

PART 1 - Work Item Breakdown

1. Create Epics

Example Epics:

User Management

Account Services

Fund Transfer

Transaction History

Security & Compliance

2. Under Each Epic, Create Features

Example for 'User Management' Epic:

User Registration

User Authentication

Password Reset

3. Break Features into User Stories

Example for 'User Registration' Feature:

As a new user, I want to register using email and password.

As an admin, I want to approve user registrations (Optional).

4. Break User Stories into Tasks

Example Tasks for Registration User Story:

Design registration form (Frontend)

Create API for registration (Backend)

Add user details to database

Validate user inputs

Final

Overview

Boards

Work Items

Boards

Backlogs

Sprints

Queries

Delivery Plans

Analytics views

Repos

Pipelines

Test Plans

Artifacts

Project settings

azure21c002 / final / Boards / Work Items

Search

Filter by keyword

Recently updated

+ New Work Item

Open in Queries

Column Options

Import Work Items

Recycle Bin

Types

Assigned to

Status

Area Path

Tags

Comments

Activity Date

ID	Title	Assigned To	Status	Area Path	Tags	Comments	Activity Date
99	filter transactions by date or type (credit/debit)	A.G	Resolved	final			7/7/2025 11:55:36 AM
87	confirmation after a successful fund transfer	A.G	Active	final			7/7/2025 11:55:22 AM
69	create a savings or current account	A.G	Resolved	final			7/7/2025 11:55:17 AM
51	receive a verification email after registration	A.G	Active	final			7/7/2025 11:55:12 AM
111	track all user activities for audit purposes	A.G	Active	final			7/7/2025 11:55:04 AM
116	Implement log rotation or archiving	A.G	New	final			7/7/2025 11:29:40 AM
115	Create log review UI (admin only)	A.G	New	final			7/7/2025 11:28:36 AM
114	Store logs securely with timestamp	A.G	New	final			7/7/2025 11:28:13 AM
113	Record key actions (login, transfer, updates)	A.G	New	final			7/7/2025 11:27:55 AM
112	Create logging mechanism	A.G	New	final			7/7/2025 11:27:27 AM
110	Perform security audit	A.G	New	final			7/7/2025 11:26:07 AM
109	Implement secure key storage	A.G	New	final			7/7/2025 11:25:38 AM
108	Secure database connections	A.G	New	final			7/7/2025 11:24:50 AM
107	Encrypt stored data	A.G	New	final			7/7/2025 11:24:20 AM
106	Use hashing algorithm for passwords	A.G	New	final			7/7/2025 11:23:51 AM
104	Display summary of total credits/debits	A.G	New	final			7/7/2025 11:21:23 AM
103	Handle empty results	A.G	New	final			7/7/2025 11:20:57 AM
102	Modify backend API to accept filters	A.G	New	final			7/7/2025 11:20:31 AM
101	Modify backend API to accept filters	A.G	New	final			7/7/2025 11:20:03 AM
100	Add filter options (date, type) in UI	A.G	New	final			7/7/2025 11:19:32 AM
98	Format timestamps and amounts	A.G	New	final			7/7/2025 11:15:03 AM
97	Display data in table format	A.G	New	final			7/7/2025 11:14:37 AM
96	Implement pagination or scrolling	A.G	New	final			7/7/2025 11:14:16 AM
95	Create API to fetch transaction data	A.G	New	final			7/7/2025 11:13:54 AM
94	Build transaction history UI	A.G	New	final			7/7/2025 11:13:34 AM
93	see a list of my past transactions	A.G	New	final			7/7/2025 11:13:10 AM
90	Handle error for insufficient balance	A.G	New	final			7/7/2025 11:12:38 AM
91	Update transaction history	A.G	New	final			7/7/2025 11:12:09 AM
90	Send confirmation email/SMS	A.G	New	final			7/7/2025 11:11:48 AM
89	Display transfer summary (amount, recipient, timestamp)	A.G	New	final			7/7/2025 11:11:27 AM
88	Generate transaction confirmation message	A.G	New	final			7/7/2025 11:10:53 AM
85	Transfer Confirmation	Unassigned	New	final			7/7/2025 11:08:15 AM
86	Store transfer record in database	A.G	New	final			7/7/2025 11:07:16 AM
85	Add amount to receiver's account	A.G	New	final			7/7/2025 11:06:54 AM
