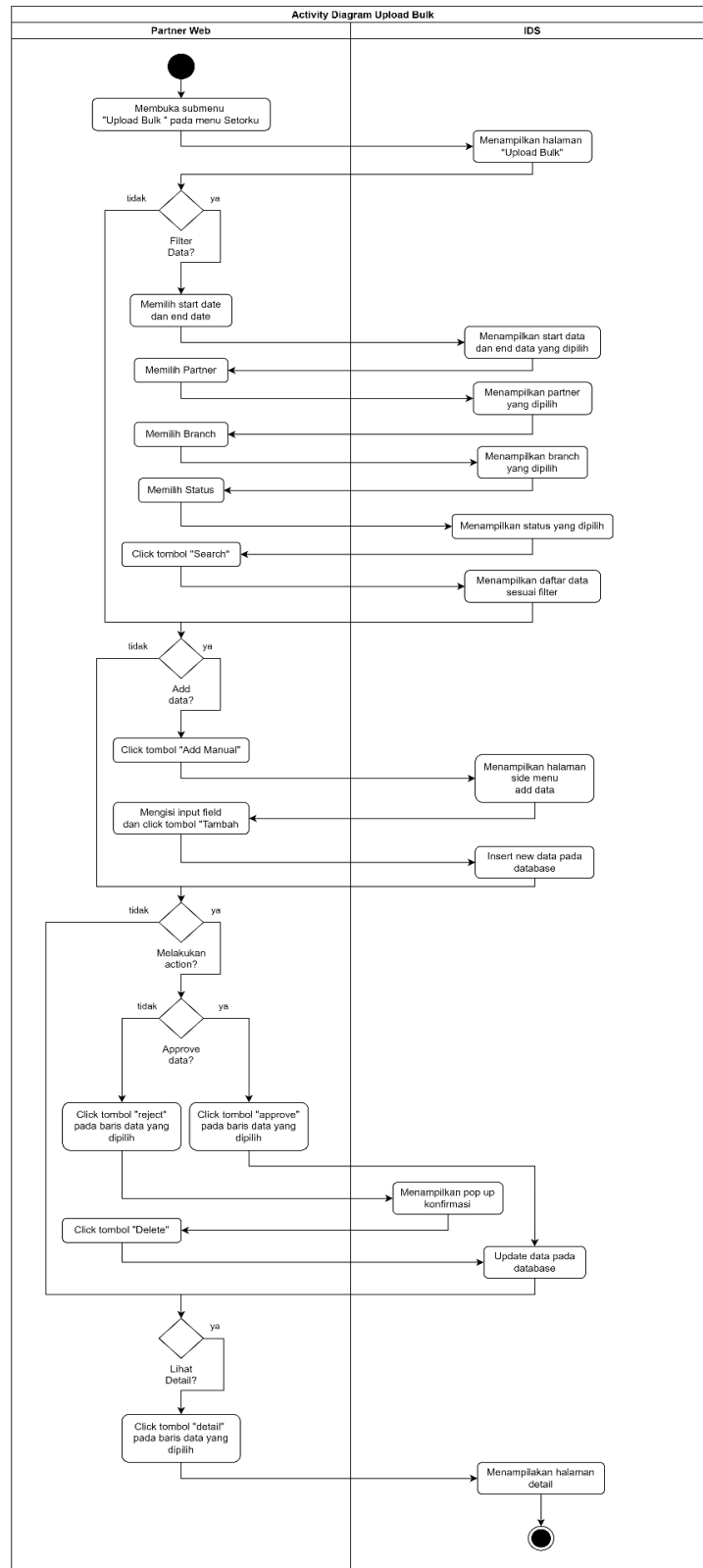
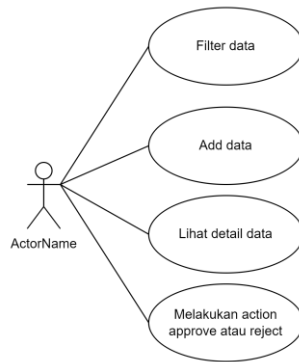


1. Masuklah ke web berikut dan buka menu Setorku -> Bulk Upload:
URL: <https://stg-sob.ids.id/auth/login>
Username : ryobbranch
Password : Test_1234
2. Pelajari Functional Design pada halaman 2 dan 3.
3. Dari Functional Design tersebut, tulis jawaban untuk poin-poin berikut:
 - Buka console web browser setelah membuka menu tersebut dan periksalah apakah ada pesan yang tercetak
 - Tuliskan endpoint API beserta header dan parameter yang dikirim saat melakukan klik di semua tombol page tersebut
 - Buatlah dokumen Test Case
 - Buatlah selenium script untuk menjalankan test case yang dibuat pada poin sebelumnya dan capture hasil/bug reportnya

