

DEVOPS PROJECT

MAKE YOUR WISHES COME TRUE

- ZABOUJ Chadi
- ZEJLI Mohamed-Nadir
- HANINE Hamza
- MAAYOU Adil

3. User Stories

1. As a user, I want to write a daily journal to organize my thoughts and emotions.

Description

- Users should be able to create, edit, and view their daily journal entries.
- Journals should be stored and retrievable for future reference.

Acceptance Criteria

1. A feature to add a journal entry with a date.
2. Users can update or delete past entries.
3. Entries are displayed in chronological order.

2. As a user, I want to track home decoration tasks to create a pleasant living space.

Description

- Users should be able to add, edit, and complete decoration tasks.
- Progress tracking should be available for completed vs. pending tasks.

Acceptance Criteria

1. Users can add new tasks with descriptions.
2. Tasks can be marked as "complete."
3. A progress bar visually represents task completion.

3. As a user, I want daily reminders for self-care tasks to prioritize my well-being.

Description

- Users should receive notifications to perform self-care activities such as exercise, meditation, and eating healthy.

Acceptance Criteria

1. Users receive reminders for specific tasks (e.g., "Drink water").
2. Completed tasks are logged and displayed in a weekly summary.

4. As a user, I want daily suggestions for acts of kindness to make a positive impact

Description:

- Users should receive random suggestions for simple acts of kindness, such as helping someone or giving a compliment.

Acceptance Criteria:

- Generate random kindness suggestions daily.
- Users can log completed acts of kindness.

5. As a user, I want to be encouraged to try new activities to expand my experiences.

Description:

- The system should suggest random new activities based on predefined options (e.g., "Run a marathon" or "Learn a new skill").

Acceptance Criteria:

- Generate random activity suggestions

6. As a user, I want reminders to spend time with loved ones.

Description:

- The system should help users plan and remind them about events with family and friends.

Acceptance Criteria:

- Users can add events with a date.
- Reminders are sent before the event.

7. As a user, I want recommendations for concerts to enjoy live music with friends.

The system suggests concerts or music events based on user preferences.

Acceptance Criteria:

Suggest local concerts using a mock list.
Users can save events to their calendar.

8. As a user, I want reminders to plan a night under the stars for a unique experience.

The system should provide reminders and tips for planning a night outdoors (e.g., bring blankets, check the weather).

Acceptance Criteria:

Notifications include tips for a night outdoors.
Users can mark the experience as completed.

9. As a user, I want suggestions for outdoor adventures to explore new terrains.

The system should recommend hiking trails or travel destinations for users to explore.

Acceptance Criteria:

Generate random hiking or travel suggestions.
Users can save suggested destinations.

10. As a user, I want daily study reminders to help me stay consistent with my learning goals.

Description:

- The system should encourage users to acquire new knowledge, research, and explore affordable online learning opportunities.

Acceptance Criteria:

- Users can log study sessions.
- The system sends reminders to study daily.
- Weekly summaries show time spent on studying.

11. As a user, I want daily practice reminders to learn and improve at playing an instrument.

Description:

- Users should receive simple tips to learn instruments like the ukulele, making it fun and engaging.

Acceptance Criteria:

- Users receive daily reminders for practice.
- Progress can be logged (e.g., time practiced).

12. As a user, I want daily mindfulness reminders to practice conscious breathing and reduce stress.

Description:

- The system should provide daily reminders to practice 5 minutes of conscious breathing to help users stay calm and mindful.

Acceptance Criteria:

- Users receive daily reminders for 5-minute breathing exercises.
- The system tracks how often users complete these exercises.

4. Pipeline Details

- **Overview of the Pipeline Structure:**

Each pipeline is designed using Jenkins for CI/CD processes. Every month, a new pipeline is added, which automates code builds, testing, and deployment. Pipelines are configured to handle unique features for each month's user story.

The main structure of the pipelines is 'Plan' 'Design' 'Test' 'Deliver'.

Plan: This stage focuses on defining the goals, requirements, and roadmap for the project.

