

Salesforce to AWS S3 File Uploader - GitHub Guide

1. Create the GitHub Repository

- Go to <https://github.com/new>
- Set Repository name: `salesforce-aws-s3-file-uploader`
- Description: `Secure file upload from Salesforce to AWS S3 using Lambda & pre-signed URLs`
- Set to Public
- Initialize with a README.md
- Click "Create repository"

2. Clone the Repository Locally

Run these commands in your terminal:

```
git clone https://github.com/<your-username>/salesforce-aws-s3-file-uploader.git
cd salesforce-aws-s3-file-uploader
```

3. Set Up the Project Structure

Create the following folders and structure:

- lambda/lambda_function.py
- salesforce/Apex/S3SignedUrlService.cls
- salesforce/LWC/uploadToS3/ (with .js, .html, .js-meta.xml)
- salesforce/RemoteSiteSetting.md
- aws-setup.md
- screenshots/demo-upload.png
- screenshots/aws-lambda-code.png

4. Add Lambda Code

Add `lambda_function.py` with the AWS Lambda code that generates the pre-signed URL using boto3.

Set environment variable: `BUCKET_NAME` = your S3 bucket name.

Salesforce to AWS S3 File Uploader - GitHub Guide

5. Add Apex Code to Salesforce Folder

Create the Apex class `S3SignedUrlService.cls` and include the method to make the HTTP callout to API Gateway to fetch the signed URL.

6. Add LWC Files

Add the 3 required LWC files to `uploadToS3` folder:

- `uploadToS3.js`
- `uploadToS3.html`
- `uploadToS3.js-meta.xml`

This includes UI for selecting a file and uploading to the signed S3 URL.

7. Add AWS Setup Instructions

Create `aws-setup.md` and document:

- S3 CORS settings
- IAM Policy for Lambda
- Lambda code and API Gateway setup

8. Add Screenshots (Optional)

Include `demo-upload.png` and `aws-lambda-code.png` in a `screenshots/` folder to enhance the repo visual presentation.

9. Commit and Push Everything

Run the following Git commands:

```
git add .  
git commit -m "Initial commit: Salesforce + AWS S3 file uploader"  
git push origin main
```