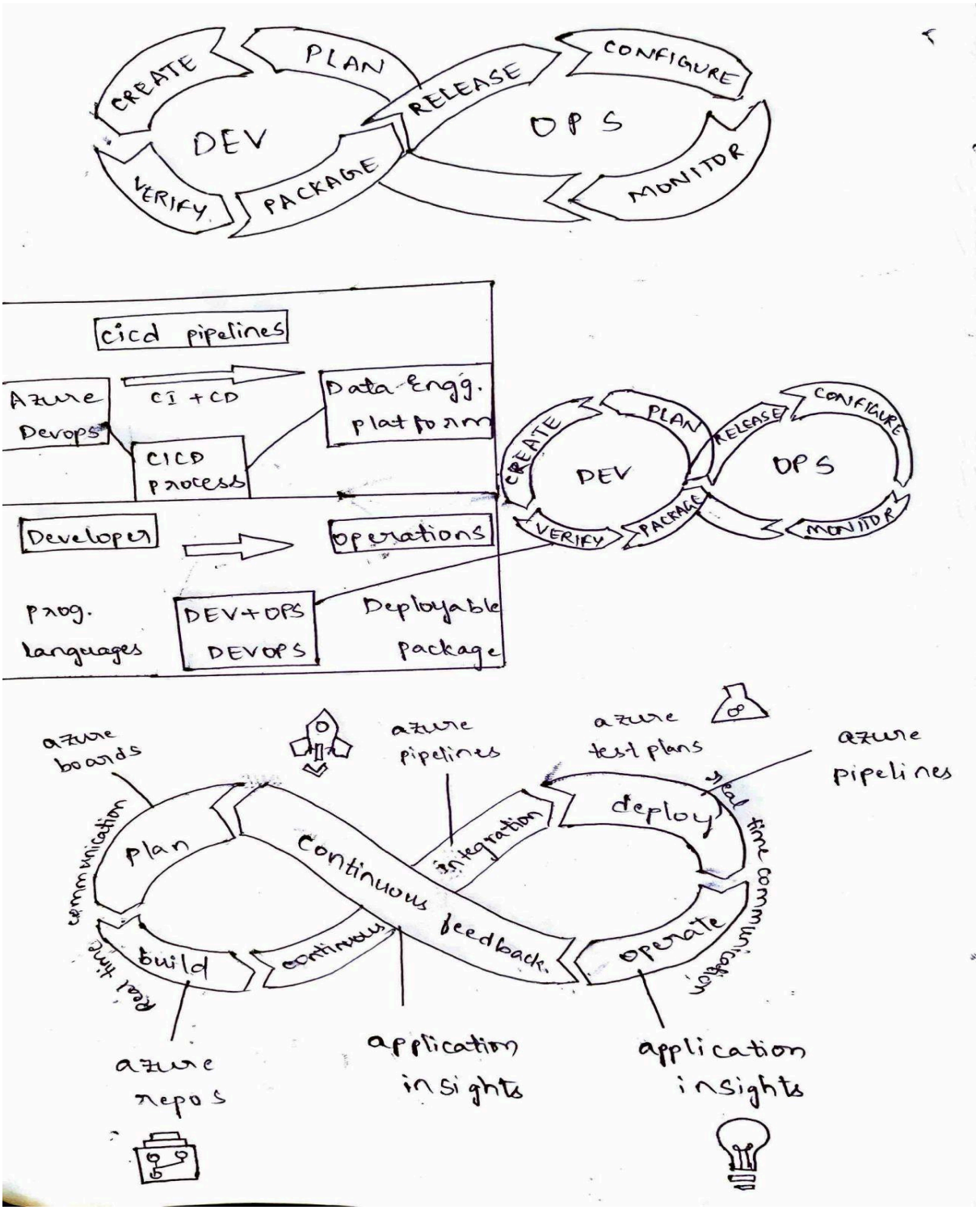
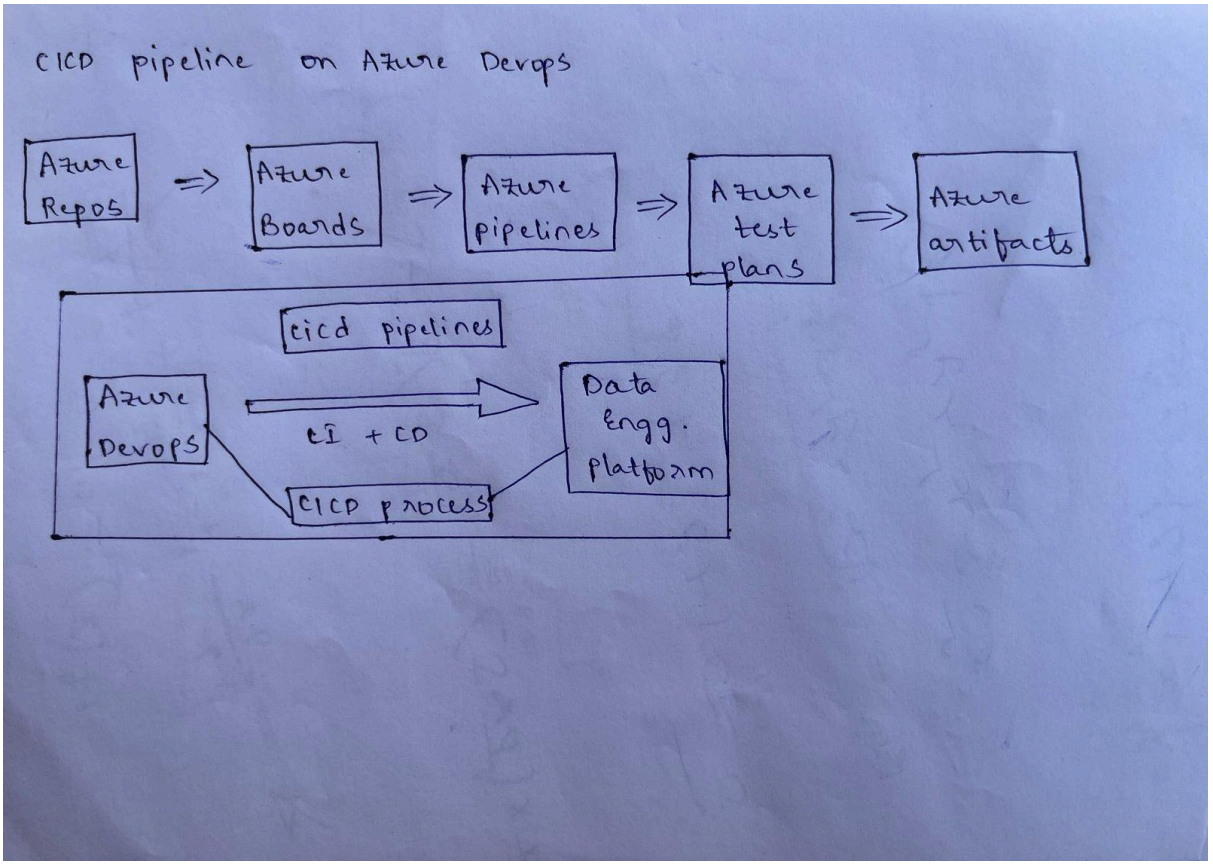
NAME : AKULA SHARATH CHANDRA
BATCH: DATA ENGINEERING
DATE :22-02-2024
TOPIC: AZURE DevOps

