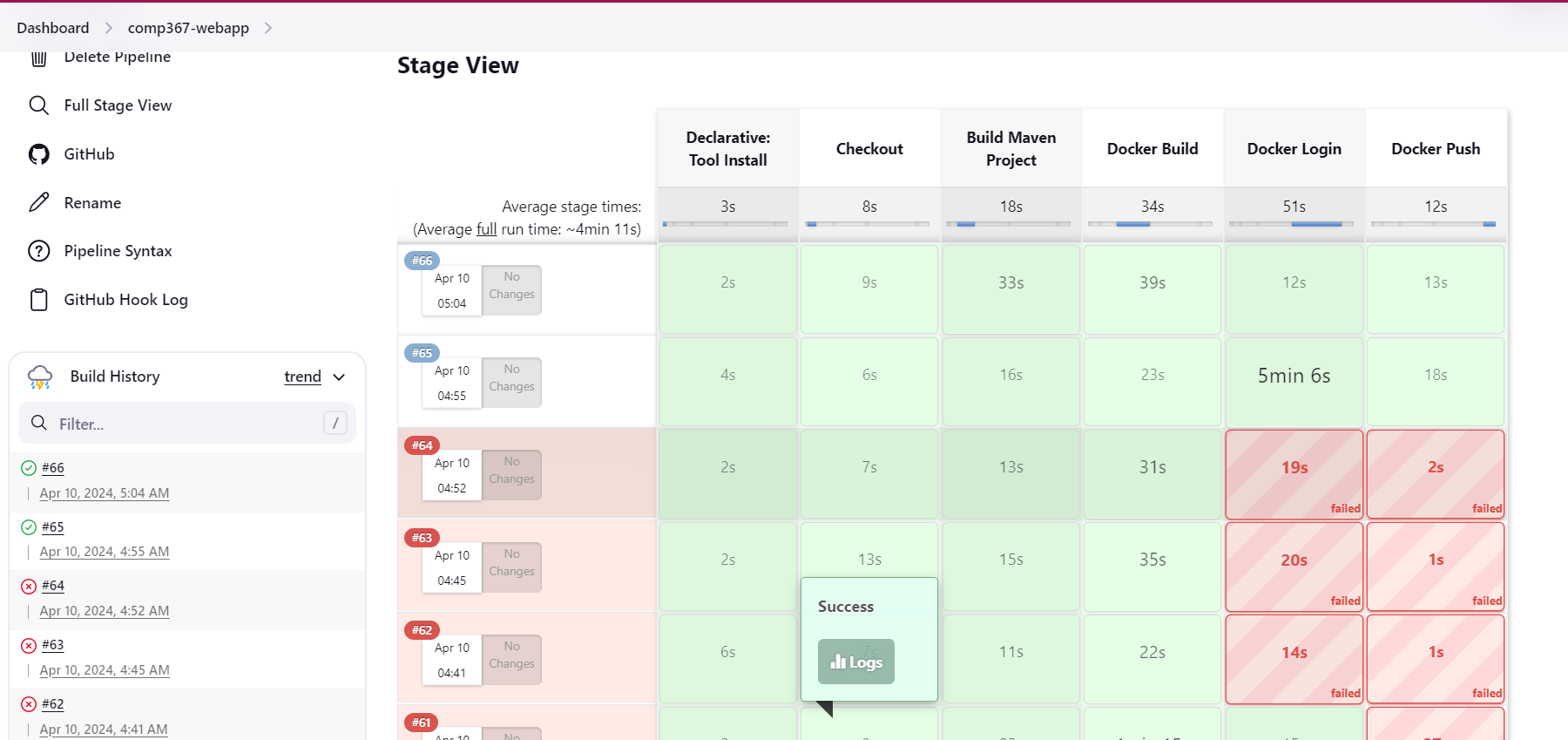


[**https://github.com/scriptsorcerer04/comp367-webapp**](https://github.com/scriptsorcerer04/comp367-webapp)

The various files, including the Dockerfile and Jenkins file, are available at this project's GitHub URL.

