Connecting SAP CAP (Cloud Application Programming) with GitHub

Author:

Name: Mahesh L

Role: SAP UI5 Consultant

Email: lokkidimahesh6@gmail.com

1. Introduction

- Overview of SAP CAP and its importance in cloud-native development.
- Why version control with GitHub is essential for CAP projects.

2. Prerequisites

- Required tools (Git, BAS, GitHub account).
- Initial setup for your SAP CAP project.

3. Step-by-Step Guide to Connect SAP CAP with GitHub

- Initializing a local Git repository.
- Creating a new repository on GitHub.
- Linking local repository with GitHub.

4. Committing and Pushing Changes

- How to stage, commit, and push your changes to GitHub.
- Git commands explanation.

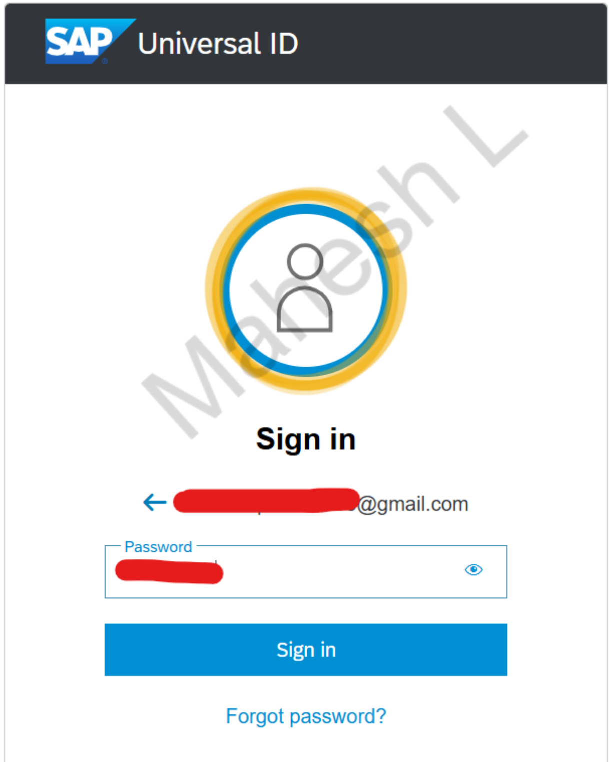
5.Importance of GitHub

1. **Version Control & Collaboration**
   - Tracks changes in code with Git.
   - Multiple developers can work on the same project simultaneously.
2. **Backup & Deployment**
   - Stores code securely in the cloud.
   - Supports CI/CD for automated testing and deployment.
3. **Security & Project Management**
   - Provides private/public repositories with access control.
   - Helps manage projects with issues, pull requests, and documentation.

1.1 Logging into SAP BTP Account

Screenshot 1: Logging into Your SAP BTP Account

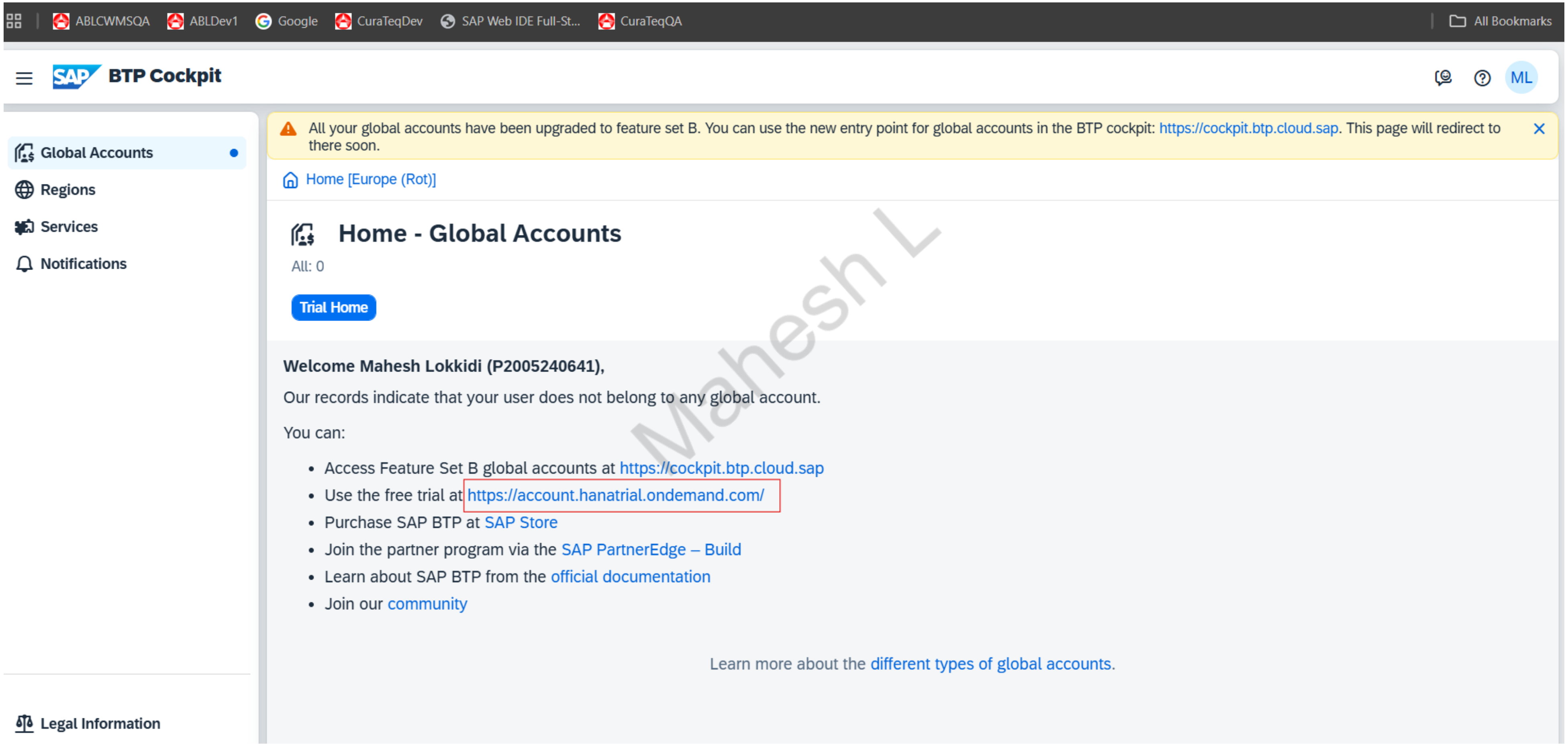
1. Open your browser and navigate to the SAP Business Technology Platform (BTP) cockpit login page.
2. Enter your credentials (email and password) associated with your SAP BTP account.
3. Click **Log in** to access your BTP account dashboard.



Sign in

[redacted]@gmail.com

Password: [redacted]



All your global accounts have been upgraded to feature set B. You can use the new entry point for global accounts in the BTP cockpit: https://cockpit.btp.cloud.sap. This page will redirect to there soon.

Home [Europe (Rot)]

Home - Global Accounts

All: 0

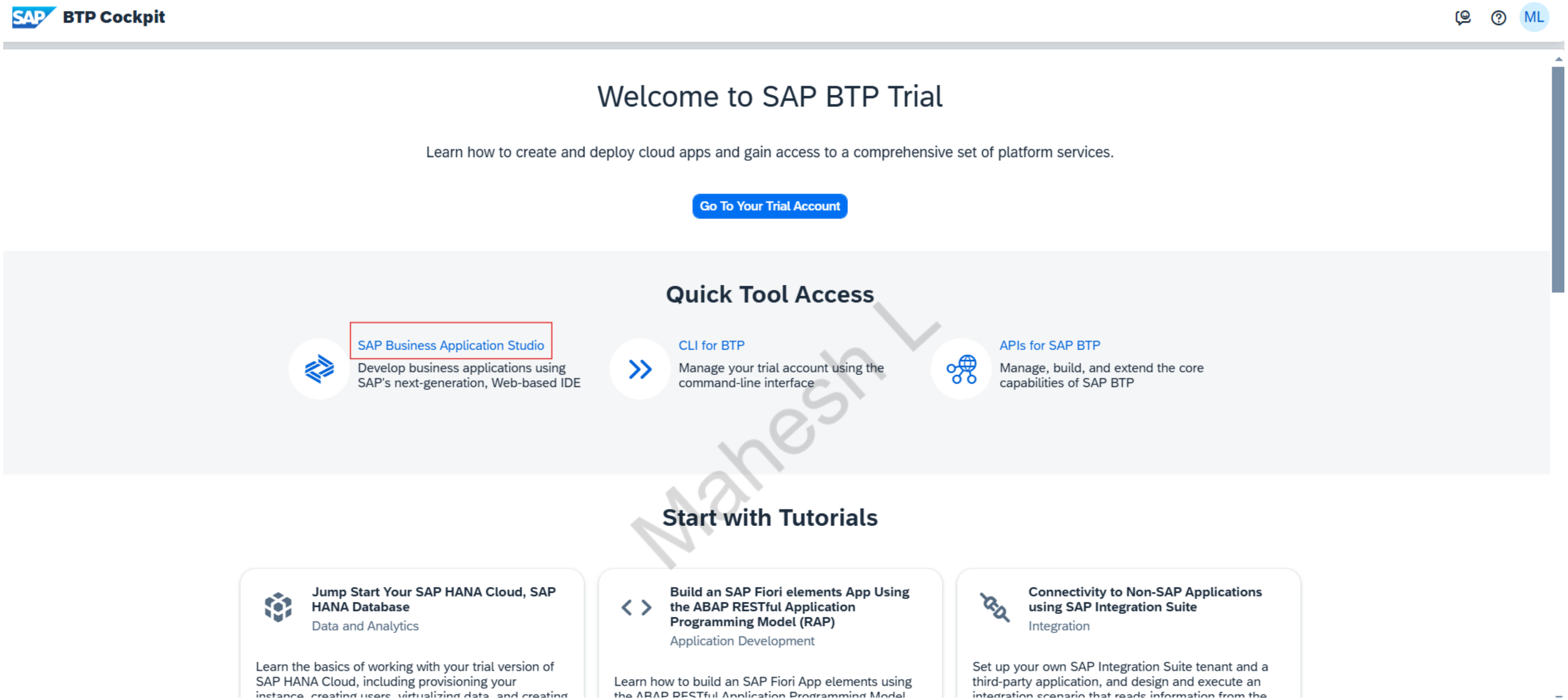
Welcome Mahesh Lokkidi (P2005240641),

Our records indicate that your user does not belong to any global account.

You can:

- Access Feature Set B global accounts at https://cockpit.btp.cloud.sap
- Use the free trial at https://account.hanatrial.ondemand.com/
- Purchase SAP BTP at SAP Store
- Join the partner program via the SAP PartnerEdge – Build
- Learn about SAP BTP from the official documentation
- Join our community

Learn more about the different types of global accounts.



Welcome to SAP BTP Trial

Learn how to create and deploy cloud apps and gain access to a comprehensive set of platform services.

Quick Tool Access

SAP Business Application Studio
Develop business applications using SAP's next-generation, Web-based IDE

CLI for BTP
Manage your trial account using the command-line interface

APIs for SAP BTP
Manage, build, and extend the core capabilities of SAP BTP

Start with Tutorials

Jump Start Your SAP HANA Cloud, SAP HANA Database
Data and Analytics

Learn the basics of working with your trial version of SAP HANA Cloud, including provisioning your instance, creating users, virtualizing data, and creating

Build an SAP Fiori elements App Using the ABAP RESTful Application Programming Model (RAP)
Application Development

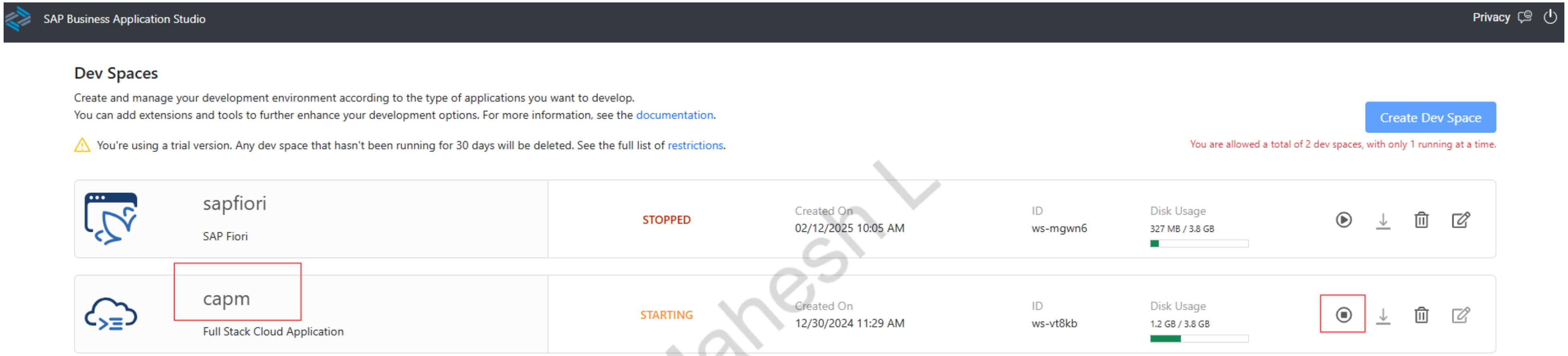
Learn how to build an SAP Fiori App elements using the ABAP RESTful Application Programming Model

Connectivity to Non-SAP Applications using SAP Integration Suite
Integration

Set up your own SAP Integration Suite tenant and a third-party application, and design and execute an integration scenario that reads information from the

1.2 Accessing the CAPM Space in SAP BTP

1. After logging into your SAP BTP account, go to the **CAPM Space**.
2. On the left-hand side of the dashboard, click on the **Start** button.
3. From the available options, click on **CAPM** to enter the CAPM space.



Dev Spaces

Create and manage your development environment according to the type of applications you want to develop.
You can add extensions and tools to further enhance your development options. For more information, see the documentation.

⚠ You're using a trial version. Any dev space that hasn't been running for 30 days will be deleted. See the full list of restrictions.

You are allowed a total of 2 dev spaces, with only 1 running at a time.