1)DevOps: DevOps is the union of people, process, and products to enable continuous delivery of value to your end users.





INTRODUCTION TO AZURE DevOps:

i)Azure Boards:

Track work with Kanban boards, backlogs, team dashboards, and custom reporting

→**Connected from idea to release:**

Track all your ideas at every development stage and keep your team aligned with all code changes linked directly to work items.

→**Scrum ready:**

Use built-in scrum boards and planning tools to help your teams run sprints, stand-ups, and planning meetings.

→**Project insights:**Gain new insights into the health and status of your project with powerful analytics tools and Dashboard.

ii)Azure Repos:

Unlimited private Git repo hosting and support for TFVC that scales from a hobby project to the world's largest Git repositories

→**Works with your Git client:**

Securely connect with and push code into your Git repos from any IDE, editor, or Git client.

→**Web hooks and API integration:**

Add validations and extensions from the marketplace or build your own using web hooks and REST APIs.

→**Semantic code search:**

Quickly find what you're looking for with code-aware search that understands classes and variables.

iii)Azure Pipelines:

Cloud-hosted pipelines for Linux, Windows and macOS, with unlimited minutes for open source.

→**Any language, any platform, any cloud:**

Build, test, and deploy Node.js, Python, Java, PHP, Ruby, C/C++. NET, Android, and iOS apps. Run in parallel on Linux, macOS, and Windows. Deploy to Azure, AWS, GCP or on-premises.

→**Extensible:**

Explore and implement a wide range of community- built build, test, and deployment tasks, along with hundreds of extensions from Slack to SonarCloud. Support for YAML, reporting and more.

→**Best-in-class for open source:**

Ensure fast continuous Integration/continuous delivery (CI/CD) pipelines for every open source project. Get unlimited build minutes for all open up to 10 free parallel

jobs across Linux, macOS and source projects with Windows.

iv)Azure Test Plans:

Get end-to-end traceability. Run tests and log defects from your browser. Track and assess quality throughout your testing lifecycle.

→**Capture rich data:**

Capture rich scenario data as you execute tests to make discovered defects actionable. Explore user stories without test cases or test steps. You can create test cases directly from your exploratory test sessions.

→**Test across web and desktop:**

Test your application where it lives. Complete scripted tests across desktop or web scenarios, Test on- premises application from the cloud and vice-versa.