**Figure 3**

Screenshot of your pipeline build history



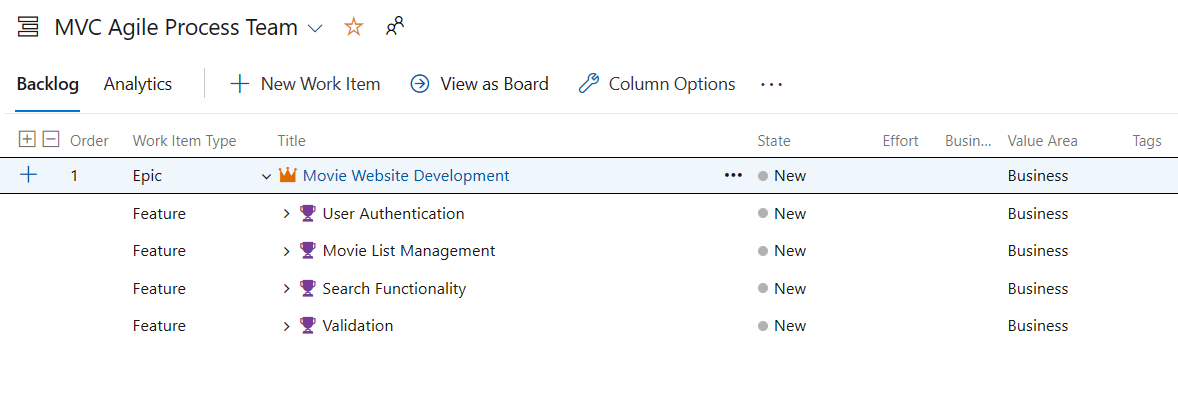
We have successfully automated the process of producing a Docker image for our Maven Java web application by implementing this CI/CD pipeline with Jenkins. By adhering to recommended procedures, such safely storing login credentials and versioning our code and pictures, we can guarantee an effective, dependable, and easily maintainable development workflow. We can react to changing requirements and provide value to users more quickly if we can deploy updated versions of our application with ease.

**Question Two: Use Agile Process in Azure Devops**

In this assignment, we use ASP.NET Core MVC as our main focus and use Azure DevOps to build Agile software development approaches. We create a project called "MVC Agile Process" and organize it using an Agile Work Item Process. The project's development requirements are arranged through the creation of Epics, Features, User Stories, and Tasks. We also set up two iterations within Sprints to add search and validation features and gradually build MVC functionalities