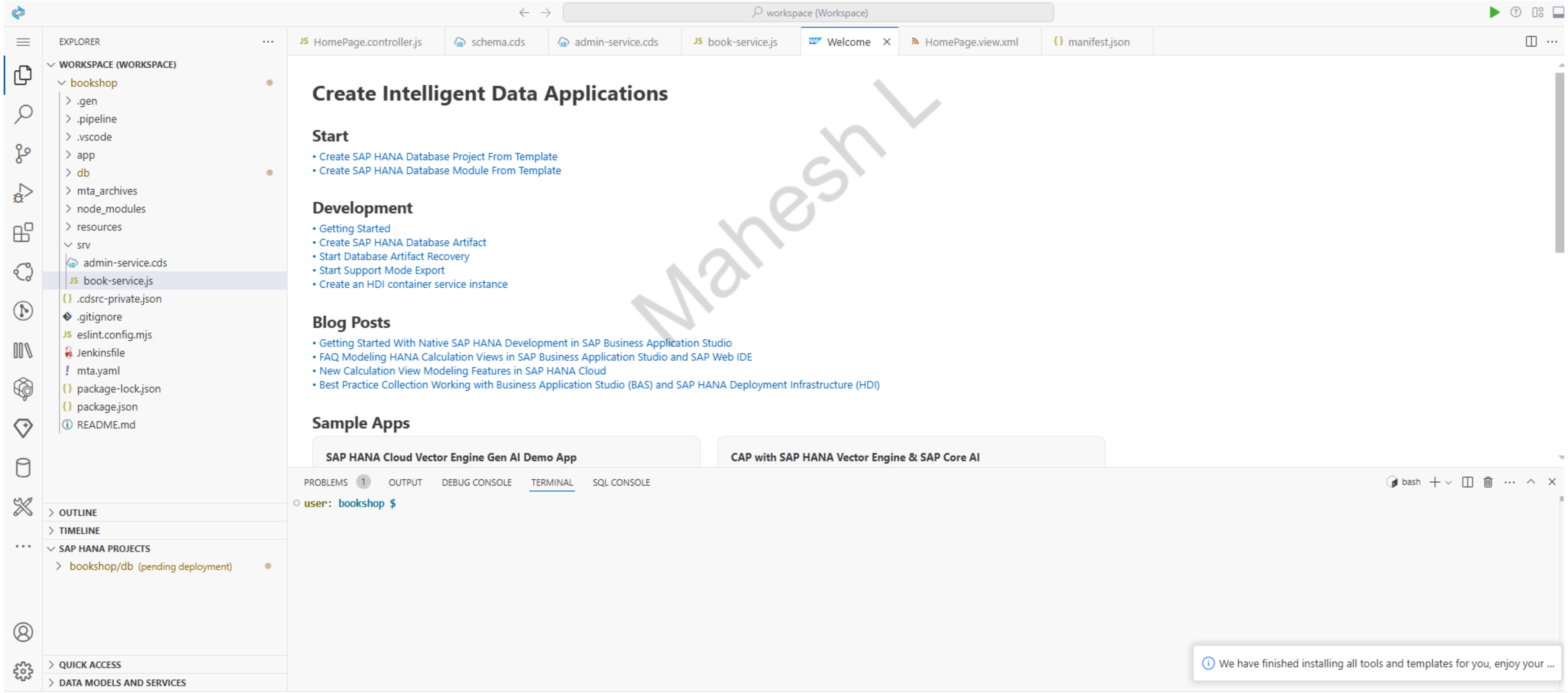
	Status	Created On	ID	Disk Usage
sapfiori SAP Fiori	STOPPED	02/12/2025 10:05 AM	ws-mgwn6	327 MB / 3.8 GB
capm Full Stack Cloud Application	STARTING	12/30/2024 11:29 AM	ws-vt8kb	1.2 GB / 3.8 GB

1.3 Creating a New bookshop CAP Project

If you don't have an existing CAP project, you can follow the steps in the document provided below to create a new one.

1. Go to the provided link for the detailed guide on how to create a new SAP CAP project.
2. Follow the steps outlined in the guide to set up your project.

https://www.linkedin.com/posts/mahesh-lokkidi-8664b9135_sap-capm-steps-pdf-guide-activity-7278389861781315585-nKxN?utm_source=share&utm_medium=member_desktop&rcm=ACoAACD__I4Bx2jyujXgeNLseFsss4Zzd5zrhW0



Create Intelligent Data Applications

Start

- Create SAP HANA Database Project From Template
- Create SAP HANA Database Module From Template

Development

- Getting Started
- Create SAP HANA Database Artifact
- Start Database Artifact Recovery
- Start Support Mode Export
- Create an HDI container service instance

Blog Posts

- Getting Started With Native SAP HANA Development in SAP Business Application Studio
- FAQ Modeling HANA Calculation Views in SAP Business Application Studio and SAP Web IDE
- New Calculation View Modeling Features in SAP HANA Cloud
- Best Practice Collection Working with Business Application Studio (BAS) and SAP HANA Deployment Infrastructure (HDI)

Sample Apps

SAP HANA Cloud Vector Engine Gen AI Demo App

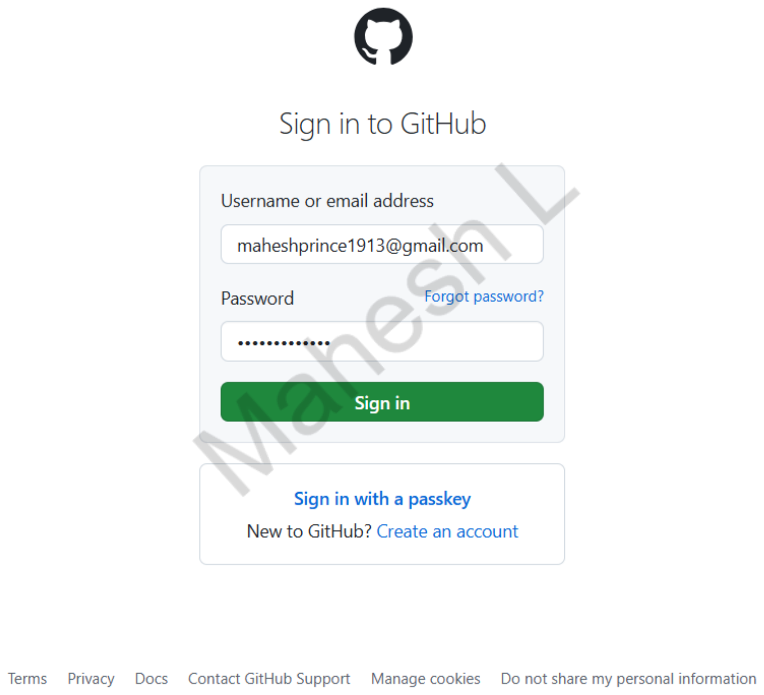
CAP with SAP HANA Vector Engine & SAP Core AI

```
user: bookshop $
```

We have finished installing all tools and templates for you, enjoy your ...

2.1 Go to GitHub and Log In

- Open GitHub in your browser.
- Enter your credentials (username/email and password) to log in.
- If you have enabled two-factor authentication (2FA), complete the verification process.



Sign in to GitHub

Username or email address

maheshprince1913@gmail.com

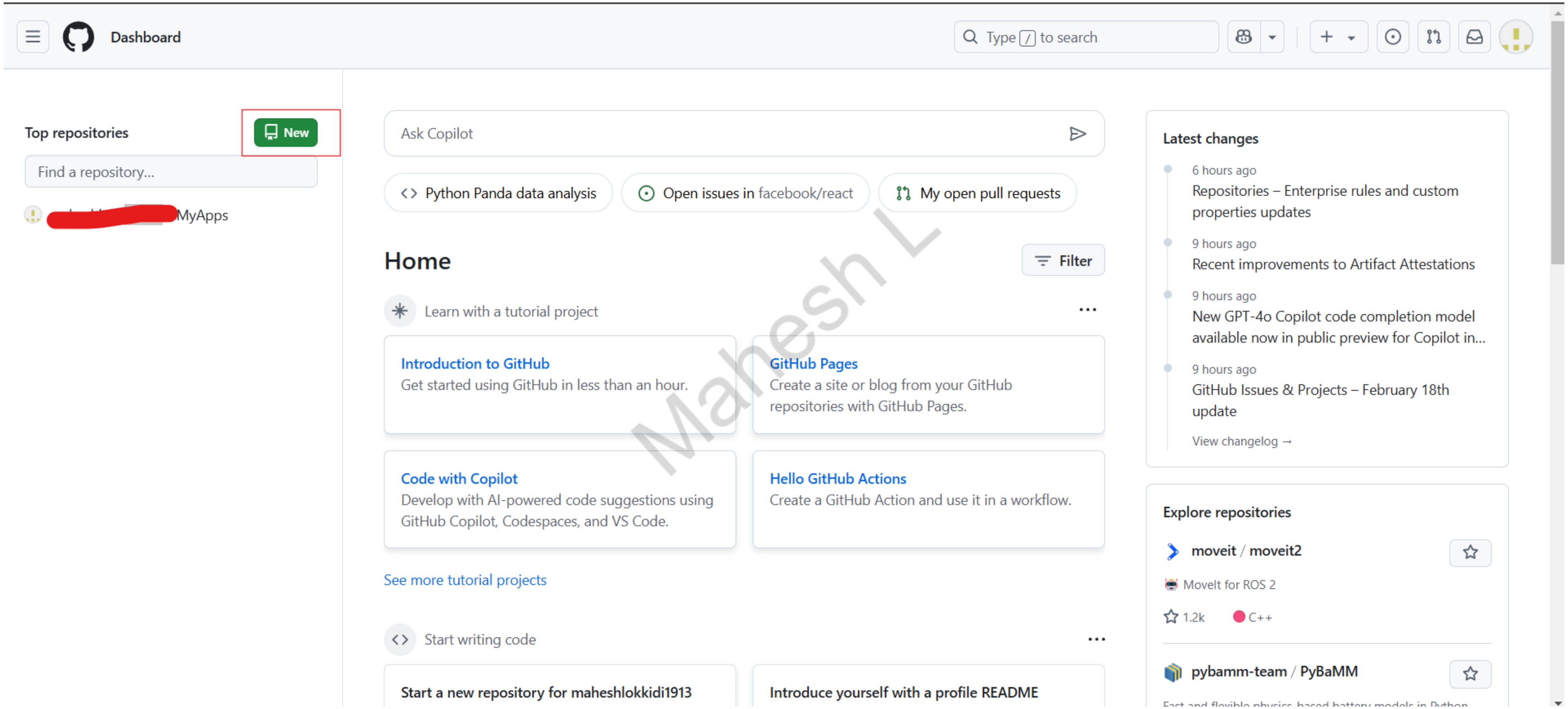
Password

••••••••••••••

New to GitHub? Create an account

2.2 Create a New Repository

- After logging into GitHub, click on the **"+"** icon in the top-right corner.
- Select **"New repository"** from the dropdown menu.



Top repositories

New

Find a repository...

[redacted]MyApps

Ask Copilot

<> Python Panda data analysis

Open issues in facebook/react

My open pull requests

Home

Filter

Learn with a tutorial project

Introduction to GitHub
Get started using GitHub in less than an hour.

GitHub Pages
Create a site or blog from your GitHub repositories with GitHub Pages.

Code with Copilot
Develop with AI-powered code suggestions using GitHub Copilot, Codespaces, and VS Code.

Hello GitHub Actions
Create a GitHub Action and use it in a workflow.

See more tutorial projects

<> Start writing code

Start a new repository for maheshlokkidi1913

Introduce yourself with a profile README

Latest changes

6 hours ago
Repositories – Enterprise rules and custom properties updates

9 hours ago
Recent improvements to Artifact Attestations

9 hours ago
New GPT-4o Copilot code completion model available now in public preview for Copilot in...

9 hours ago
GitHub Issues & Projects – February 18th update

View changelog →

Explore repositories

moveit / moveit2
MoveIt for ROS 2
☆ 1.2k ● C++

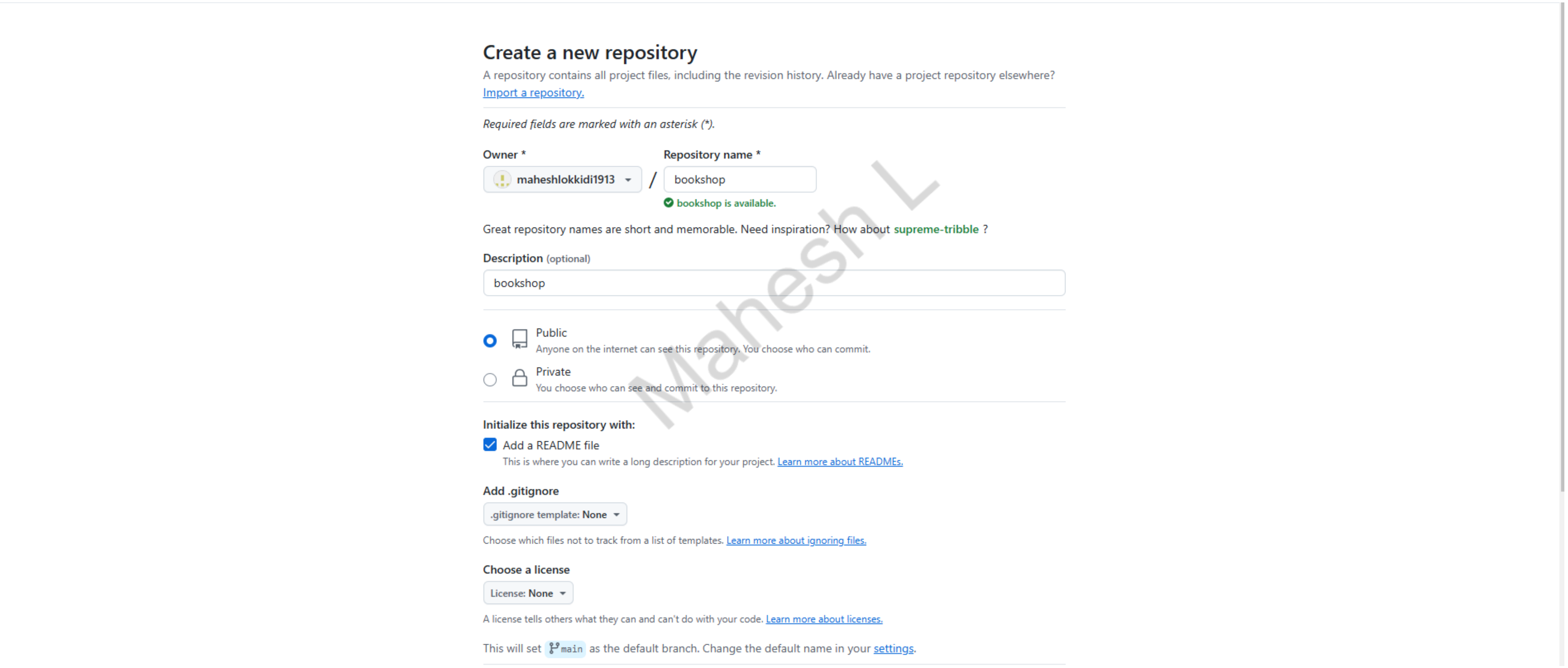
pybamm-team / PyBaMM
Fast and flexible physics-based battery models in Python

- Enter a **repository name**, add a description (optional), and choose the visibility (Public/Private).

a. **Public**: Anyone can see the repository.

b. **Private**: Only you and invited collaborators can access it.

- Click **"Create repository"** to proceed



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner * maheshlokkidi1913 / Repository name * bookshop

bookshop is available.

Great repository names are short and memorable. Need inspiration? How about supreme-tribble ?