→**Get end-to-end traceability:**

Leverage the same test tools across your engineers and user acceptance testing stakeholders. Pay for the tools only when you need them.

v)Azure Artifacts:

Create and share npm, and NuGet package feeds from public and private sources - fully integrated into CI/CD pipelines

→**Manage all package types:**

Get universal artifact management for Maven, npm, and NuGet.

→**Add packages to any pipeline:**

Share packages, and use built-in CI/CD, versioning, and testing.

→**Share code efficiently:**

Easily share code across small teams and large enterprises.

2)TECHNOLOGIES WHICH SUPPORTS DEVOPS:

i)Continuous Integration (CI):

→Improve software development quality and speed

→When you use Azure Pipelines or Jenkins to build apps in the cloud and deploy to Azure, each time you commit code, it's automatically built and tested and bugs are detected faster.

ii)Continuous Deployment (CD):

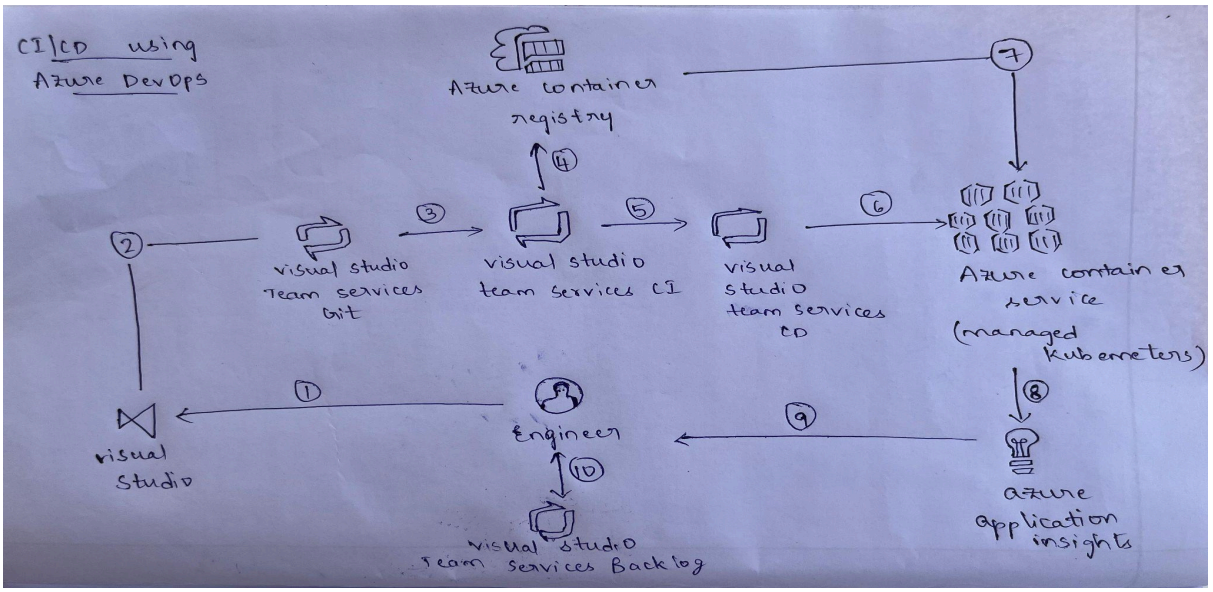
→By combining continuous integration and infrastructure as code (IaC), you'll achieve identical deployments and the confidence to deploy to production at any time.

→With continuous deployment, you can automate the entire process from code commit to production if your CI/CD tests are successful.

iii)Continuous Learning & Monitoring:

→With Azure Application Insights you can identify how your applications are performing and test if the recent deployment made things better or worse.

→Using CI/CD practices, paired with monitoring tools, you'll be able to safely deliver features to your customers as soon as they're ready.



