Project Codes:

1. Sample code for Text Extraction:

```
import boto3
import time

textract_client = boto3.client('textract')

bucket_name = 'ericsson-project-uddipan-1803'
pdf_file_key = 'MPTEL Payment info.pdf'

response = textract_client.start_document_text_detection(
    DocumentLocation={'S3Object': {'Bucket': bucket_name, 'Name':
pdf_file_key}}
)

job_id = response['JobId']

while True:
    result = textract_client.get_document_text_detection(JobId=job_id)
    status = result['JobStatus']

    if status == 'SUCCEEDED':
        break
    elif status == 'FAILED' or status == 'PARTIAL_SUCCESS':
        print("Text extraction job failed or partially succeeded.")
        break

    print("Job status:", status)
    time.sleep(5)

extracted_text = ''
for item in result['Blocks']:
    if item['BlockType'] == 'LINE':
        extracted_text += item['Text'] + '\n'

print(extracted_text)
```

Sample code to extract text from a pdf and print it:

```
import os
import boto3
import time
from flask import Flask, request, render_template

app = Flask(__name__)

s3_client = boto3.client('s3')
textract_client = boto3.client('textract')

bucket_name = 'ericsson-project-uddipan-1803'

@app.route('/')
def upload_form():
    return render_template('upload.html')

@app.route('/upload', methods=['POST'])
def upload_file():
    pdf_file = request.files['pdf_file']
    if pdf_file:
        pdf_file_key = f'uploads/{pdf_file.filename}'
        s3_client.upload_fileobj(pdf_file, bucket_name, pdf_file_key)

        response = textract_client.start_document_text_detection(
            DocumentLocation={'S3Object': {'Bucket': bucket_name, 'Name':
pdf_file_key}}
        )

        job_id = response['JobId']

        while True:
            result = textract_client.get_document_text_detection(JobId=job_id)
            status = result['JobStatus']

            if status == 'SUCCEEDED':
                break
            elif status == 'FAILED' or status == 'PARTIAL_SUCCESS':
                return "File upload and text extraction failed."

            time.sleep(5)

        extracted_text = ''
        for item in result['Blocks']:
            if item['BlockType'] == 'LINE':
                extracted_text += item['Text'] + '\n'
```

```
        return f"File uploaded and text
extracted:<br><pre>{extracted_text}</pre>"
    return "File upload failed."

if __name__ == '__main__':
    app.run(debug=True)
```

Code to extract text from aadhar card pdf and check for the presence of a 12 digit number:

```
import os
import boto3
import time
import re
from flask import Flask, request, render_template

app = Flask(__name__)

s3_client = boto3.client('s3')
textract_client = boto3.client('textract')

bucket_name = 'ericsson-project-uddipan-1803'

@app.route('/')
def upload_form():
    return render_template('upload.html')

@app.route('/upload', methods=['POST'])
def upload_file():
    pdf_file = request.files['pdf_file']
    if pdf_file:
        pdf_file_key = f'uploads/{pdf_file.filename}'
        s3_client.upload_fileobj(pdf_file, bucket_name, pdf_file_key)

        response = textract_client.start_document_text_detection(
            DocumentLocation={'S3Object': {'Bucket': bucket_name, 'Name':
pdf_file_key}}
        )

        job_id = response['JobId']

        while True:
            result = textract_client.get_document_text_detection(JobId=job_id)
            status = result['JobStatus']

            if status == 'SUCCEEDED':
                break
            elif status == 'FAILED' or status == 'PARTIAL_SUCCESS':
```

```
                return "File upload and text extraction failed."

            time.sleep(5)

        extracted_text = ''
        for item in result['Blocks']:
            if item['BlockType'] == 'LINE':
                extracted_text += item['Text'] + '\n'

        pattern = r'\d{4} \d{4} \d{4}'
        has_12_digit_number = re.search(pattern, extracted_text)

        if has_12_digit_number:
            return f"File uploaded and contains a 12-digit number in the
specified format:<br><pre>{extracted_text}</pre>"
        else:
            return ' ' '<script>alert("Incorrect PDF");
window.location.href="/"</script>' ' '

    return "File upload failed."
if __name__ == '__main__':
    app.run(debug=True)
```