Description (optional)

bookshop

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

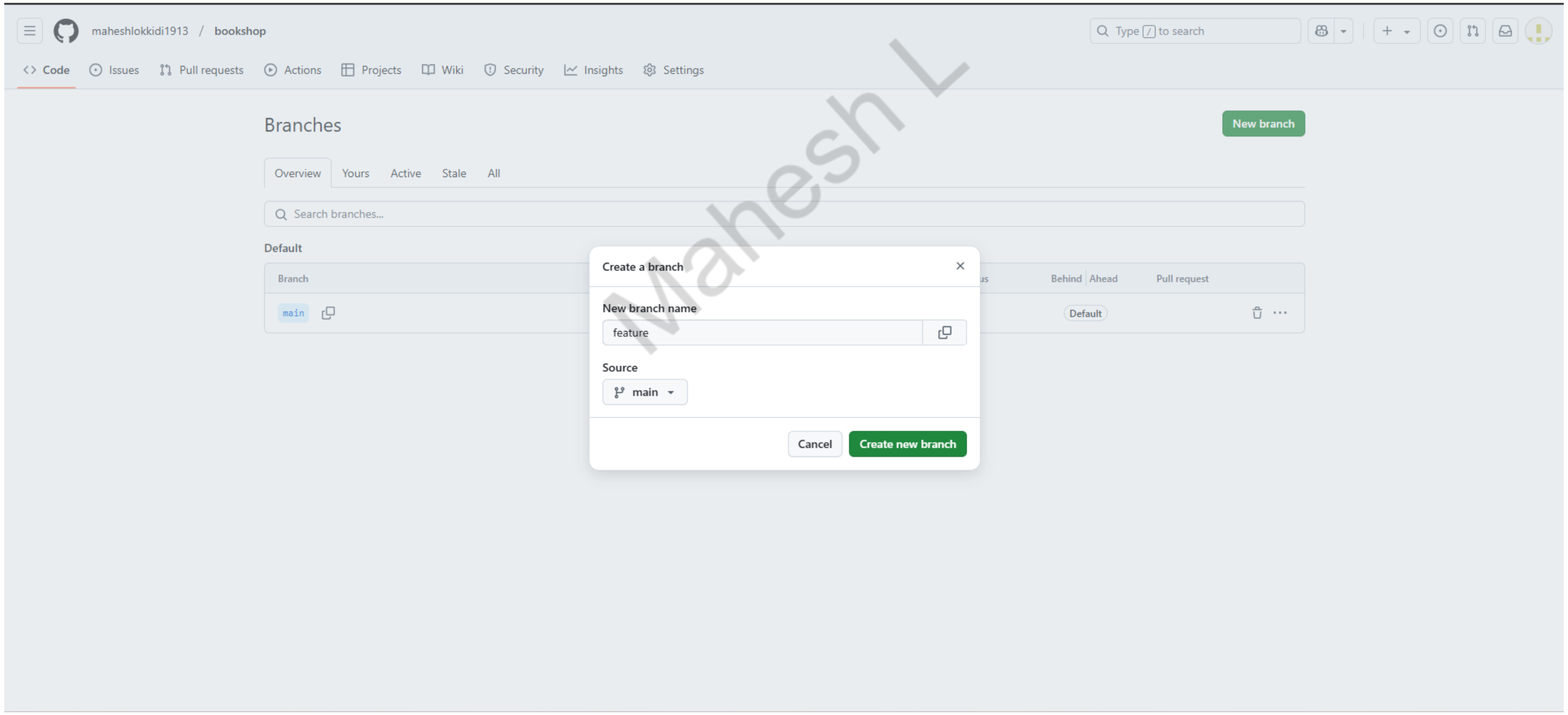
This will set main as the default branch. Change the default name in your settings.

2.3 Create a Feature Branch

- After creating the repository, navigate to the **"Code"** section.
- Click on the **"main"** branch dropdown.
- Type **"feature"** as the new branch name and click **"Create branch: feature"**.

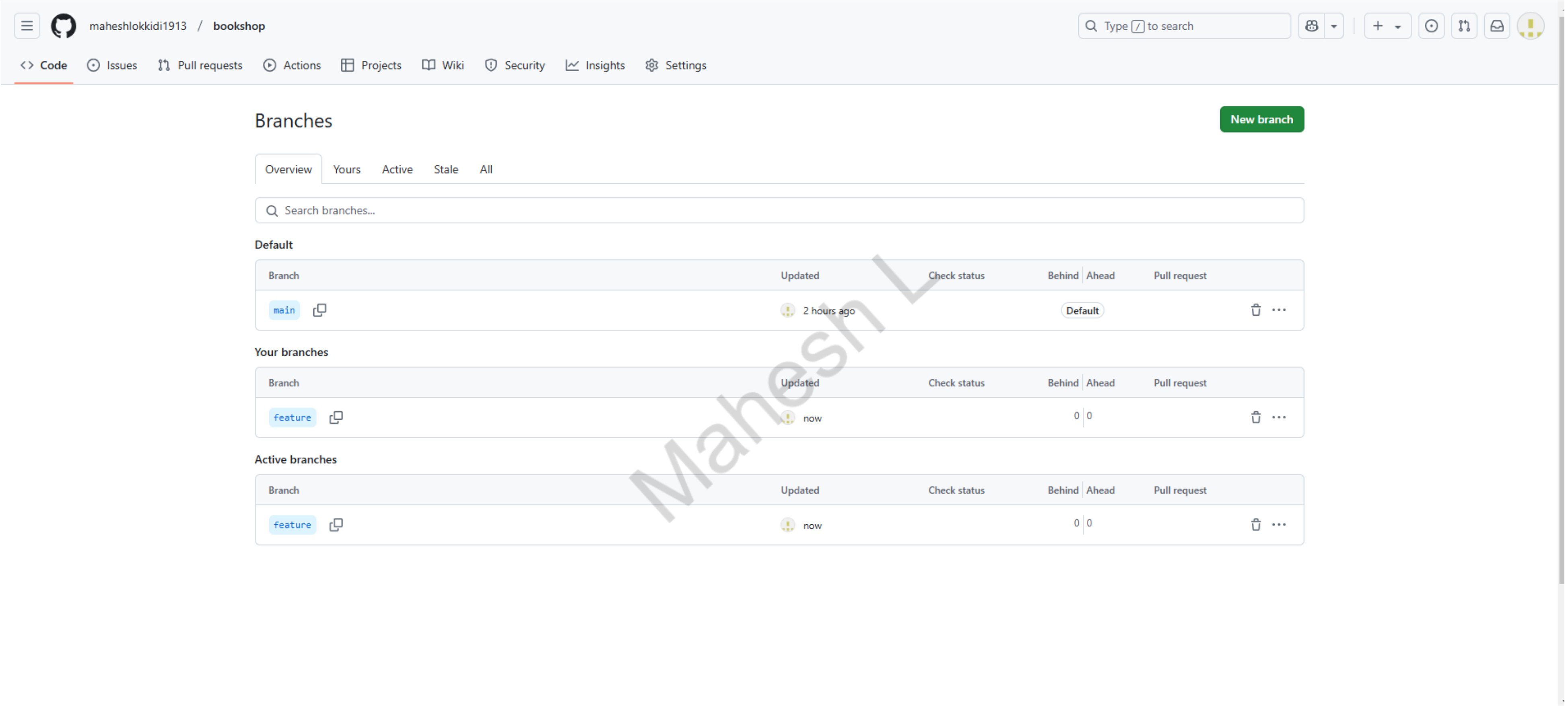
Why Do We Use Main and Feature Branches?

- **Main Branch**: This is the stable version of the project, usually used for production-ready code.
- **Feature Branch**: Developers create feature branches to work on new functionalities or fixes without affecting the main branch. Once the changes are tested and verified, they can be merged into the main branch.



New branch name: feature

Source: main



Branches

Default

Branch	Updated	Check status	Behind	Ahead	Pull request
main	2 hours ago		Default		

Your branches

Branch	Updated	Check status	Behind	Ahead	Pull request
feature	now		0	0	

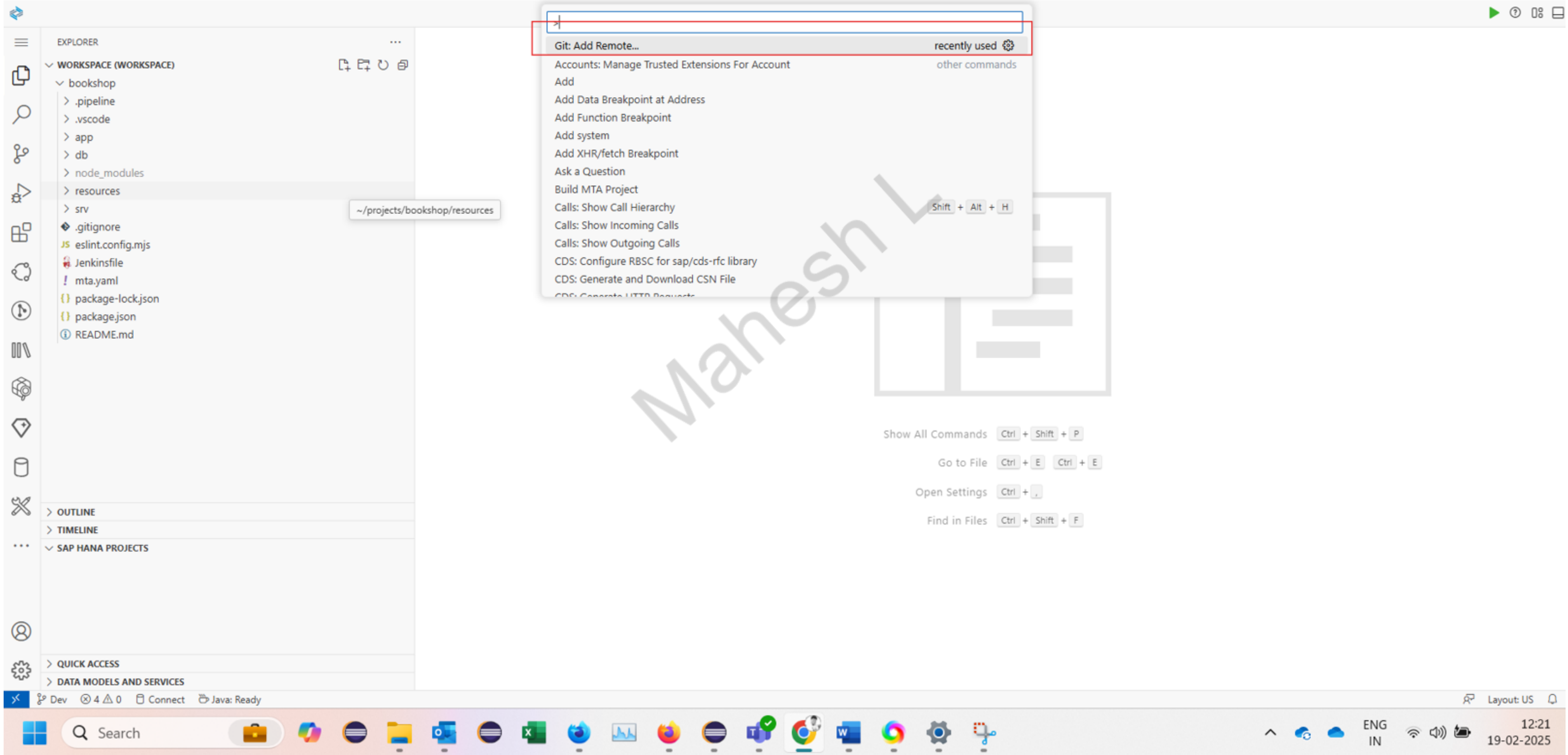
Active branches

Branch	Updated	Check status	Behind	Ahead	Pull request
feature	now		0	0	

3.1 Add Remote Repository in BAS

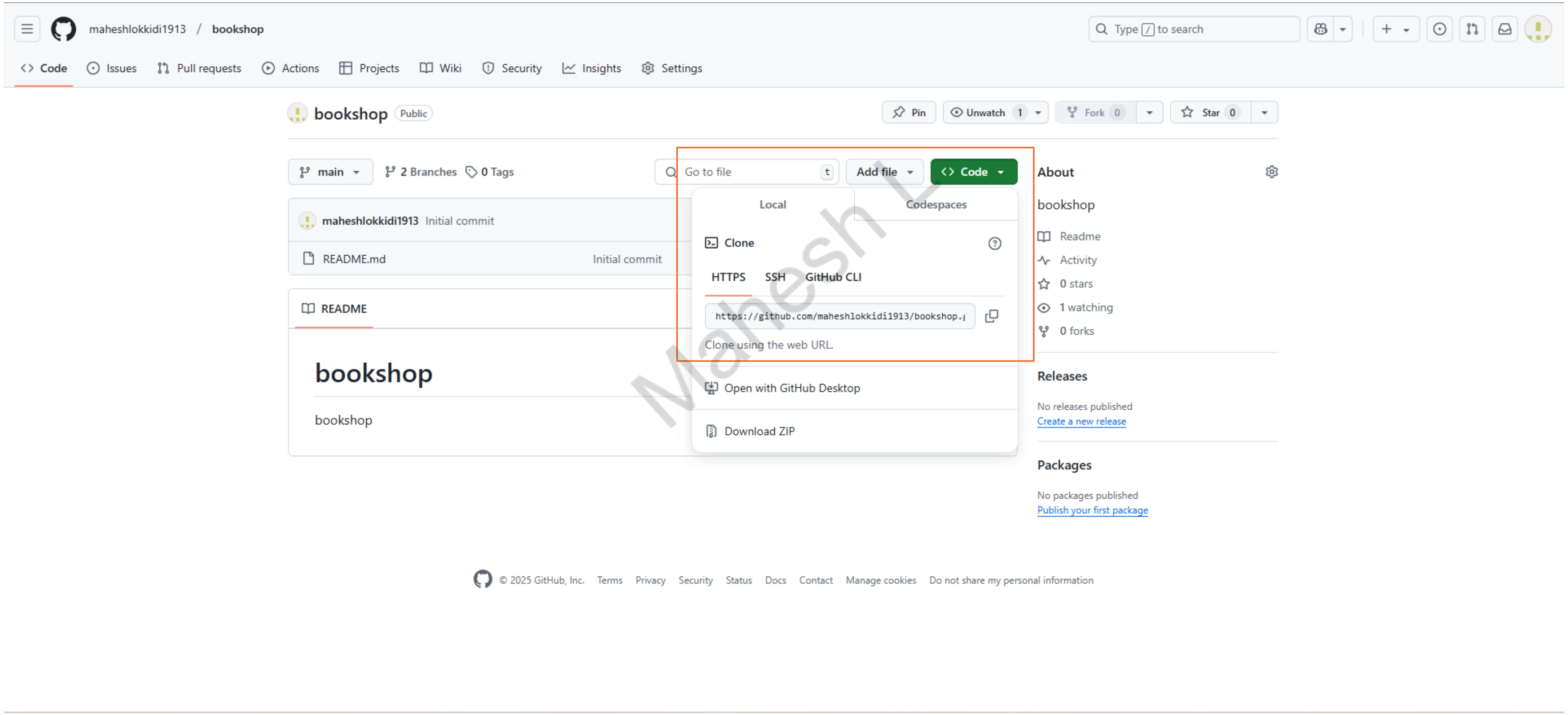
- Open **SAP Business Application Studio (BAS)**.
- Go to **View** → **Command Palette**.
- In the command palette, type:

```
Git: Add Remote
```