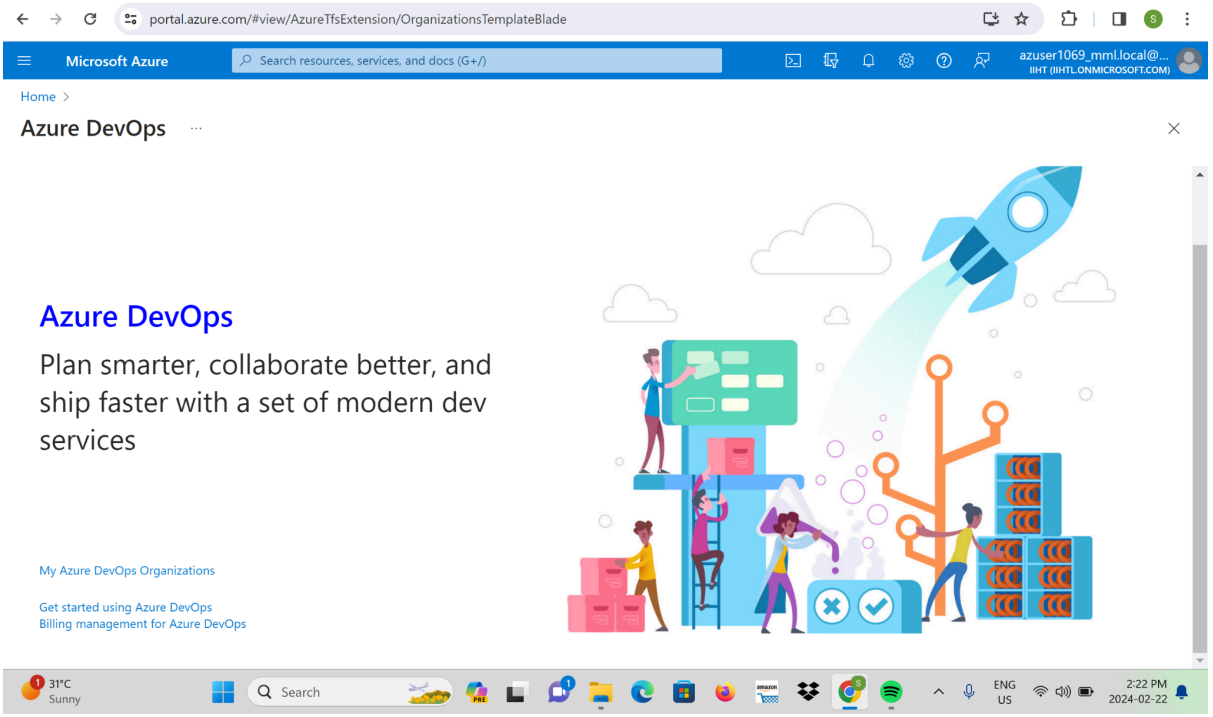
CREATING AZURE DEVOPS ACCOUNT:

→ Firstly we need to login to our azure account.

→Then we need to search azure devops.

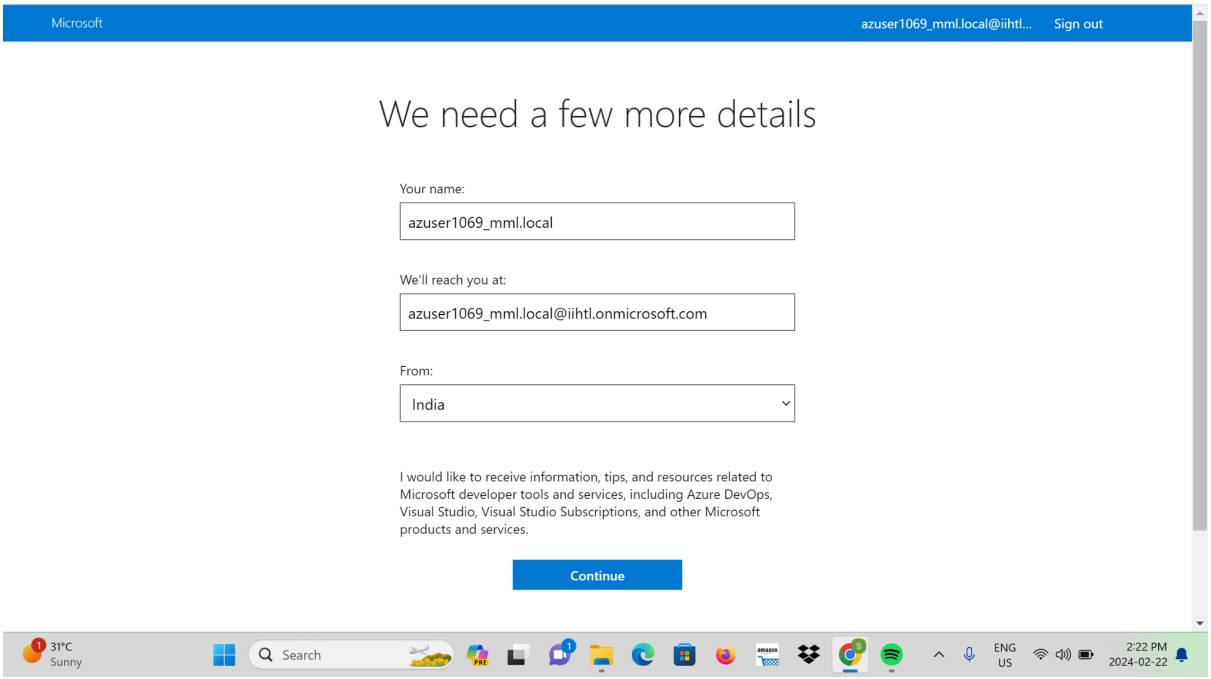
→ After clicking on the azure devops we can see this kind of interface .

→Then we need to click on the “My Azure DevOps Organisations.