Design: This stage ensures that the architecture, workflow, and any infrastructure-related aspects are properly set up and validated.

Test: Validate the code and design by running automated tests. (Unit tests, integrated... Etc.)

Deliver: If tests are successful, finalize and deploy the application or deliverables to the target environment.

- Configuration for Jenkins:

The screenshot shows the Jenkins configuration page for a new project. The 'GitHub project' checkbox is checked. The 'Project url' field contains the GitHub repository link: `https://github.com/MNadirZeji/DevopsProjectWishApp.git/`. Below this is an 'Advanced' dropdown menu. The 'Pipeline' section is expanded, showing the 'Definition' dropdown set to 'Pipeline script from SCM'. The 'SCM' dropdown is set to 'Git'. Under 'Repositories', the 'Repository URL' field contains the same GitHub link, and the 'Credentials' dropdown is set to '- none -'. There is an '+ Add' button and an 'Advanced' dropdown. Below the repositories section is an 'Add Repository' button. The 'Branches to build' section is expanded, showing the 'Branch Specifier (blank for 'any')' field set to `*/main`.

Pipeline script from SCM, then github link and branch name. (in our case: “main”).

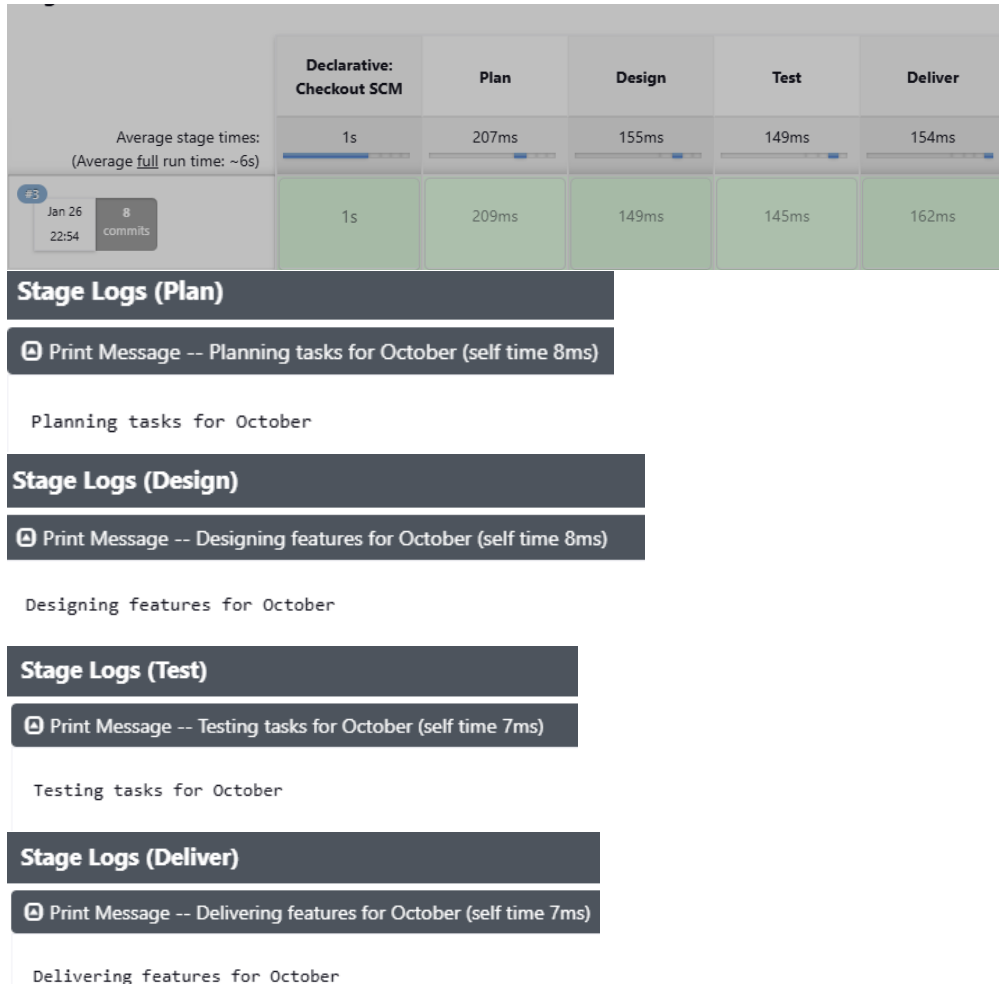
The screenshot shows the 'Script Path' field in the Jenkins configuration. The field contains the path: `.github/workflows/pipeline_Chadi`.