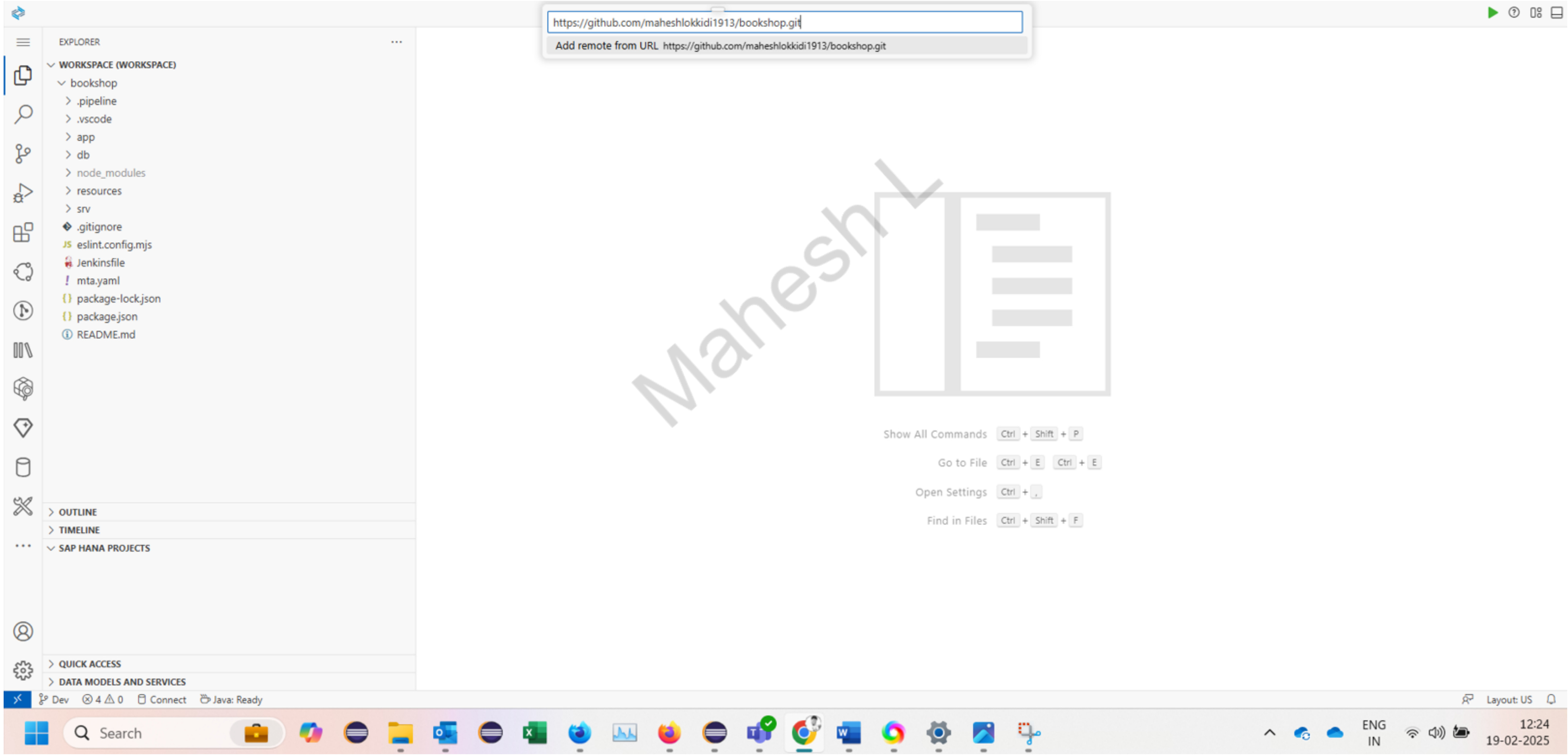
3.2 Copy GitHub Repository URL

- Open your **GitHub repository**.
- Click on the **"Code"** button.
- Under the **HTTPS** tab, copy the repository URL.
  - Example: https://github.com/username/repository.git



3.3 Paste Repository URL in Command Palette

- Go back to **SAP Business Application Studio (BAS)**.
- In the **Command Palette**, after selecting **"Git: Add Remote"**, paste the **GitHub repository URL** (copied in step 3.2).
- Press **Enter** to confirm

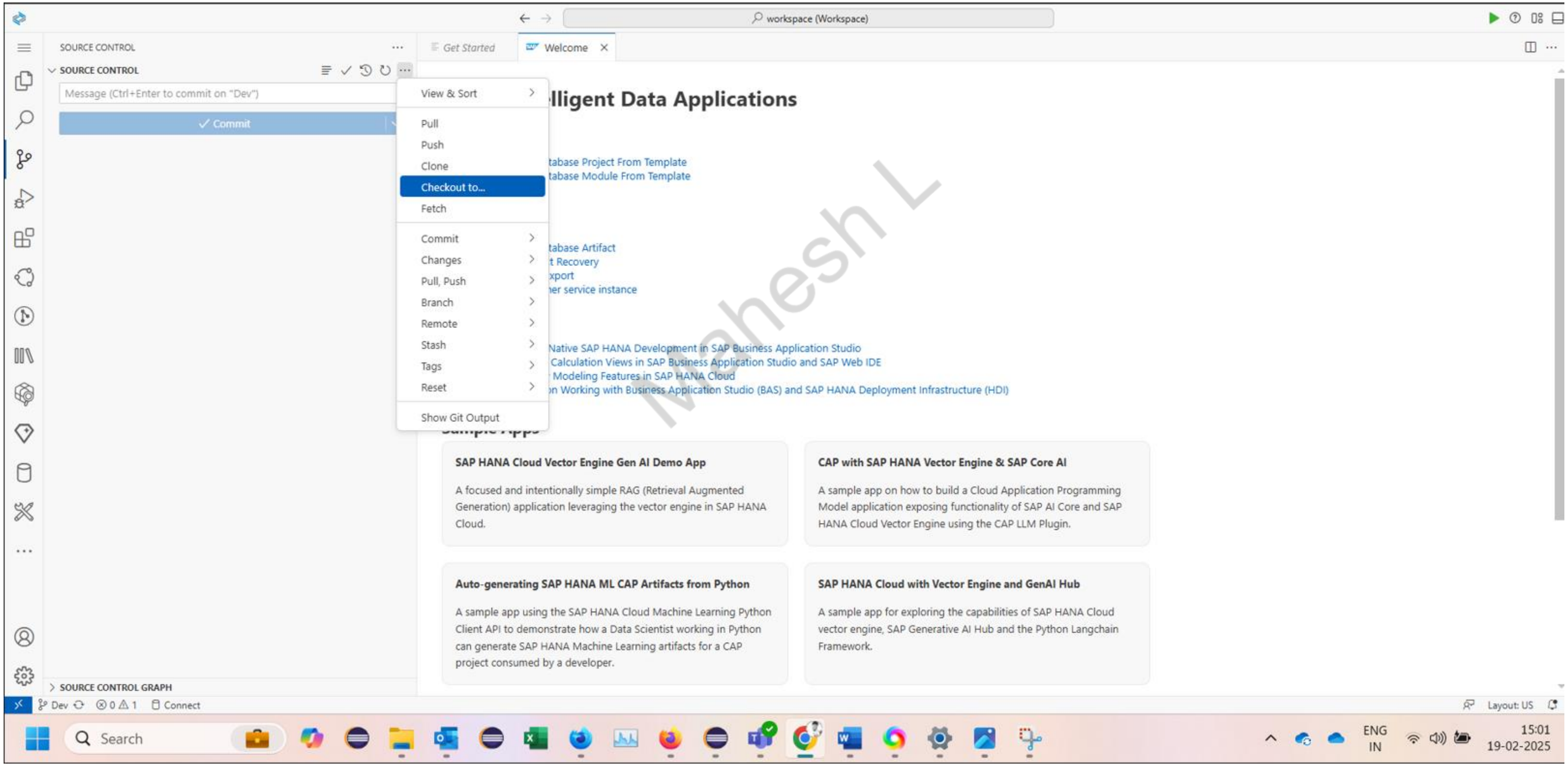


https://github.com/maheshlokkidi1913/bookshop.git

Add remote from URL https://github.com/maheshlokkidi1913/bookshop.git

3.4 Switch to the Feature Branch

- In **SAP Business Application Studio (BAS)**, open the **Source Control** panel.
- Click on the **branch dropdown** (usually showing "main").
- Select **"feature"** to switch to the feature branch.
- If the feature branch is not listed, click **"Checkout to..."** and select **"feature"**.


workspace (Workspace)
SOURCE CONTROL
SOURCE CONTROL
Message (Ctrl+Enter to commit on "Dev")
✓ Commit
Get Started
Welcome
View & Sort
Pull
Push
Clone
Checkout to...
Fetch
Commit
Changes
Pull, Push
Branch
Remote
Stash
Tags
Reset
Show Git Output
lligent Data Applications
tabase Project From Template
tabase Module From Template
tabase Artifact
t Recovery
xport
ner service instance
Native SAP HANA Development in SAP Business Application Studio
Calculation Views in SAP Business Application Studio and SAP Web IDE
Modeling Features in SAP HANA Cloud
on Working with Business Application Studio (BAS) and SAP HANA Deployment Infrastructure (HDI)
Sample Apps
SOURCE CONTROL GRAPH
Dev
⊗ 0 ⚠ 1
Connect
Layout: US
Search
ENG
IN
15:01
19-02-2025

SAP HANA Cloud Vector Engine Gen AI Demo App

A focused and intentionally simple RAG (Retrieval Augmented Generation) application leveraging the vector engine in SAP HANA Cloud.

CAP with SAP HANA Vector Engine & SAP Core AI

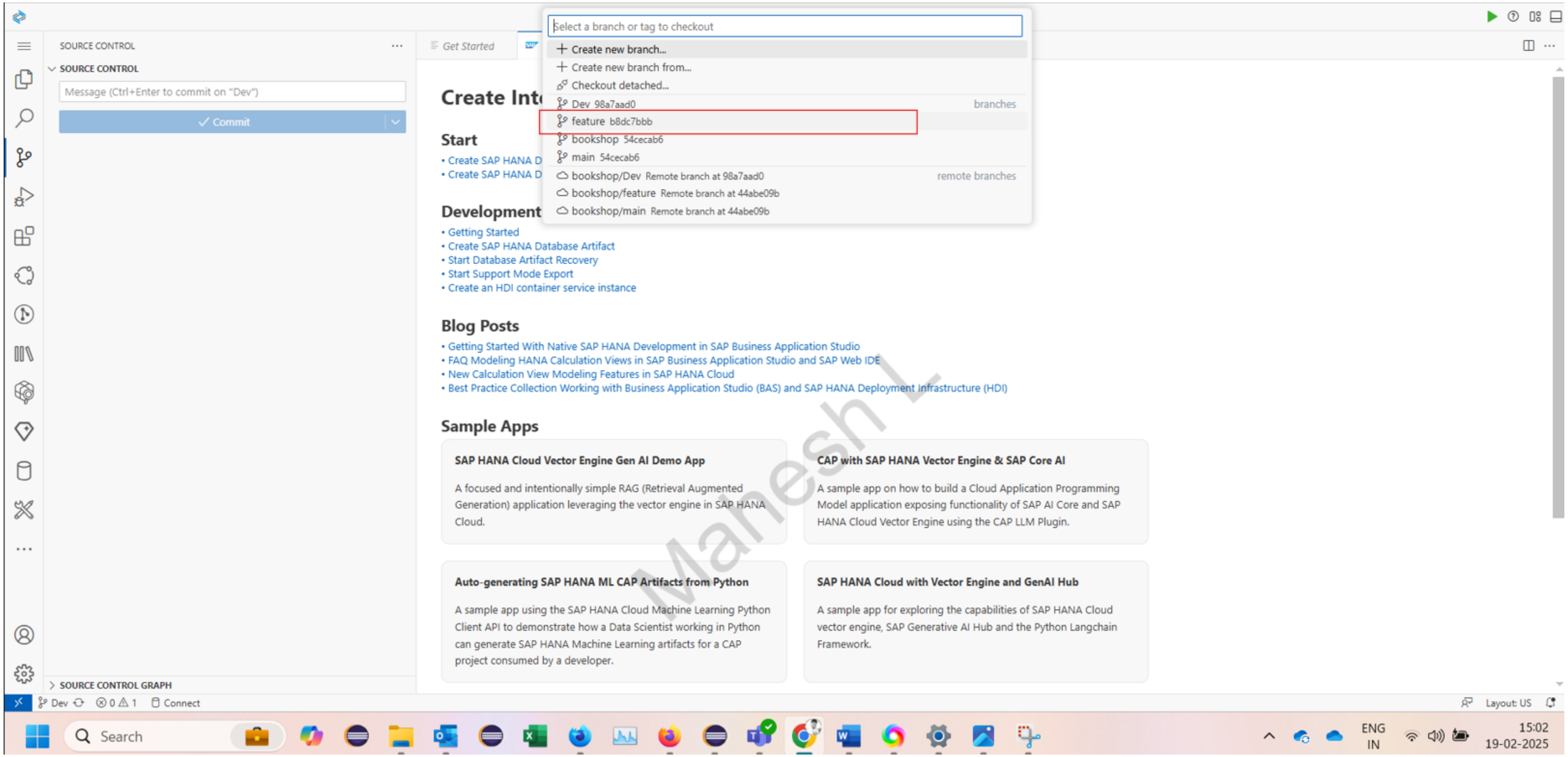
A sample app on how to build a Cloud Application Programming Model application exposing functionality of SAP AI Core and SAP HANA Cloud Vector Engine using the CAP LLM Plugin.

Auto-generating SAP HANA ML CAP Artifacts from Python

A sample app using the SAP HANA Cloud Machine Learning Python Client API to demonstrate how a Data Scientist working in Python can generate SAP HANA Machine Learning artifacts for a CAP project consumed by a developer.

SAP HANA Cloud with Vector Engine and GenAI Hub

A sample app for exploring the capabilities of SAP HANA Cloud vector engine, SAP Generative AI Hub and the Python Langchain Framework.



SOURCE CONTROL
SOURCE CONTROL
Message (Ctrl+Enter to commit on "Dev")
✓ Commit
Get Started
Select a branch or tag to checkout
+ Create new branch...
+ Create new branch from...
Checkout detached...
Dev 98a7aad0 branches
feature b8dc7bbb
bookshop 54cecab6
main 54cecab6
bookshop/Dev Remote branch at 98a7aad0 remote branches
bookshop/feature Remote branch at 44abe09b
bookshop/main Remote branch at 44abe09b
SOURCE CONTROL GRAPH
Dev ⊗ 0 ⚠ 1 Connect
Layout: US
Search
ENG IN
15:02
19-02-2025

Create Int

Start

- Create SAP HANA D
- Create SAP HANA D

Development

- Getting Started
- Create SAP HANA Database Artifact
- Start Database Artifact Recovery
- Start Support Mode Export
- Create an HDI container service instance

Blog Posts

- Getting Started With Native SAP HANA Development in SAP Business Application Studio
- FAQ Modeling HANA Calculation Views in SAP Business Application Studio and SAP Web IDE
- New Calculation View Modeling Features in SAP HANA Cloud
- Best Practice Collection Working with Business Application Studio (BAS) and SAP HANA Deployment Infrastructure (HDI)

Sample Apps

SAP HANA Cloud Vector Engine Gen AI Demo App

A focused and intentionally simple RAG (Retrieval Augmented Generation) application leveraging the vector engine in SAP HANA Cloud.

CAP with SAP HANA Vector Engine & SAP Core AI

A sample app on how to build a Cloud Application Programming Model application exposing functionality of SAP AI Core and SAP HANA Cloud Vector Engine using the CAP LLM Plugin.