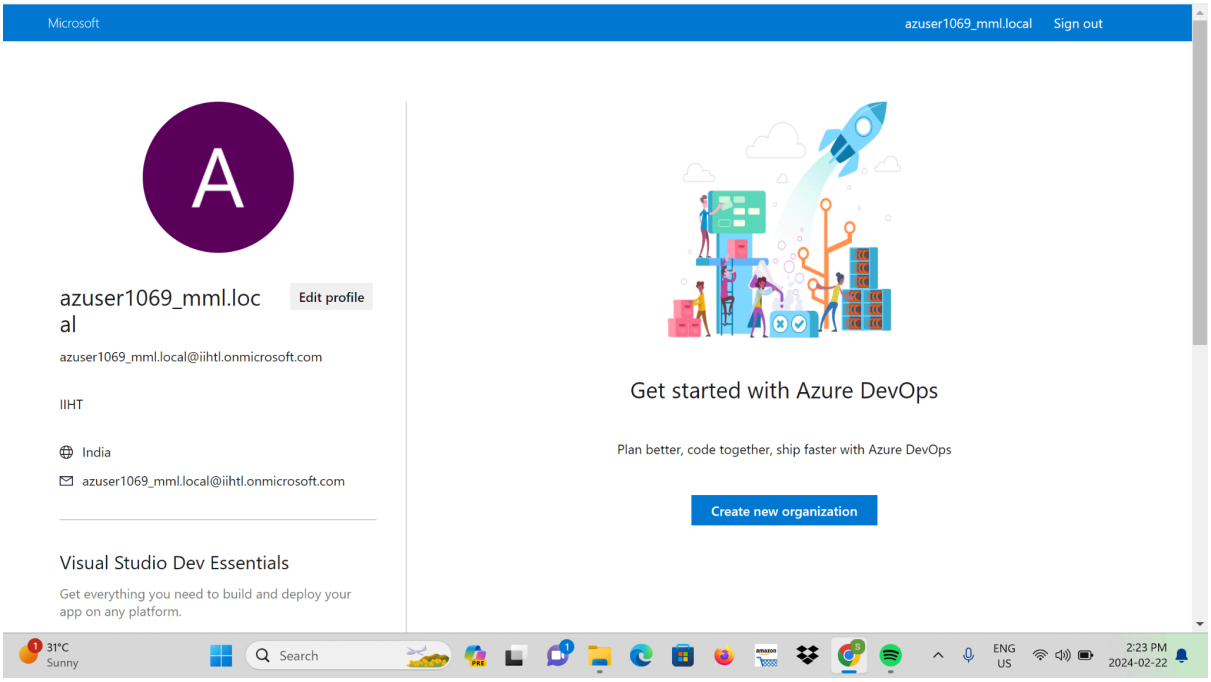
→After that we need to create an azure devops account by giving the name and microsoft account

and region .