Example of script path for pipeline file to use.

Please refer to [Section 5](#) for final results obtained.

5. Demonstration

- **Final results:**



6. Conclusion

- **Summary of the Project and Outcomes:**

The project successfully delivered 12 functional pipelines, each tailored to a unique user story. Through consistent CI/CD practices, the team ensured high-quality releases each month, contributing to an enriched user experience.

7. Appendices

1. Example Configuration Files

- **Pipeline Script (Jenkinsfile):**

```
version: "1.0"
pipeline:
  parameters:
    - name: MONTH
      description: "Enter the month (July, August, September)"
      type: string
  stages:
    - name: Plan
      steps:
        - script: |
            echo "Planning tasks for ${params.MONTH}"
            if [ "${params.MONTH}" = "July" ]; then
              echo "Identifying recommendations for concerts in
${params.MONTH}."
            elif [ "${params.MONTH}" = "August" ]; then
              echo "Planning a night under the stars in
${params.MONTH}."
            elif [ "${params.MONTH}" = "September" ]; then
              echo "Planning outdoor adventures in ${params.MONTH}."
            else
              echo "Invalid month entered!"
            fi
        - name: Design
          steps:
            - script: |
                echo "Designing features for ${params.MONTH}"
        - name: Test
          steps:
            - script: |
                echo "Testing features for ${params.MONTH}"
        - name: Deliver
          steps:
            - script: |
                echo "Delivering features for ${params.MONTH}"
```

- **Ansible Script:**

```
---
- name: Deploy Monthly Features
  hosts: 8080
  vars:
    month: "{{ month }}"
  tasks:
```

```

- name: Plan tasks for the given month
  debug:
    msg: >
      Planning tasks for {{ month }}.
      {% if month == 'July' %}
      Identifying recommendations for concerts for {{ month
}}}.

      {% elif month == 'August' %}
      Planning a night under the stars for {{ month }}.
      {% elif month == 'September' %}
      Planning outdoor adventures for {{ month }}.
      {% else %}
      Invalid month entered!
      {% endif %}
- name: Design features for the given month
  debug:
    msg: >
      Designing features for {{ month }}.
- name: Test features for the given month
  debug:
    msg: >
      Testing tasks for {{ month }}.
- name: Deliver features for the given month
  debug:
    msg: >
      Delivering features for {{ month }}.

```

2. User Documentation

- **How to Run the Pipeline:**
 - Open your Jenkins dashboard.
 - Navigate to the pipeline job and click “Build with Parameters.”
 - Enter the desired month (e.g., July, August, September) in the **MONTH** field.
 - Click “Build” to start the pipeline.
- **Expected Outputs:**
 - The pipeline will execute tasks for the selected month, including:
 - Planning stage.
 - Designing features.
 - Testing functionality.
 - Delivering final features.
- **Error Handling:**
 - If an invalid month is entered, the pipeline will display:
Invalid month entered!

3. Technical Documentation

- **Setup:**
 - Install Jenkins and ensure it is running.

- Create a new pipeline project in Jenkins and paste the provided Jenkinsfile script.
- **Parameters:**
 - MONTH: A string parameter used to define the workflow for the selected month.
- **Troubleshooting:**
 - Ensure the Jenkinsfile syntax is correct by validating it in the Jenkins pipeline editor.
 - Check console logs for debugging any unexpected behavior.

4. Links to the Project's Git Repository

- Main Repository:
<https://github.com/MNadirZejli/DevopsProjectWishApp/tree/main>
- Branches for Reference:
 - main: Production-ready pipeline scripts.