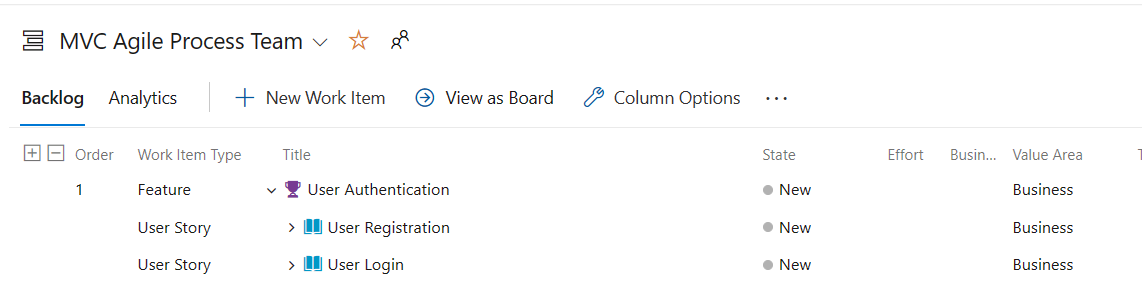
**Figure 1**

The main epic details with two child features



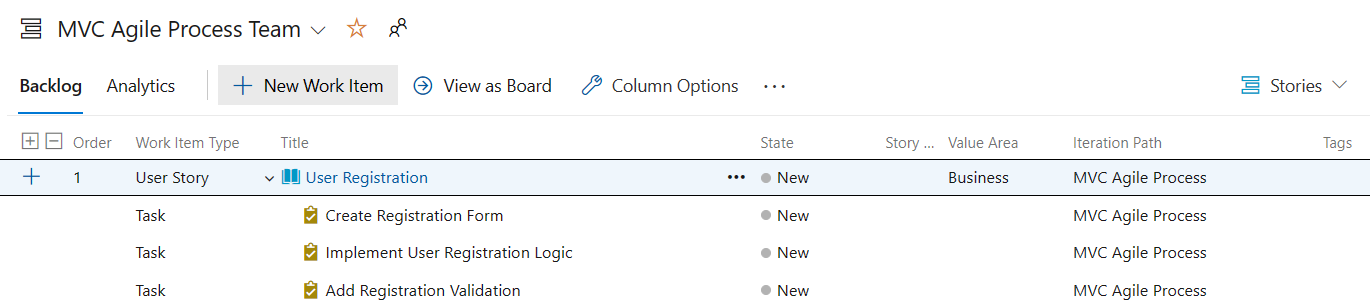
**Figure 2**

Each Feature with details that has child User Story



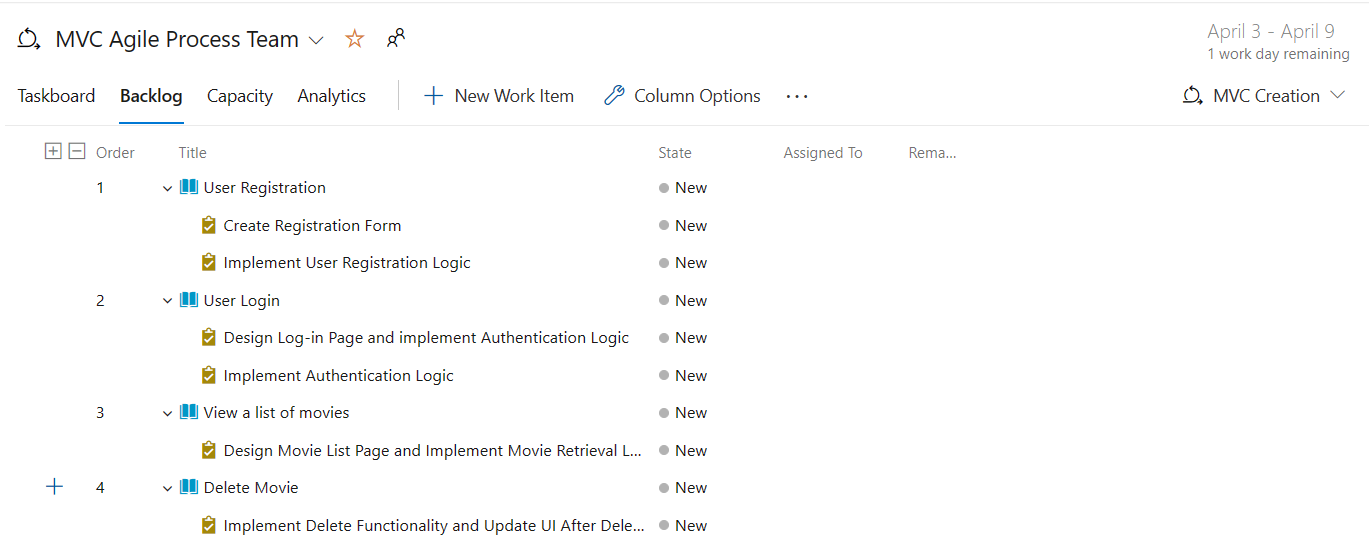
**Figure 3**

Each User Story with details that has child Tasks



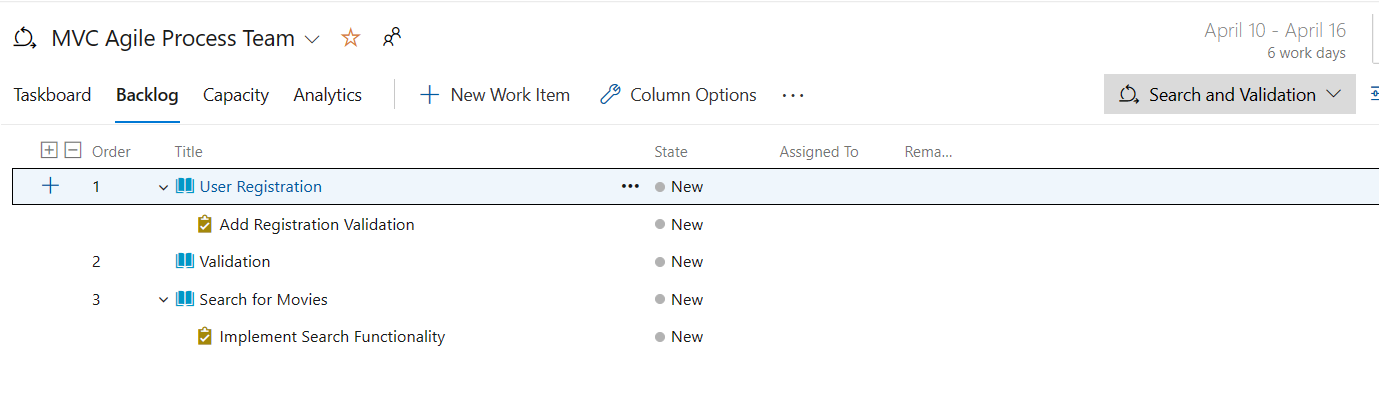
**Figure 4**

Iteration 1 with tasks, start and end date [3 marks]