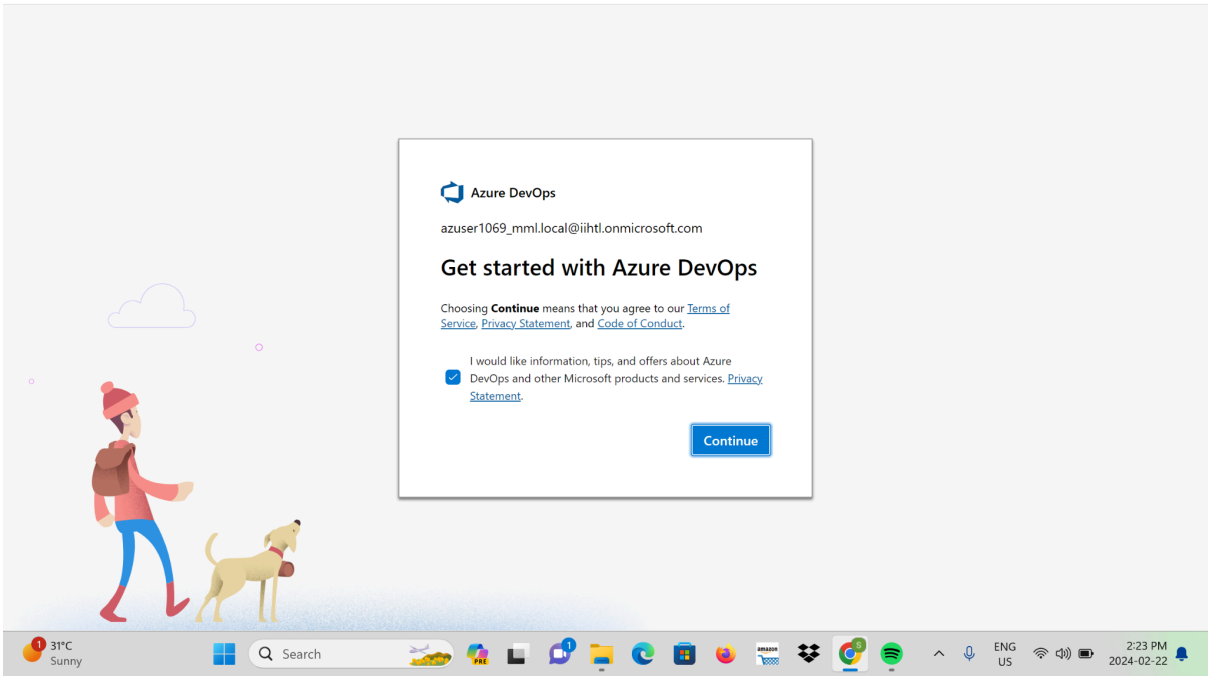


→After creating an account we need to create an organization.

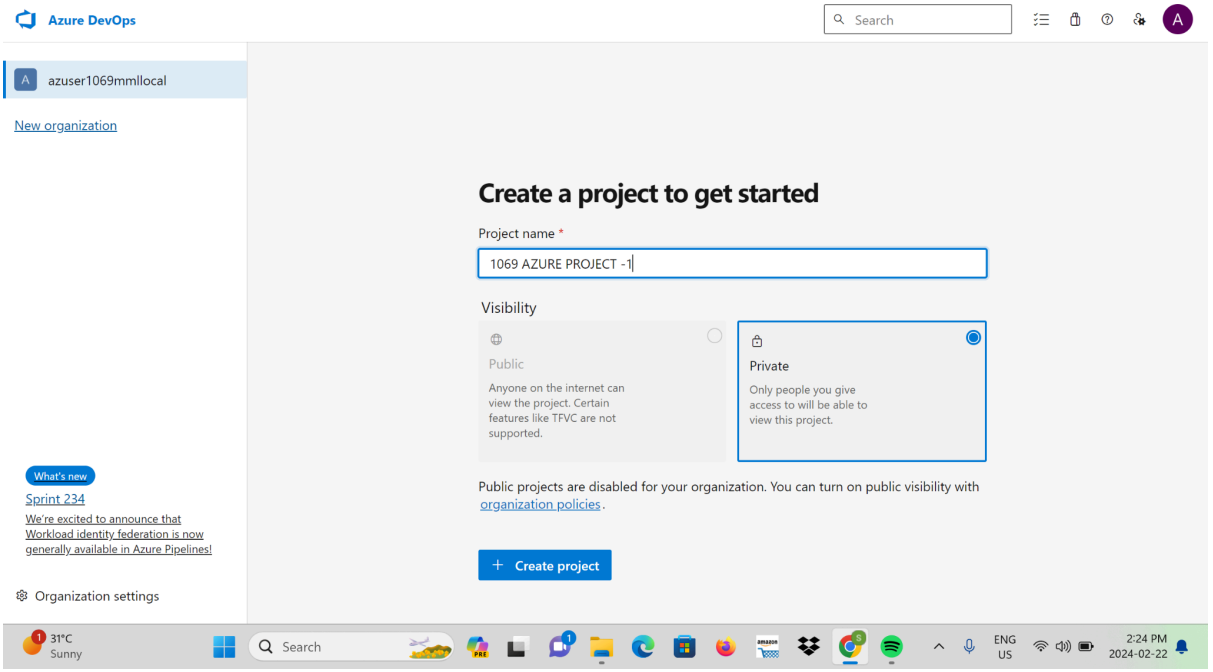
→For creating an organization we need to click on “create new organization” .