Auto-generating SAP HANA ML CAP Artifacts from Python

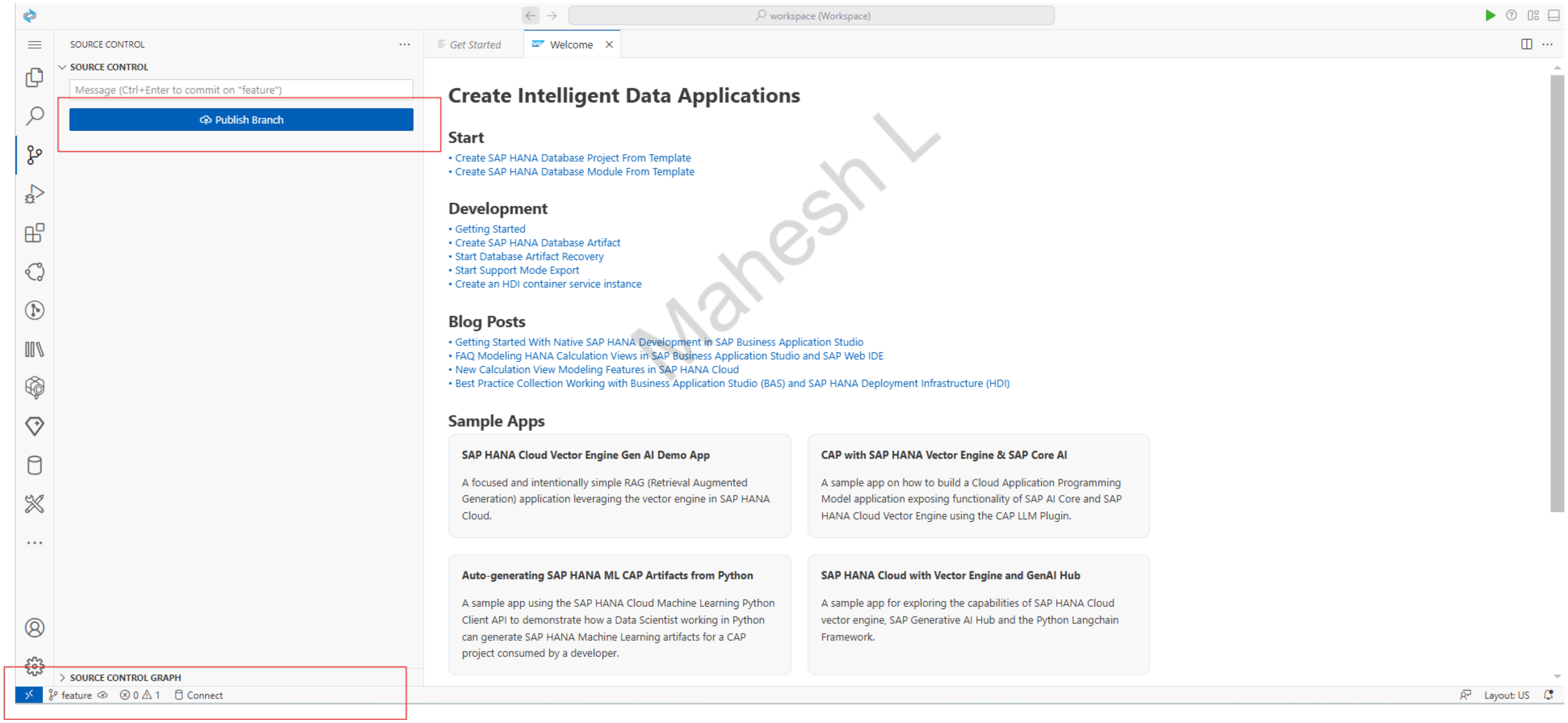
A sample app using the SAP HANA Cloud Machine Learning Python Client API to demonstrate how a Data Scientist working in Python can generate SAP HANA Machine Learning artifacts for a CAP project consumed by a developer.

SAP HANA Cloud with Vector Engine and GenAI Hub

A sample app for exploring the capabilities of SAP HANA Cloud vector engine, SAP Generative AI Hub and the Python Langchain Framework.

3.5 Publish the Feature Branch

- In **SAP Business Application Studio (BAS)**, open the **Source Control** panel.
- Ensure you are on the **feature** branch (as selected in step 3.4).
- Click on the **"Publish Branch"** button to push the feature branch to GitHub.
- This makes the branch available in the remote repository for collaboration.



Create Intelligent Data Applications

Start

- Create SAP HANA Database Project From Template
- Create SAP HANA Database Module From Template

Development

- Getting Started
- Create SAP HANA Database Artifact
- Start Database Artifact Recovery
- Start Support Mode Export
- Create an HDI container service instance

Blog Posts

- Getting Started With Native SAP HANA Development in SAP Business Application Studio
- FAQ Modeling HANA Calculation Views in SAP Business Application Studio and SAP Web IDE
- New Calculation View Modeling Features in SAP HANA Cloud
- Best Practice Collection Working with Business Application Studio (BAS) and SAP HANA Deployment Infrastructure (HDI)

Sample Apps

SAP HANA Cloud Vector Engine Gen AI Demo App

A focused and intentionally simple RAG (Retrieval Augmented Generation) application leveraging the vector engine in SAP HANA Cloud.

CAP with SAP HANA Vector Engine & SAP Core AI

A sample app on how to build a Cloud Application Programming Model application exposing functionality of SAP AI Core and SAP HANA Cloud Vector Engine using the CAP LLM Plugin.

Auto-generating SAP HANA ML CAP Artifacts from Python

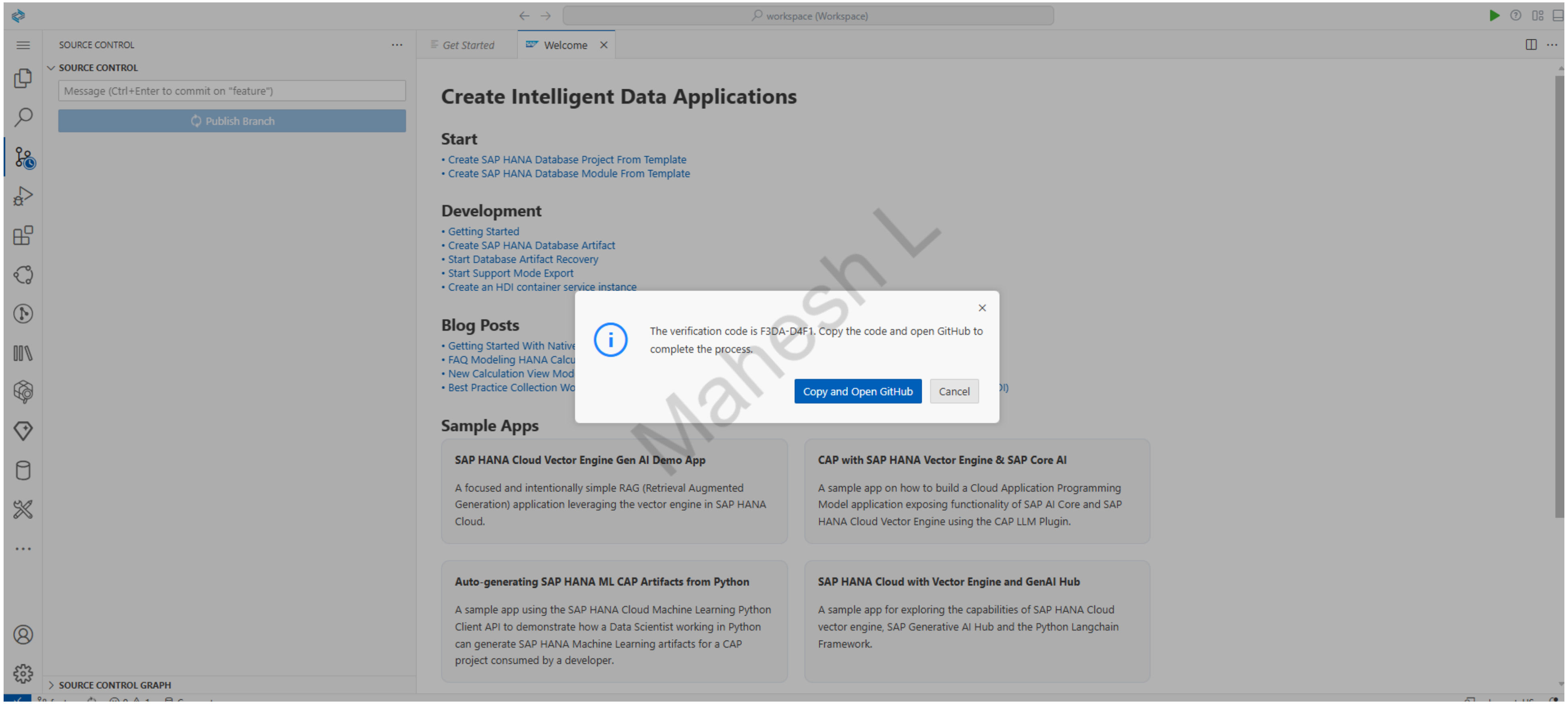
A sample app using the SAP HANA Cloud Machine Learning Python Client API to demonstrate how a Data Scientist working in Python can generate SAP HANA Machine Learning artifacts for a CAP project consumed by a developer.

SAP HANA Cloud with Vector Engine and GenAI Hub

A sample app for exploring the capabilities of SAP HANA Cloud vector engine, SAP Generative AI Hub and the Python Langchain Framework.

3.6 Verify the Branch on GitHub

- Once the feature branch is published, **GitHub Code** will be displayed in BAS.
- Open **GitHub** in your browser.



Create Intelligent Data Applications

Start

- Create SAP HANA Database Project From Template
- Create SAP HANA Database Module From Template

Development

- Getting Started
- Create SAP HANA Database Artifact
- Start Database Artifact Recovery
- Start Support Mode Export
- Create an HDI container service instance

Blog Posts

- Getting Started With Native
- FAQ Modeling HANA Calcu
- New Calculation View Mod
- Best Practice Collection Wo

The verification code is F3DA-D4F1. Copy the code and open GitHub to complete the process.

Sample Apps

SAP HANA Cloud Vector Engine Gen AI Demo App

A focused and intentionally simple RAG (Retrieval Augmented Generation) application leveraging the vector engine in SAP HANA Cloud.

CAP with SAP HANA Vector Engine & SAP Core AI

A sample app on how to build a Cloud Application Programming Model application exposing functionality of SAP AI Core and SAP HANA Cloud Vector Engine using the CAP LLM Plugin.

Auto-generating SAP HANA ML CAP Artifacts from Python

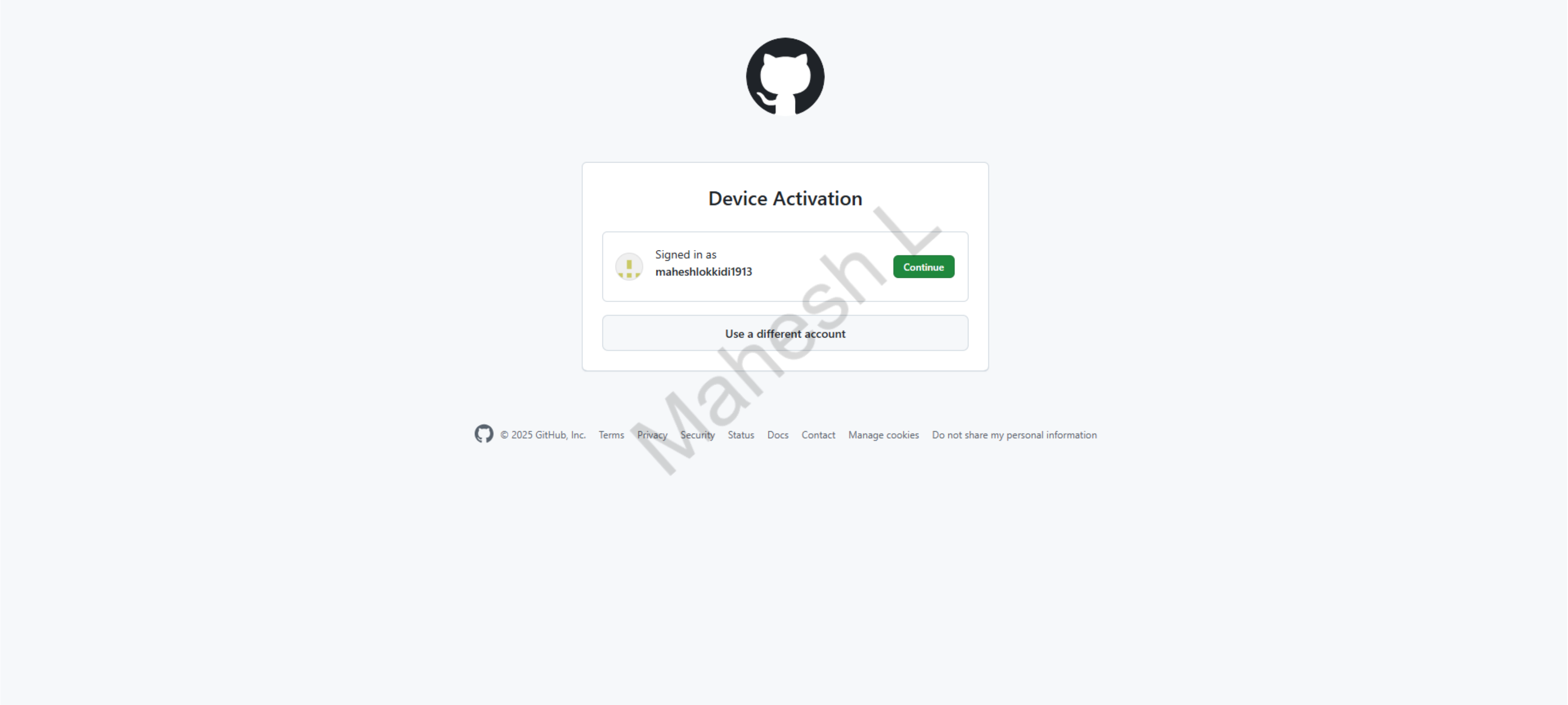
A sample app using the SAP HANA Cloud Machine Learning Python Client API to demonstrate how a Data Scientist working in Python can generate SAP HANA Machine Learning artifacts for a CAP project consumed by a developer.

SAP HANA Cloud with Vector Engine and GenAI Hub

A sample app for exploring the capabilities of SAP HANA Cloud vector engine, SAP Generative AI Hub and the Python Langchain Framework.

3.7 Device Activation and Authorization

- When connecting BAS with GitHub for the first time, **GitHub may ask for authentication**.
- A **GitHub authentication window** will appear, asking you to sign in.
- If prompted, enter the **GitHub authentication code** displayed in BAS.