84	Deduct amount from sender's account	A.G	New	final			7/7/2025 11:06:34 AM
83	Validate input fields (amount > 0, account exists)	A.G	New	final			7/7/2025 11:06:11 AM
82	Design transfer form (account number, amount, remarks)	A.G	New	final			7/7/2025 11:05:48 AM
81	Transfer funds to another account.	A.G	New	final			7/7/2025 11:05:18 AM
80	Format currency display	A.G	New	final			7/7/2025 11:04:48 AM
79	Handle session/authentication check	A.G	New	final			7/7/2025 11:04:26 AM
78	Show balance on dashboard	A.G	New	final			7/7/2025 11:04:07 AM
77	Build API to fetch balance from the database	Unassigned	New	final			7/7/2025 11:03:39 AM
76	Create balance display UI	A.G	New	final			7/7/2025 11:03:07 AM
74	Notify user of successful account creation	A.G	New	final			7/7/2025 11:01:05 AM
73	Generate unique account number	A.G	New	final			7/7/2025 11:00:11 AM
72	Store account details in the database	A.G	New	final			7/7/2025 10:59:39 AM
71	Create API to capture account type and user details	A.G	New	final			7/7/2025 10:59:06 AM
70	Design account creation form	A.G	New	final			7/7/2025 10:58:21 AM
68	Notify user of successful password reset	A.G	New	final			7/7/2025 10:54:35 AM
67	Update password in the database	A.G	New	final			7/7/2025 10:54:02 AM
66	Validate reset token	A.G	New	final			7/7/2025 10:53:55 AM
65	Generate reset token and send email link	A.G	New	final			7/7/2025 10:51:59 AM
64	Create Forgot Password UI	A.G	New	final			7/7/2025 10:51:19 AM
63	reset my password using a verification link sent to my email	A.G	Resolved	final			7/7/2025 10:41:52 AM
62	Display error messages for failed login	A.G	New	final			7/7/2025 10:40:41 AM
61	Generate authentication token/session	A.G	New	final			7/7/2025 10:39:50 AM
60	Validate user credentials against the database	A.G	New	final			7/7/2025 10:38:46 AM
59	Implement backend authentication logic	A.G	New	final			7/7/2025 10:37:43 AM
58	Design login form UI	A.G	New	final			7/7/2025 10:37:06 AM
56	Send confirmation/verification email	A.G	New	final			7/7/2025 10:35:23 AM
55	Validate email format and password strength	A.G	New	final			7/7/2025 10:34:23 AM
54	Store user data in the database	A.G	New	final			7/7/2025 10:33:48 AM
53	Create backend API for registration	A.G	New	final			7/7/2025 10:33:17 AM
52	Design the registration form UI	A.G	New	final			7/7/2025 10:32:45 AM
50	register using email and password	A.G	New	final			7/7/2025 10:26:12 AM
49	Audit Logs	A.G	New	final			7/7/2025 10:23:47 AM
48	Encryption Sensitive Data	A.G	New	final			7/7/2025 10:23:09 AM
47	Filter Transactions	A.G	New	final			7/7/2025 10:22:18 AM
46	Display Past Transactions	A.G	New	final			7/7/2025 10:21:19 AM
44	Initiate Transfer	A.G	New	final			7/7/2025 10:19:22 AM
43	View Account Balan	A.G	New	final			7/7/2025 10:18:32 AM
42	Account Creation	A.G	New	final			7/7/2025 10:17:44 AM
41	Password Reset	A.G	New	final			7/7/2025 10:17:09 AM
40	User Authentication	A.G	New	final			7/7/2025 10:15:43 AM
39	User Registration	A.G	New	final			7/7/2025 10:15:16 AM
38	Security and Compliance	A.G	New	final			7/7/2025 10:14:37 AM
37	Transaction History	A.G	New	final			7/7/2025 10:09:18 AM
36	Fund Transfer	A.G	New	final			7/7/2025 9:57:17 AM
35	Account Services	A.G	New	final			7/7/2025 9:56:39 AM
34	User Management	A.G	New	final			7/7/2025 9:54:46 AM

Azure DevOps aaka21cs002 / final / Boards / Boards

Search

final

Overview

Boards

Work items

Boards

Backlogs

Sprints

Queries

Delivery Plans

Analytics views

Repos

Pipelines

Test Plans

Artifacts

Project settings

final Team

Board Analytics

Stories

View as backlog

New

Active 3/5

Resolved 3/5

Closed

+ New item

50 register using email and password

New

A G

Q/4

81 transfer funds to another account.