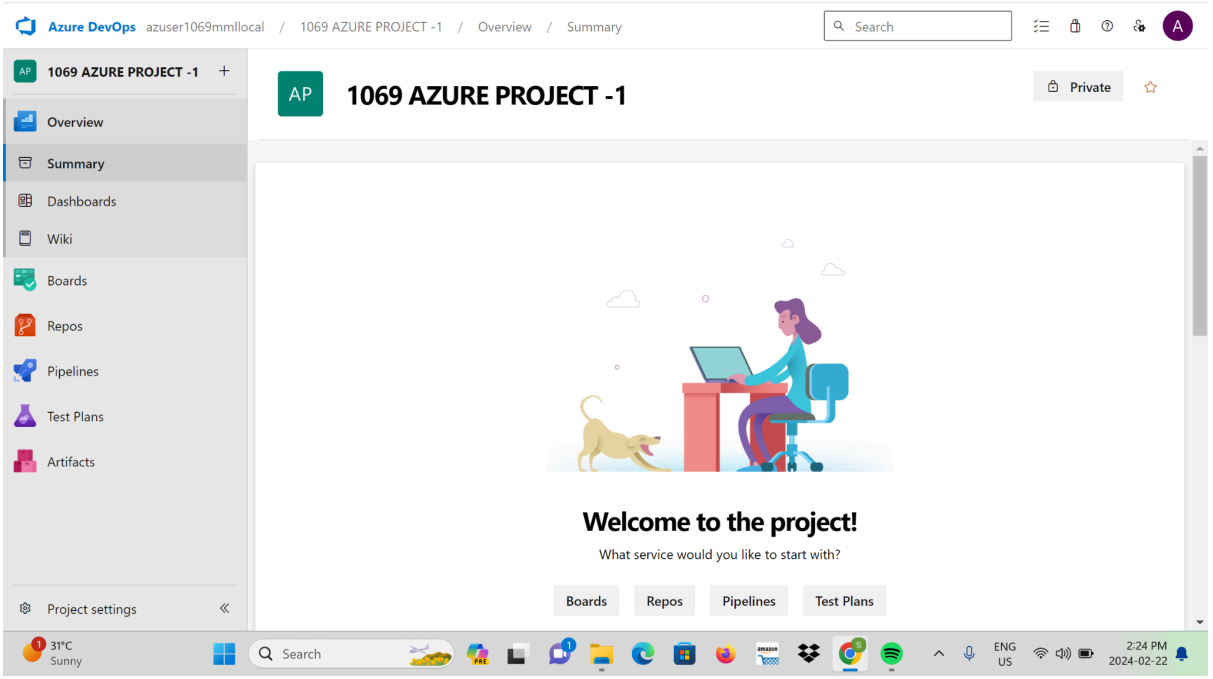
→Then we need to click on continue .



→After that we need to create a project to start the azure devops .

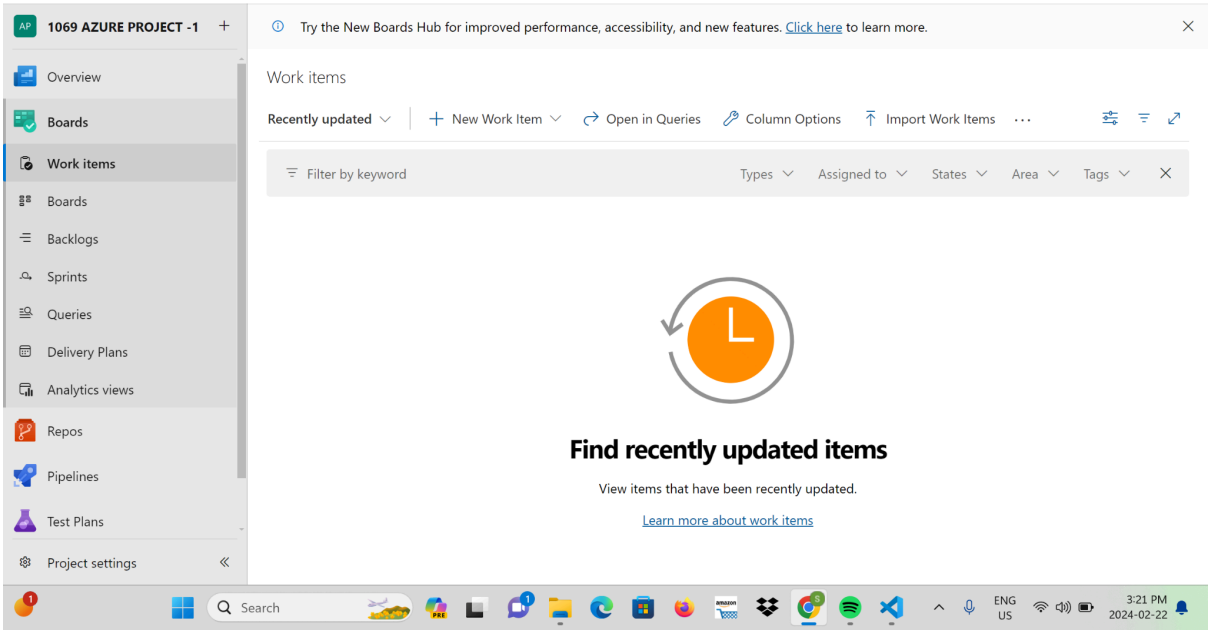


→After creating a project this is the interface of the organisation.