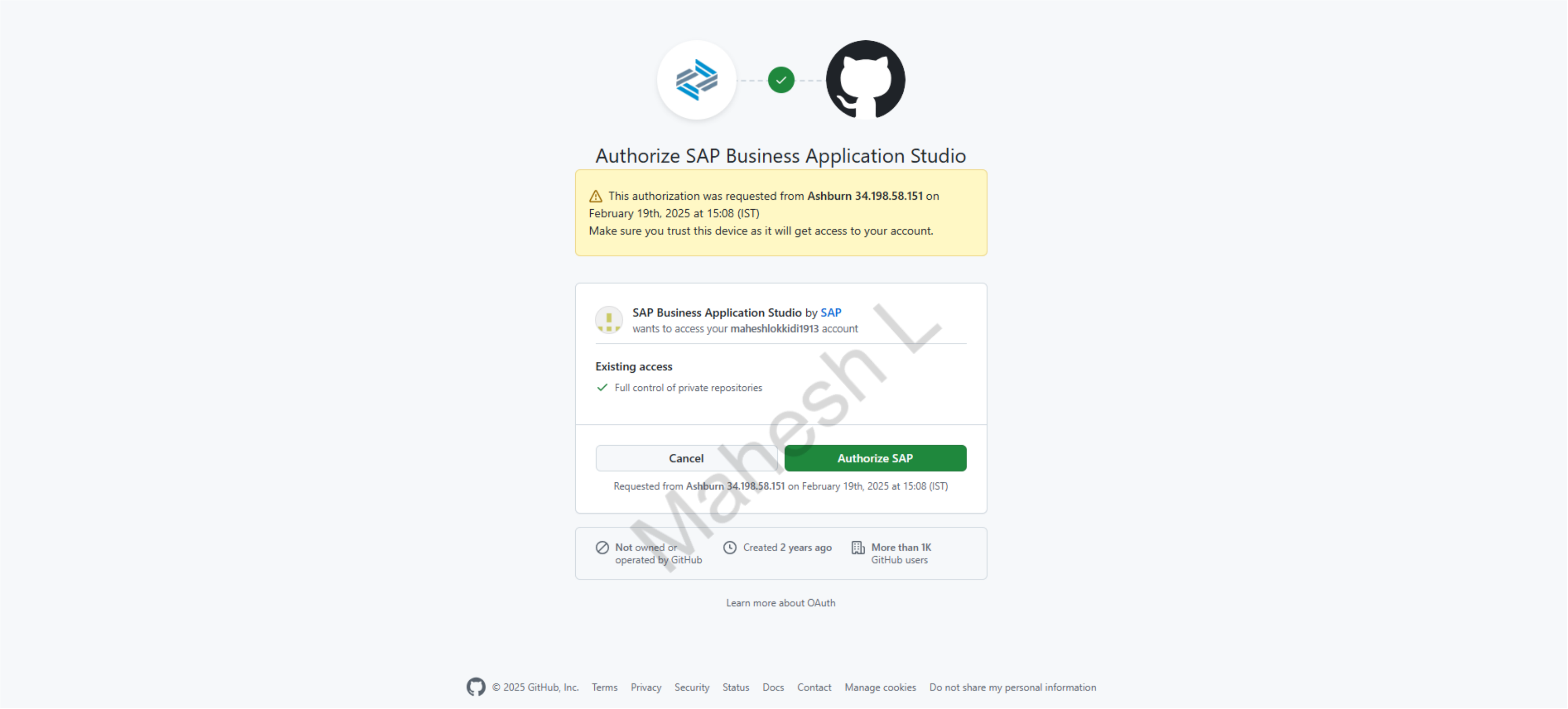


Device Activation

Signed in as
maheshlokkidi1913

Continue
Use a different account
© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

- Click **Authorize** to grant BAS access to your GitHub repository.



Authorize SAP Business Application Studio

⚠ This authorization was requested from **Ashburn 34.198.58.151** on February 19th, 2025 at 15:08 (IST)
Make sure you trust this device as it will get access to your account.

SAP Business Application Studio by SAP
wants to access your **maheshlokkidi1913** account

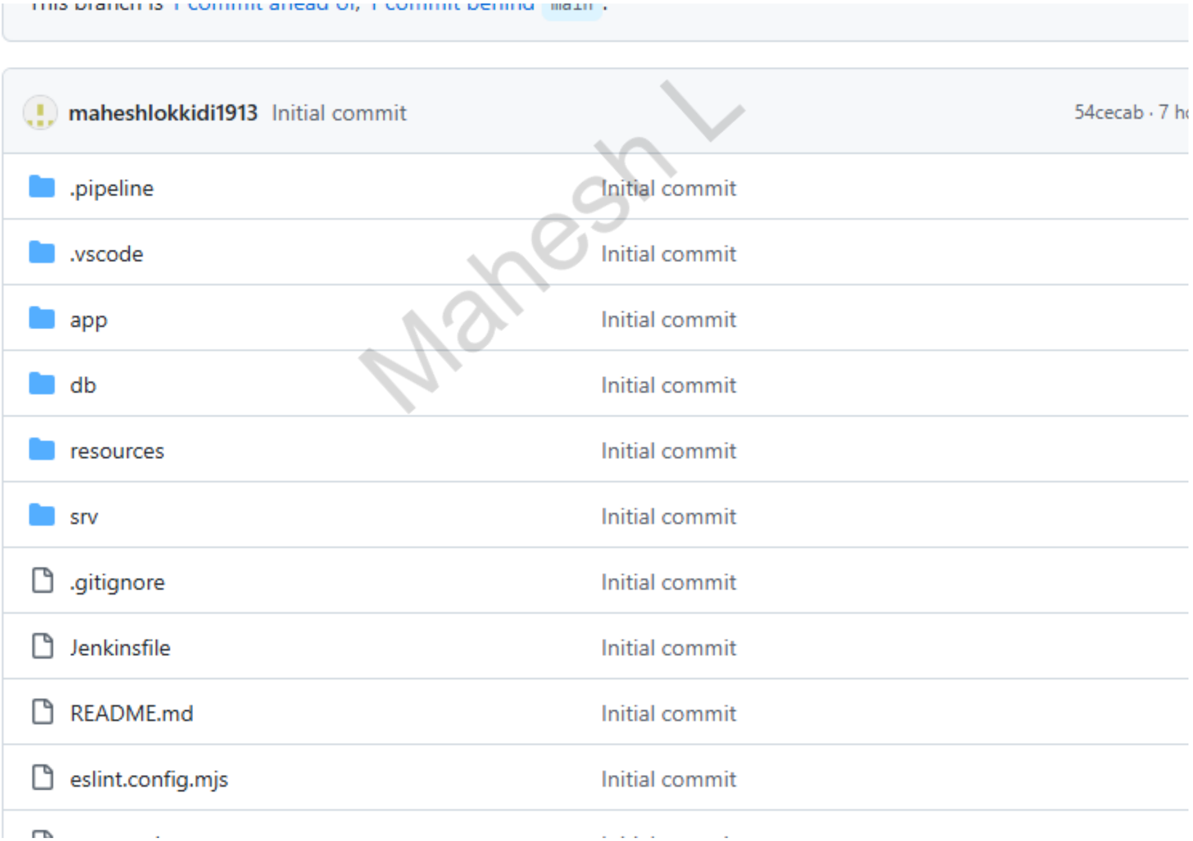
Existing access
✓ Full control of private repositories

Requested from **Ashburn 34.198.58.151** on February 19th, 2025 at 15:08 (IST)

Not owned or operated by GitHub
Created 2 years ago
More than 1K GitHub users
Learn more about OAuth
© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

3.8 Verify Your Project in the Feature Branch

- After successful authentication, go back to **GitHub**.
- Navigate to your **repository**.
- Switch to the **feature branch** using the **branch dropdown**.
- You should now see your **project files** in the **feature branch**.



This branch is 1 commit ahead of, 1 commit behind main.

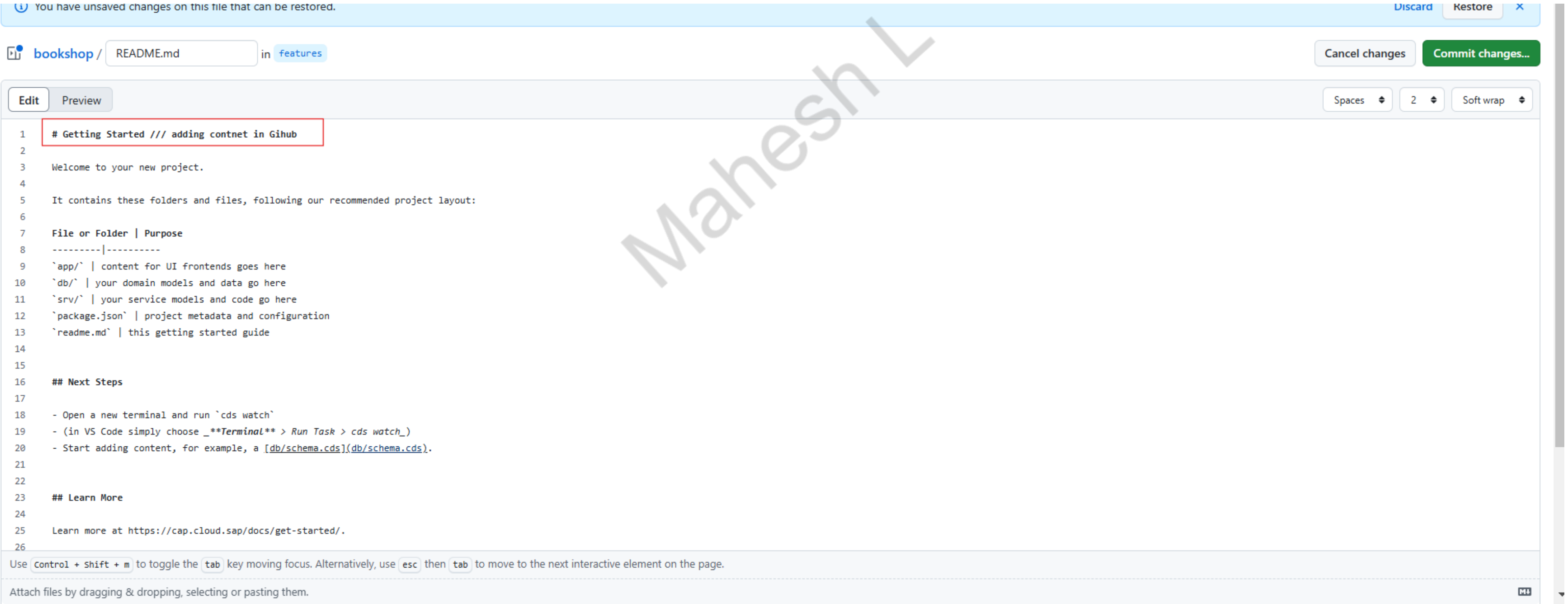
maheshlokkidi1913 Initial commit 54cecab · 7 h

.pipeline	Initial commit
.vscode	Initial commit
app	Initial commit
db	Initial commit
resources	Initial commit
srv	Initial commit
.gitignore	Initial commit
Jenkinsfile	Initial commit
README.md	Initial commit
eslint.config.mjs	Initial commit

4.0 Pull Code from GitHub

4.1 Edit the README File

- In **GitHub**, navigate to your repository.
- Open the **README.md** file.
- Click on the **edit (pencil) icon**.
- Add some text to the file (e.g., "This is a test update").



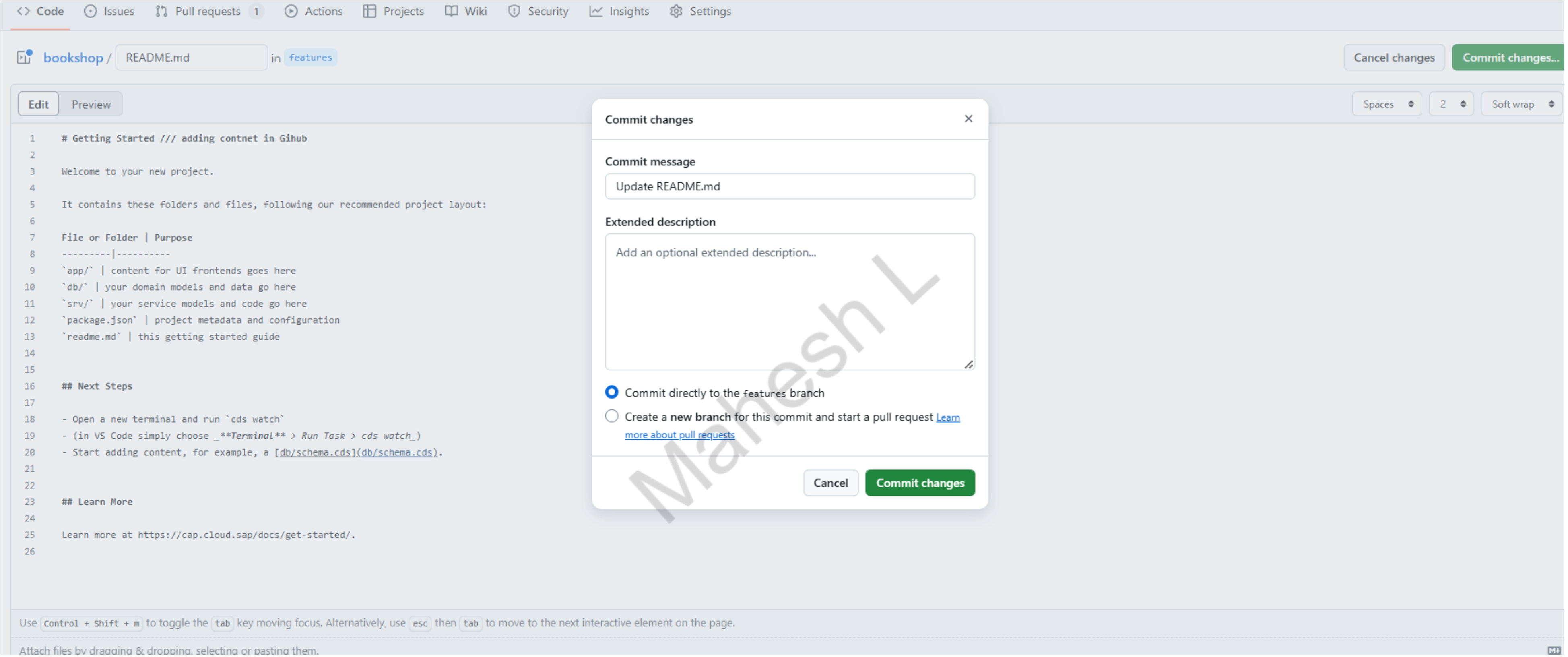
You have unsaved changes on this file that can be restored.

```
# Getting Started /// adding contnet in Gihub

Welcome to your new project.

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).


## Learn More

Learn more at https://cap.cloud.sap/docs/get-started/.

```

Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Attach files by dragging & dropping, selecting or pasting them.

- Click **"Commit changes"** to save the file.



```
# Getting Started /// adding contnet in Gihub

Welcome to your new project.

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).


## Learn More

Learn more at https://cap.cloud.sap/docs/get-started/.