New

A G

Q/5

93 see a list of my past transactions

New

A G

Q/5

111 track all user activities for audit purposes

Active

A G

Q/5

51 receive a verification email after registration

Active

A G

Q/1

87 confirmation after a successful fund transfer

Active

A G

Q/5

69 create a savings or current account

Resolved

A G

Q/5

63 reset my password using a verification link sent to my email

Resolved

A G

Q/5

99 filter transactions by date or type (credit/debit)

Resolved

A G

Q/5

105 data to be encrypted for security

Closed

A G

Q/5

75 view my current account balance

Closed

A G

Q/5

57 log in using valid credentials.

Closed

A G

Q/5

Boards

PART 2 - Azure DevOps Git Setup

1. Create a Git Repo

Azure DevOps → Repos → New Repository → Name it banking-app

Clone repo locally or use portal

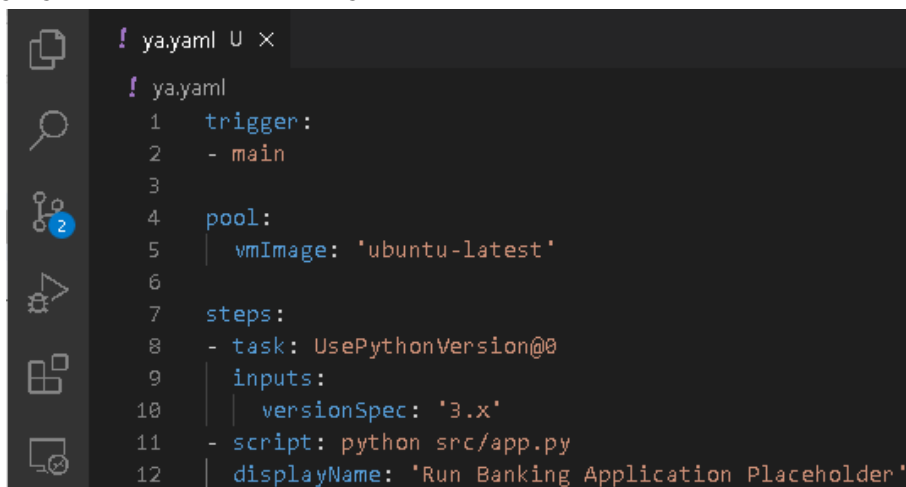
Add a basic project structure:

```
banking-app/  
├── README.md  
├── src/  
│   └── app.py # Placeholder Python file  
└── .azure-pipelines.yml
```

app.py can simply contain:

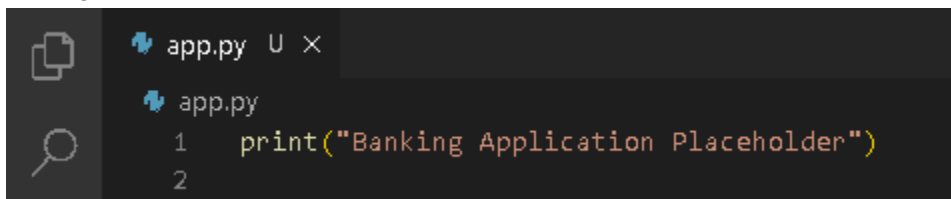
```
print("Banking Application Placeholder")
```

ya.yaml (azure-pipelines.yml)



```
! ya.yaml U ×  
! ya.yaml  
1   trigger:  
2     - main  
3  
4   pool:  
5     vmImage: 'ubuntu-latest'  
6  
7   steps:  
8     - task: UsePythonVersion@0  
9       inputs:  
10        versionSpec: '3.x'  
11     - script: python src/app.py  
12       displayName: 'Run Banking Application Placeholder'
```

app.py



```
app.py U ×  
app.py  
1   print("Banking Application Placeholder")  
2
```

Banking-app

src

- app.py
- README.md
- ya.yaml

Name	Last change	Commits
src	27m ago	26e9e27d Added app.py A G
README.md	27m ago	d5158f7c Added README.md A G
ya.yaml	26m ago	a7731861 Added ya.yaml A G

Introduction

TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

Getting Started

TODO: Guide users through getting your code up and running on their own system. In this section you can talk about:

1. Installation process
2. Software dependencies
3. Latest releases
4. API references

Build and Test

TODO: Describe and show how to build your code and run the tests.

Contribute

TODO: Explain how other users and developers can contribute to make your code better.