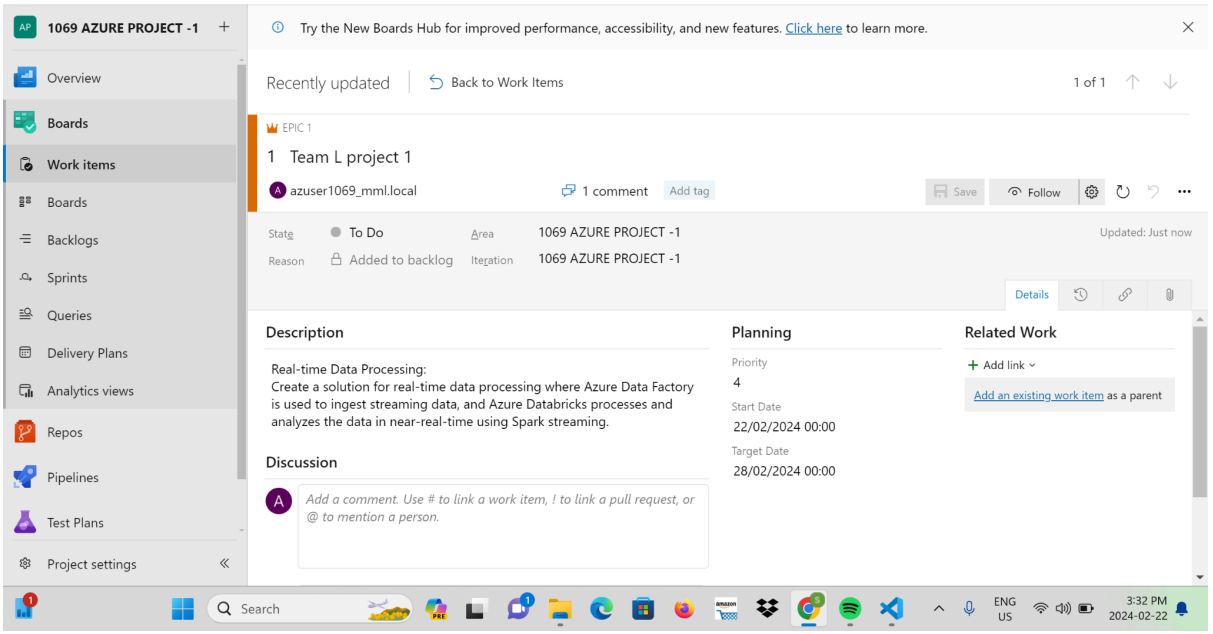
→ There we can see the overview,dashboards,repos,pipelines,test plans,artifacts.

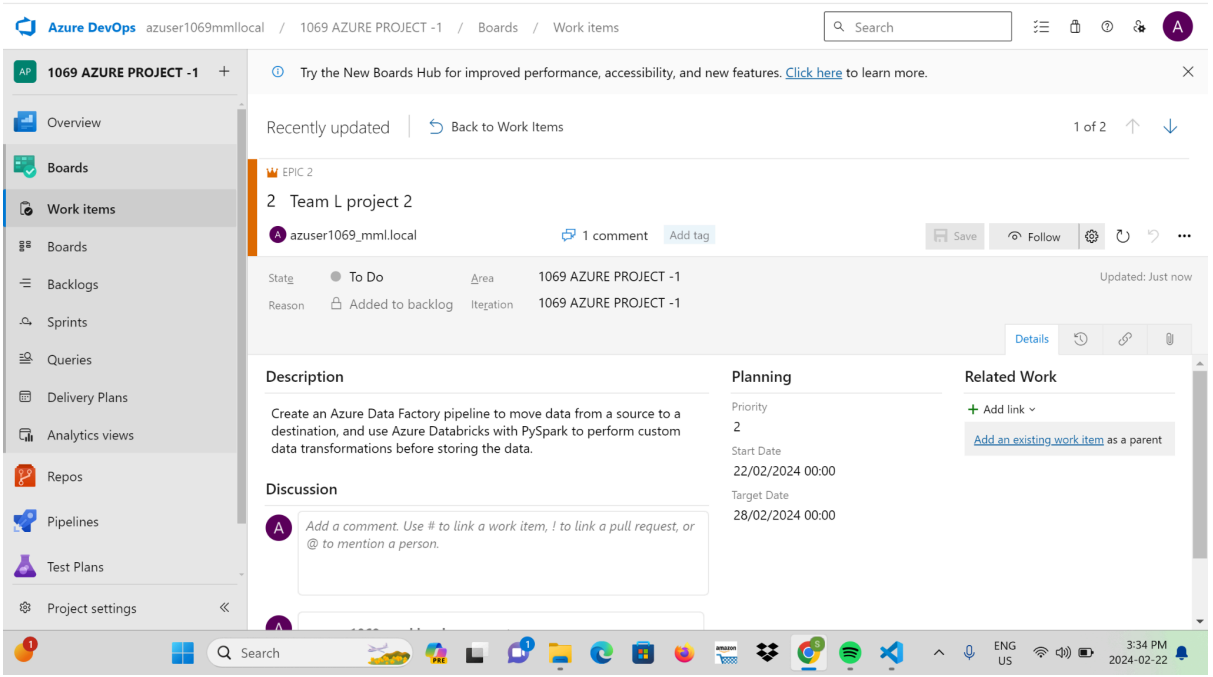
→ We can create an Epic in this dashboard.



→ To create an epic we need to click on Boards and then we need to click on the new work item there by we can see the option called “EPIC”.

→Then we need to give the title of the epic and description,priority and start date ,target date,assign to .





→Then after creating a epic we can view this epic in dashboard as shown below