```

Commit changes

Commit message: Update README.md

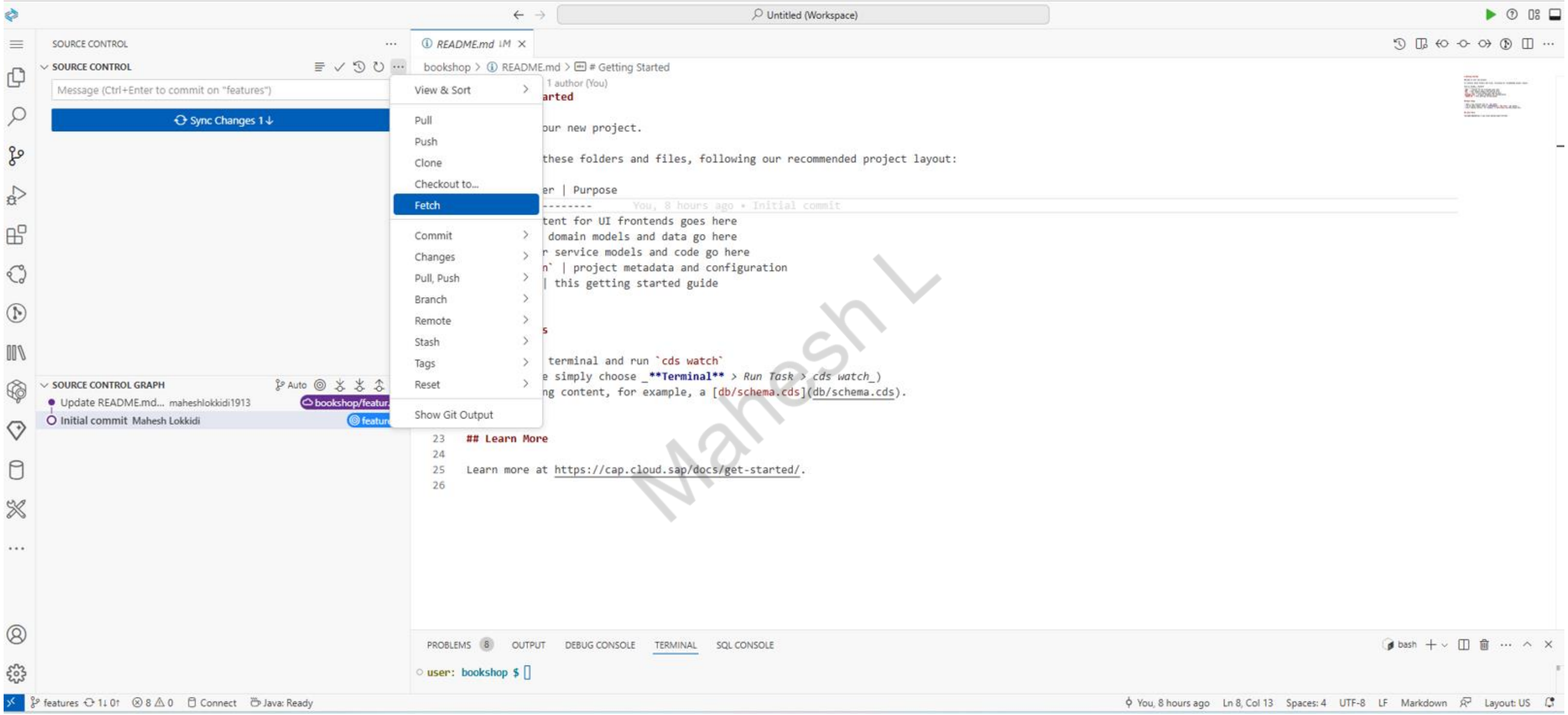
Extended description: Add an optional extended description...

● Commit directly to the `features` branch

○ Create a **new branch** for this commit and start a pull request Learn more about pull requests

Use `control + shift + m` to toggle the `tab` key moving focus. Alternatively, use `esc` then `tab` to move to the next interactive element on the page.

Attach files by dragging & dropping, selecting or pasting them.

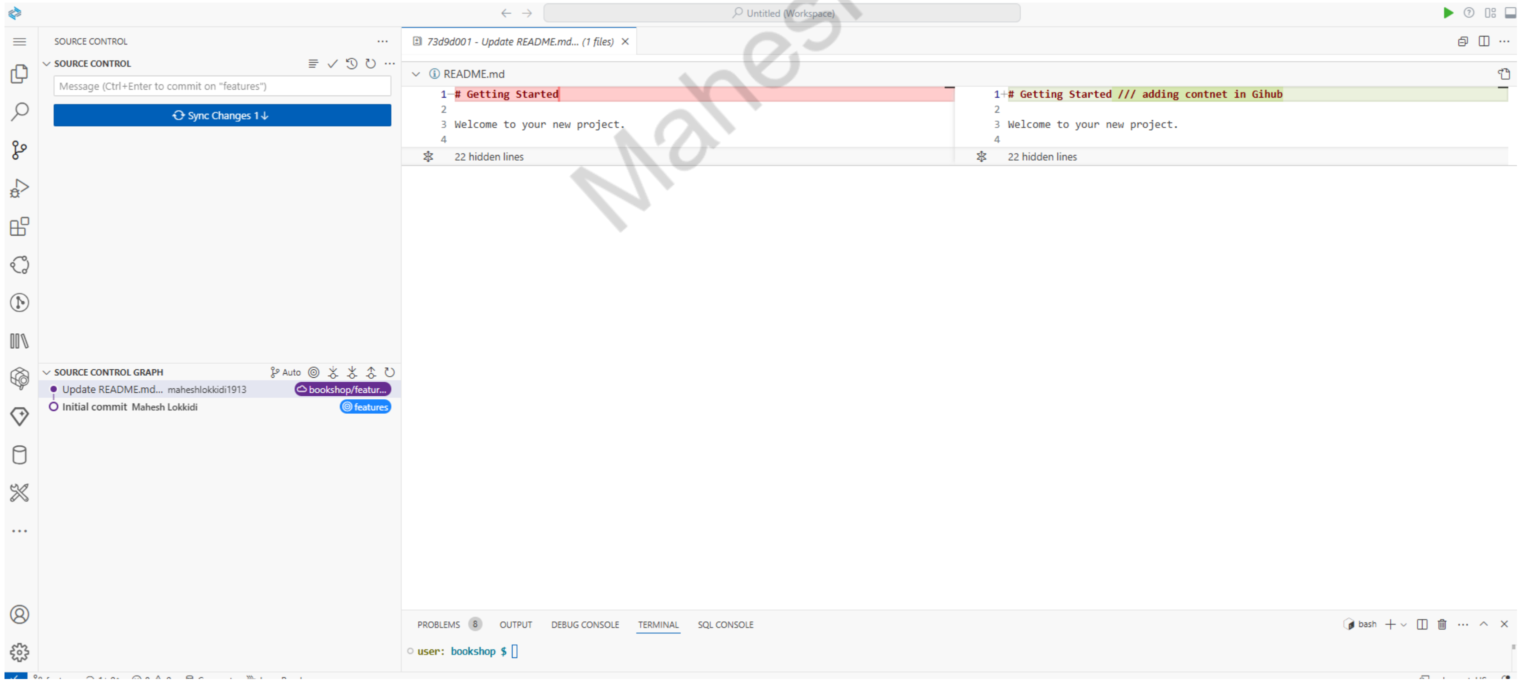


```
arted

our new project.

these folders and files, following our recommended project layout:

er | Purpose
--------
tent for UI frontends goes here
 domain models and data go here
r service models and code go here
n` | project metadata and configuration
| this getting started guide


s

 terminal and run `cds watch`
e simply choose _**Terminal** > Run Task > cds watch_)
ng content, for example, a [db/schema.cds](db/schema.cds).

## Learn More

Learn more at https://cap.cloud.sap/docs/get-started/.
```

```
user: bookshop $
```

4.2 Fetch and Compare Code in BAS

- Open **SAP Business Application Studio (BAS)**.
- Go to the **Source Control** panel.
- Click on **"Fetch"** to pull the latest changes from GitHub.
- Below the **Source Control** section, a **graph** will appear showing file updates.
- Click on the updated file (e.g., README.md).
- The file will open in a **comparison view**, displaying:
  - **Old Version Code** (left side).
  - **New Version Code** (right side with recent changes)
- Click on the Sync Changes



Update README.md... maheshlokkidi1913

Initial commit Mahesh Lokkidi

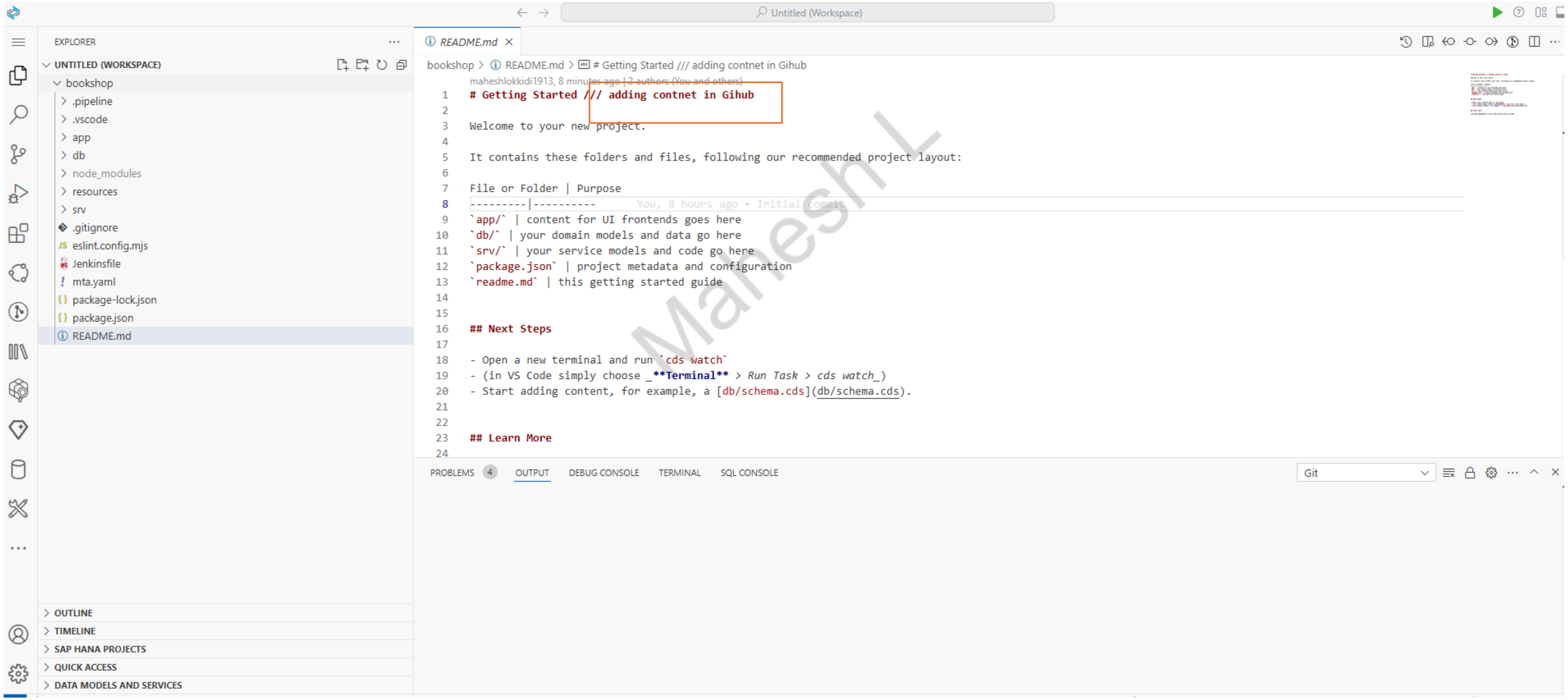
```
# Getting Started

Welcome to your new project.

```

```
# Getting Started /// adding contnet in Gihub

Welcome to your new project.

```

```
user: bookshop $
```

4.3 Sync Changes and Verify Updates

- After syncing, open the **README.md** file in BAS.
- You will now see the changes made in GitHub reflected in the file.

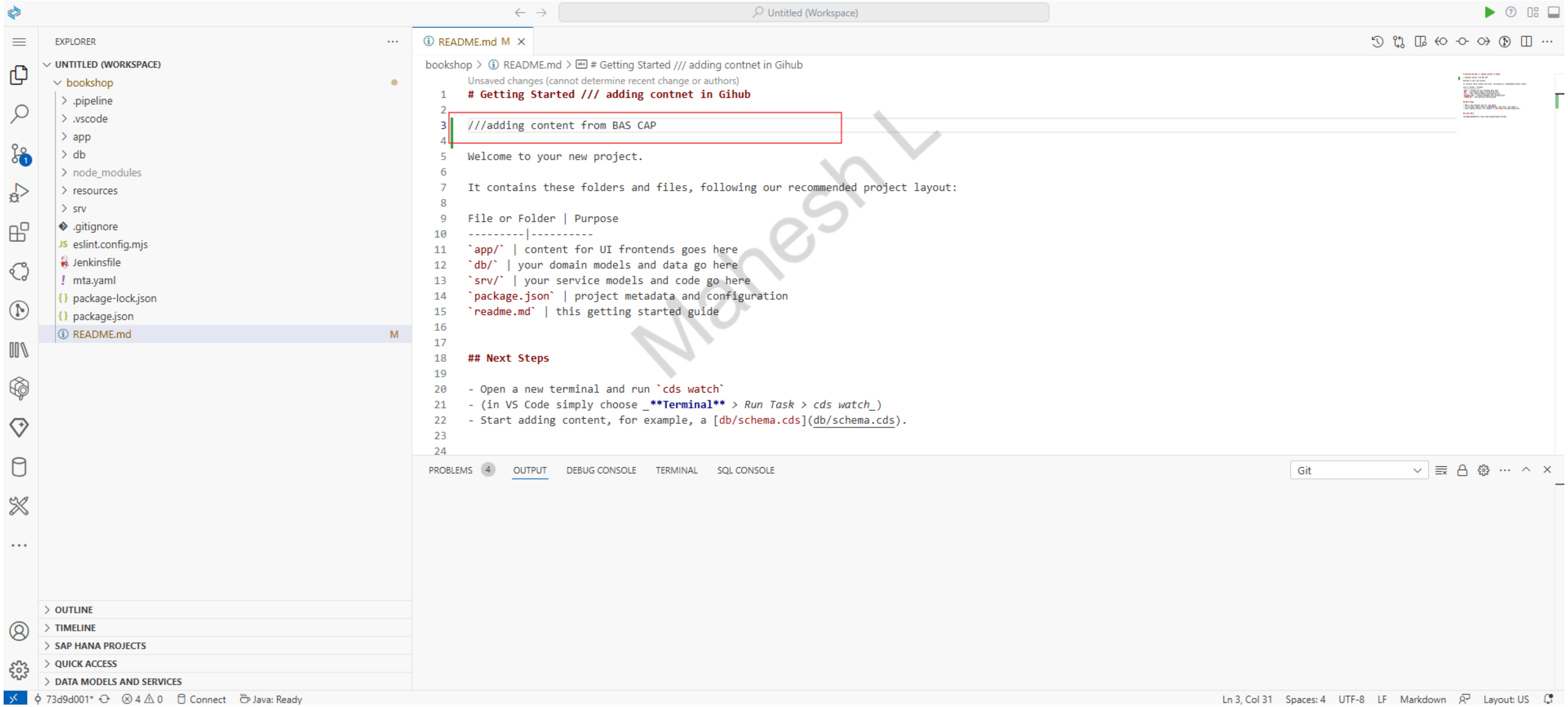


```
# Getting Started /// adding contnet in Gihub

Welcome to your new project.

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).


## Learn More