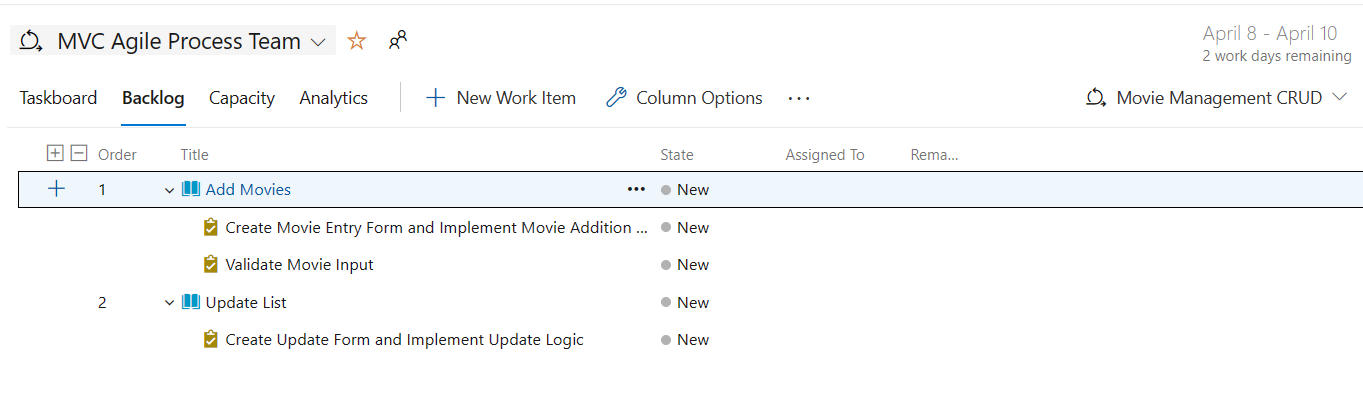
**Figure 5**

Iteration 2 with tasks, start and end date



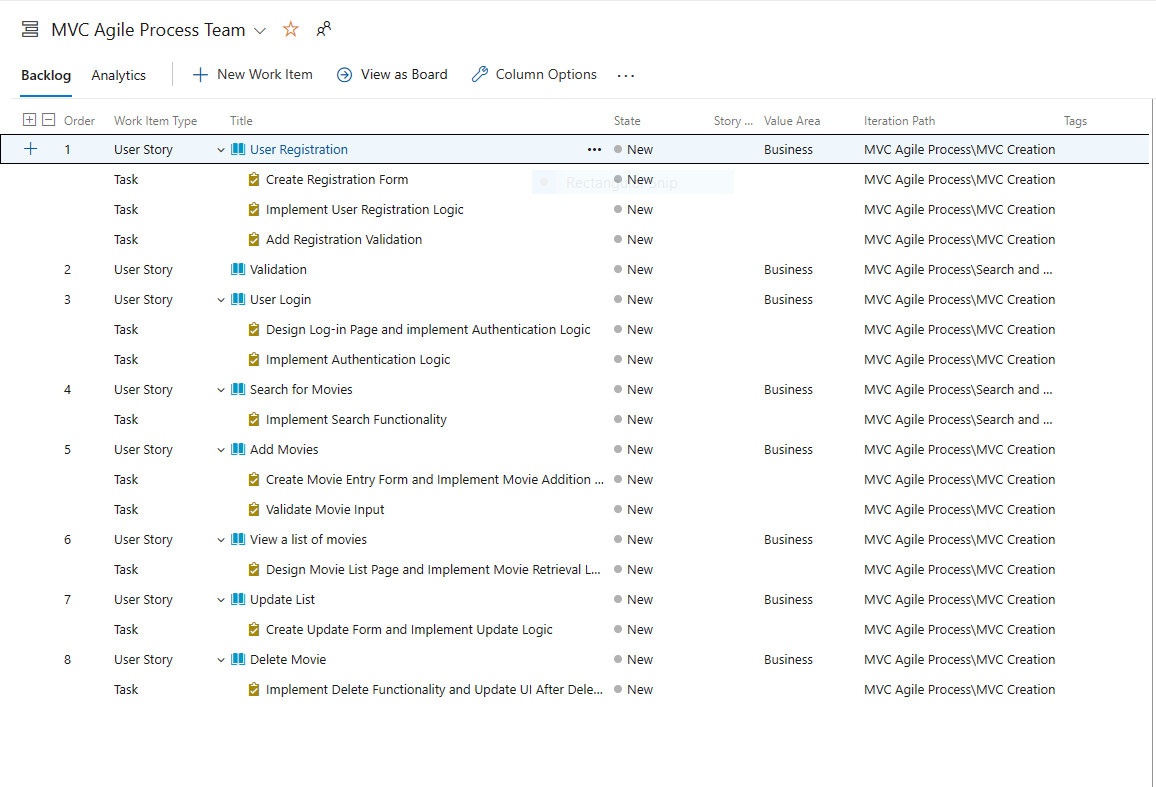
**Figure 6**

Task board showing current sprint details



**Figure 7**

Backlog showing User Stories and Features [2 marks]



The GitHub URL to the project is [scriptsorcerer04/MVC-Agile-Process (github.com)](https://github.com/scriptsorcerer04/MVC-Agile-Process)

This assignment demonstrates the real-world implementation of software development approaches by fusing Agile and Azure DevOps. The integration of Azure DevOps enables smooth project management and collaboration, while Agile approaches promote flexibility and response to evolving needs. The importance of Agile methodologies and organized project management in contemporary software development workflows is highlighted by this assignment.