

Valeria Anel Duque Salgado

NAO ID: 3306

08/09/2025

Github and Digital Respostory Management ***SPRINT 2***

IN-MEXICO PROGRAM BACKEND DEVELOPER
CERTIFICATION

User Story	Requirements
As a developer, I want to create a new GitHub repository so that I can store and manage my project online.	<p>Must provide repository name and settings.</p> <p>Repository must be visible in the account's repository list.</p>
As a developer, I want to connect my local project to a GitHub repository so that I can push my code and make it available remotely.	<p>Local project initialized with git init.</p> <p>Remote repository linked via git remote add origin <URL>.</p> <p>Files added and committed.</p> <p>Code pushed to remote branch.</p>
As a developer, I want to create a new branch so that I can work on features or fixes without affecting the main codebase.	<p>Ability to create branches with git branch branch-name.</p> <p>Ability to confirm with git branch.</p>
As a developer, I want to update the requirements.txt file so that I can define or change the dependencies for my project.	<p>Changes to requirements.txt must be saved locally.</p> <p>Updates must be committed and pushed to the correct branch.</p>
As a developer, I want to create and approve pull requests (PRs) so that I can merge contributions and maintain code quality.	<p>Changes must be committed and pushed before creating PR.</p> <p>PR must show differences between branches.</p>

	<p>Collaborators must be able to review and approve.</p> <p>GitHub must validate merge conflicts.</p>
--	---

Requirements	Stages	Time estimation	Deliverables
<p>Define repository name and visibility.</p> <p>Repository should be created in GitHub and visible in the repo list.</p>	<ol style="list-style-type: none"> 1. Navigate to github.com/new 2. . 3. Enter repository details. 4. Click “Create repository.” 	20 min	New GitHub repository ready for use.
<p>Initialize local repository with git init.</p> <p>Connect local repo to GitHub remote.</p> <p>Stage, commit, and push files.</p>	<ol style="list-style-type: none"> 1. Open Git Bash in project folder. 2. Run git init. 3. Add remote with git remote add origin <URL>. 4. Stage files with git add .. 	40 minutes	Local project uploaded and visible in GitHub repository.

	<ol style="list-style-type: none"> Commit with git commit -m "message". Push with git push origin branch-name. 		
<p>Create a new branch without affecting the main codebase.</p> <p>Verify branch creation.</p>	<ol style="list-style-type: none"> Run git branch branch-name. Verify with git branch. 	15 minutes.	New branch successfully created.
<p>Modify requirements.txt with updated dependencies.</p> <p>Push changes to correct branch.</p>	<ol style="list-style-type: none"> Edit requirements.txt. Stage with git add requirements.txt. Commit with git commit -m "update requirements". Push changes with git push origin branch-name. 	30 minutes.	Updated requirements.txt visible in GitHub repository.
<p>Create PR after pushing branch changes.</p> <p>Compare changes between branches.</p> <p>Collaborators must review and approve.</p>	<ol style="list-style-type: none"> Push committed changes. In GitHub, click "Create pull request." Add description/comments . Submit PR. 	1 hour.	Pull request created, reviewed, and successfully merged.

Merge PR without conflicts.	<ol style="list-style-type: none">5. Collaborator reviews and approves.6. Click “Merge pull request.”		
-----------------------------	--	--	--