```

5.0 Push Code to GitHub

5.1 Modify the README File

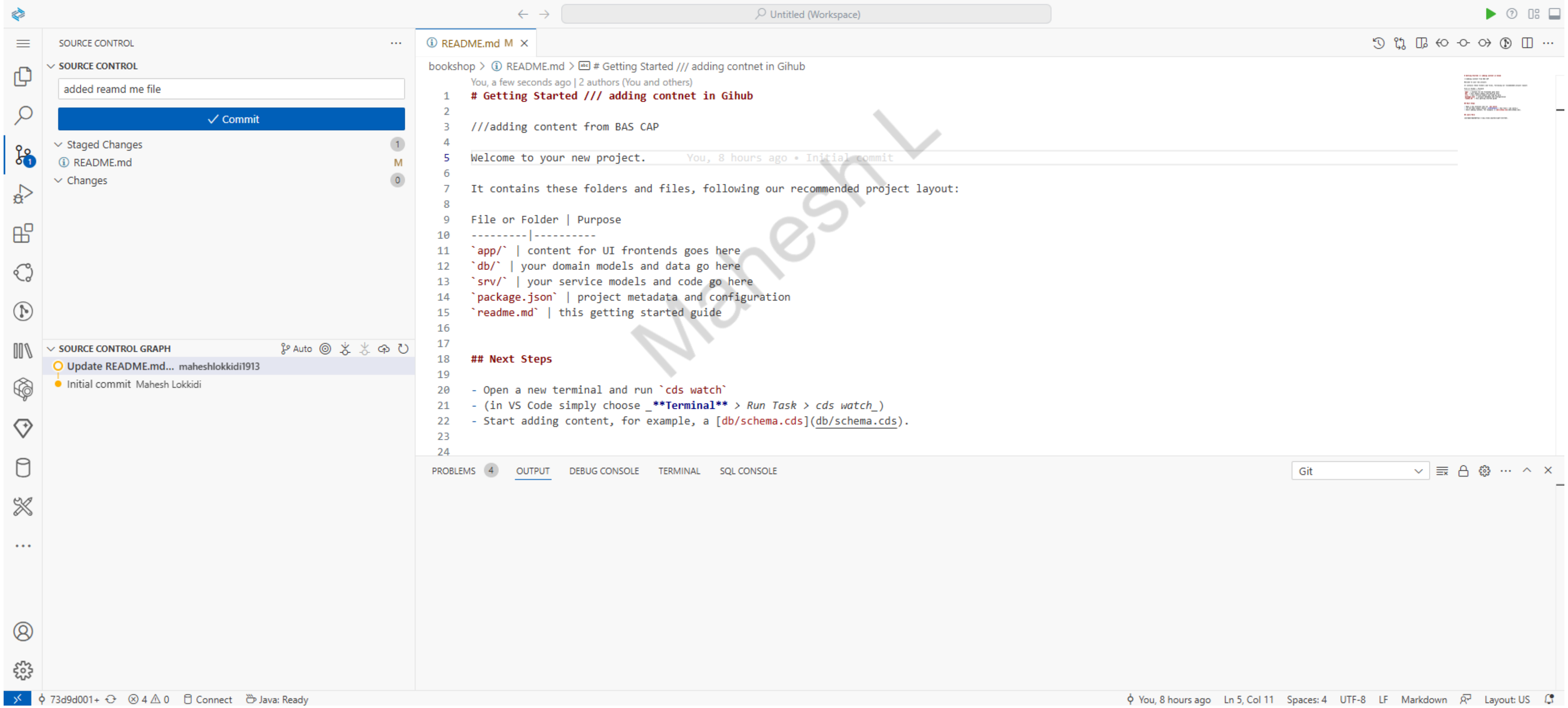
- Navigate to the **README.md** file.
- Add some content (e.g., "This is a new update from BAS").
- Save the file (Ctrl + S or Cmd + S).



```
# Getting Started /// adding contnet in Gihub

///adding content from BAS CAP

Welcome to your new project.

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).
```

5.2 Stage and Commit Changes

- Go to the **Source Control** panel.
- You will see the changes detected in the **README.md** file.
- Click the **"+" (Stage Changes)** button to stage the file.
- In the **commit message box**, enter a meaningful commit message (e.g., "Updated README file").
- Click on **"Commit"** to save the changes locally.



added reamd me file

Update README.md... maheshlokkidi1913

Initial commit Mahesh Lokkidi

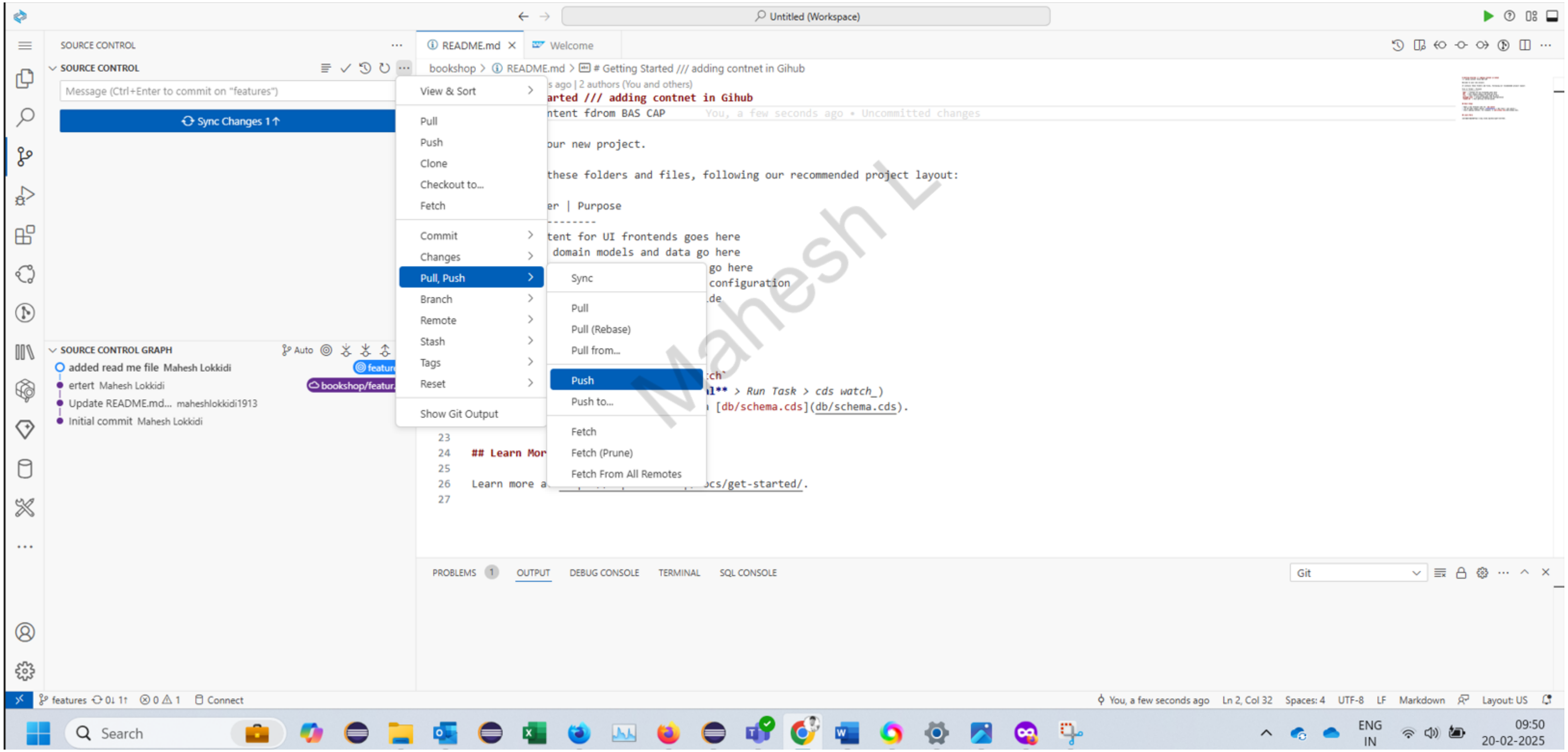
You, a few seconds ago | 2 authors (You and others)

```
# Getting Started /// adding contnet in Gihub

///adding content from BAS CAP

Welcome to your new project.        You, 8 hours ago • Initial commit

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).

```

73d9d001+ ⊗ 4 ⚠ 0 Connect Java: Ready You, 8 hours ago Ln 5, Col 11 Spaces: 4 UTF-8 LF Markdown Layout: US

5.3 Push the Code to GitHub

- In **SAP Business Application Studio (BAS)**, go to the **Source Control** panel.
- Click on the **menu** in the top-right corner.
- Select **"Push"** from the options.
- This will upload your committed changes to the **GitHub repository**.

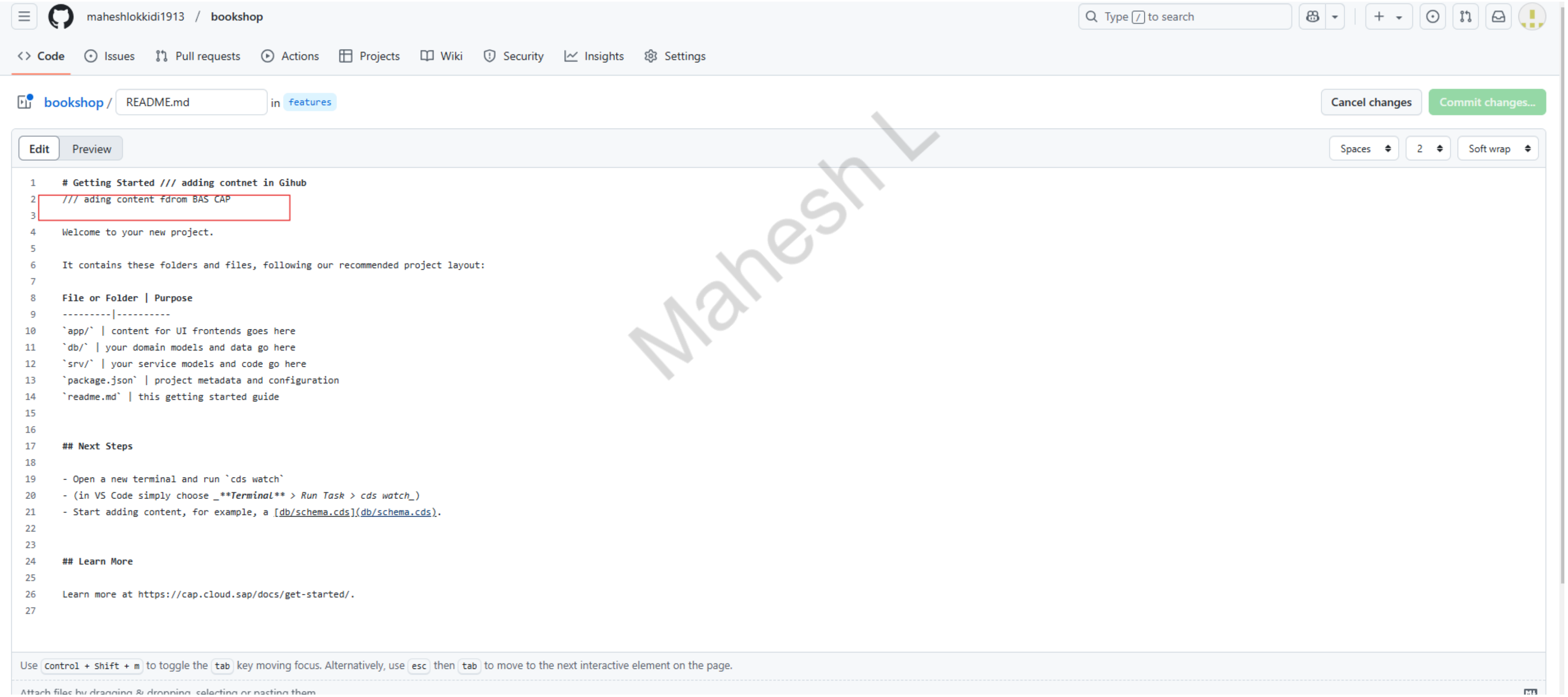


- added read me file Mahesh Lokkidi
- ertert Mahesh Lokkidi
- Update README.md... maheshlokkidi1913
- Initial commit Mahesh Lokkidi

```
arted /// adding contnet in Gihub
ntent fdrom BAS CAP   You, a few seconds ago • Uncommitted changes

our new project.

these folders and files, following our recommended project layout:

er | Purpose
--------
tent for UI frontends goes here
 domain models and data go here
                         go here
                         configuration
                         de

                         ch`
                         l** > Run Task > cds watch_)
                         [db/schema.cds](db/schema.cds).

## Learn Mor

Learn more a                    ocs/get-started/.
```

5.4 Verify Changes on GitHub

- Open **GitHub** in your browser.
- Navigate to your **repository**.
- Open the **README.md** file.
- You should now see the updated content that was pushed from BAS.



```
# Getting Started /// adding contnet in Gihub
/// ading content fdrom BAS CAP

Welcome to your new project.

It contains these folders and files, following our recommended project layout:

File or Folder | Purpose
---------|----------
`app/` | content for UI frontends goes here
`db/` | your domain models and data go here
`srv/` | your service models and code go here
`package.json` | project metadata and configuration
`readme.md` | this getting started guide


## Next Steps

- Open a new terminal and run `cds watch`
- (in VS Code simply choose _**Terminal** > Run Task > cds watch_)
- Start adding content, for example, a [db/schema.cds](db/schema.cds).


## Learn More

Learn more at https://cap.cloud.sap/docs/get-started/.

```

Use `Control + Shift + m` to toggle the `tab` key moving focus. Alternatively, use `esc` then `tab` to move to the next interactive element on the page.

Attach files by dragging & dropping, selecting or pasting them.

7.0 Merge Feature Branch into Main

7.1 Create a Pull Request

- Open **GitHub** and navigate to your repository.
- Click on the **"Pull Requests"** tab.
- Click **"New pull request"**.
- Select **"feature"** as the source branch and **"main"** as the target branch.
- Click **"Create pull request"** and add a description if needed.

7.2 Merge the Pull Request

- After reviewing the changes, click **"Merge pull request"**.
- Click **"Confirm merge"** to merge the **feature** branch into **main**.
- The code from the **feature** branch is now merged into the **main** branch.

OR

8.0 Merge Feature Branch into Main in BAS

8.1 Switch to the Main Branch

- Open **SAP Business Application Studio (BAS)**.
- Go to the **Source Control** panel.
- Click on the **branch dropdown** (currently on "feature").
- Select **"main"** to switch to the main branch.
- If the main branch is not available locally, fetch it by running:

```
git fetch origin
```

8.2 Merge Feature Branch into Main

- Open the **BAS Terminal** (View → Terminal).
- Run the following command to merge the **feature** branch into **main**:

```
git merge feature
```

- If there are conflicts, resolve them manually in the editor.
- After resolving conflicts, **stage the changes**, commit them, and proceed.

8.3 Push the Merged Code to GitHub

- After merging, push the updated main branch to GitHub by running:

```
git push origin main
```

8.4 Verify the Changes

- Open **GitHub** and navigate to your repository.
- Switch to the **main** branch and check that the changes from **feature** are now available.

Thank you..........