If you want to learn more about creating good readme files then refer the following [guidelines](#). You can also seek inspiration from the below readme files:

- [ASP.NET Core](#)
- [Visual Studio Code](#)
- [Chakra Core](#)

Azure devops repo (Files)

Banking-app

src

- app.py
- README.md
- ya.yaml

Graph	Commit	Pull Request	Status
	Added ya.yaml a7731861 A G Today at 11:51 am		failed
	Added app.py 26e9e27d A G Today at 11:50 am		
	Added README.md d5158f7c A G Today at 11:50 am		

Azure devops repo (Files)

PART 3 - Configure CI/CD Pipeline

1. Create Simple Pipeline

Example .azure-pipelines.yml :

trigger:

- main

pool:

vmImage: 'ubuntu-latest'

steps:

- task: UsePythonVersion@0

inputs:

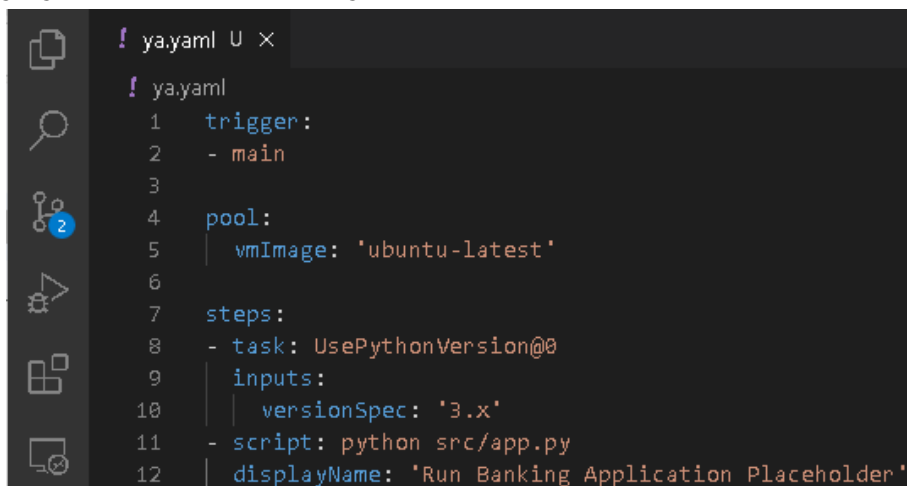
versionSpec: '3.x'

- script: python src/app.py

displayName: 'Run Banking Application Placeholder'

Pipeline succeeds if the script runs without errors.

ya.yaml (azure-pipelines.yml)



```
! ya.yaml U x
! ya.yaml
1 trigger:
2   - main
3
4 pool:
5   vmImage: 'ubuntu-latest'
6
7 steps:
8   - task: UsePythonVersion@0
9     inputs:
10      versionSpec: '3.x'
11   - script: python src/app.py
12     displayName: 'Run Banking Application Placeholder'
```

Azure DevOps aaka21c002 / final / Pipelines

Search

final +

- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Library
- Test Plans
- Artifacts

Project settings <<


Pipelines

Recent All Runs

Filter pipelines

New pipeline

Recently run pipelines

Pipeline	Last run
 Banking-app	#20250707.1 • Added ya.yaml Manually run by main 28m ago <1s

Azure devops pipelines

Azure DevOps aaka21c002 / final / Pipelines / Banking-app

Search

final +



- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Library
- Test Plans
- Artifacts

Project settings <<

Banking-app

Edit Run pipeline

Runs Branches Analytics

Description	Stages
 #20250707.1 • Added ya.yaml Manually run by main a7733861	

Azure devops pipeline runs

Azure DevOps

aka21cs002

final

Pipelines

Banking-app

20250707.1

Search

AG

final

+

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

<<

#20250707.1 • Added ya.yaml

Banking-app

Rerun failed jobs

Run new

This run will be cleaned up after 1 month based on your project settings.

Summary

Code Coverage

Manually run by AG

Repository and version

Banking-app

main

7731861

Time started and elapsed

Today at 11:52 am

<1s

Related

0 work items

0 artifacts

Tests and coverage

Get started

View 3 changes

Errors

1

No hosted parallelism has been purchased or granted. To request a free parallelism grant, please fill out the following form <https://aka.ms/azpipelines-parallelism-request>

20250707.1

View documentation for troubleshooting failed runs

Jobs

Name	Status	Duration
Job	Failed	

Azure devops pipeline run details