Activity Diagram



Use Case Diagram



Business Rule

- Module ini menjelaskan bisnis proses user dapat melihat, memfilter, add data, dan melakukan approve atau reject.
- Data yang ditampilkan adalah:
 - o Date
 - o Name
 - o Username
 - o Partner
 - o Branch
 - o Amount
 - o Total Transaction
 - o Status

Acceptance Criteria

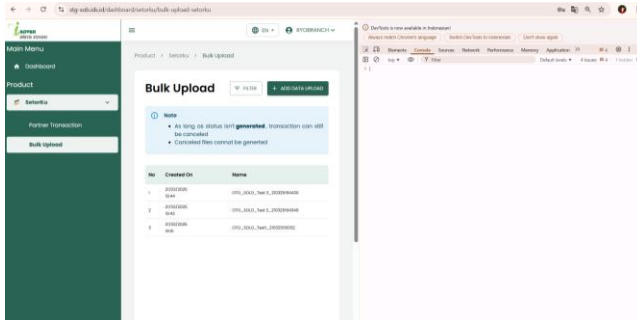
- User dapat melihat data yang telah ditambahkan pada tabel
- User dapat melakukan filter berdasar tanggal, kode partner, kode cabang atau status
- User dapat menambah data dengan melakukan pengisian data-data mandatory di menu tersebut
- Setelah melakukan approval, user akan mendapat kode bayar
- Kolom Name, alphanumeric, maksimal 255 karakter
- Kolom MerchantTransactionID, alphanumeric, maksimal 20 karakter
- Kolom UserCode, alphanumeric, maksimal 12 karakter
- Kolom UserName, alphanumeric, maksimal 55 karakter
- Kolom CustomerID, alphanumeric, maksimal 20 karakter
- Kolom Amount, decimal, maksimal 15 digit
- Kolom Periode, alphanumeric, maksimal 2 karakter

Jawaban untuk penyelesaian halaman 2 (aktifity diagram):

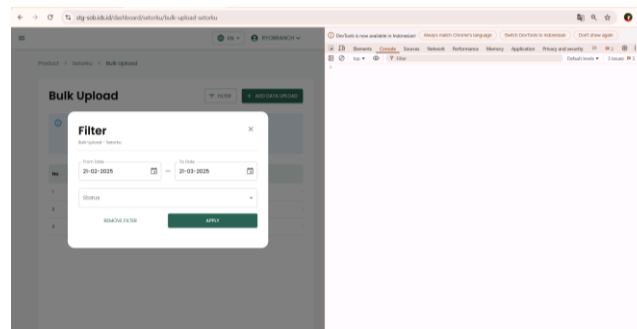
Dari Functional Design tersebut, tulis jawaban untuk poin-poin berikut:

- Buka console web browser setelah membuka menu tersebut dan periksalah apakah ada pesan yang tercetak

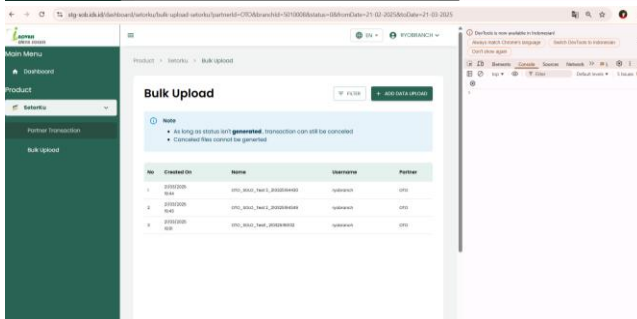
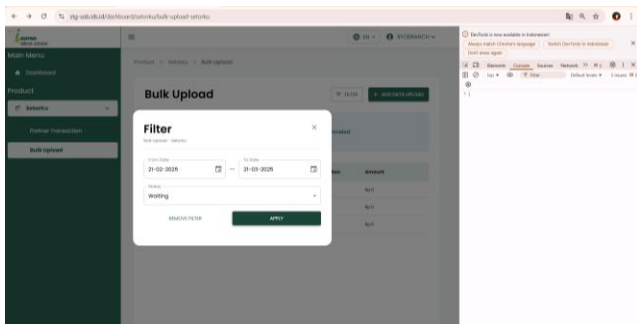
1. Jawaban untuk proses, Membuka submenu upload “bulk” pada menu Setorku: setelah di periksa tidak ditemukan ada pesan pada console.



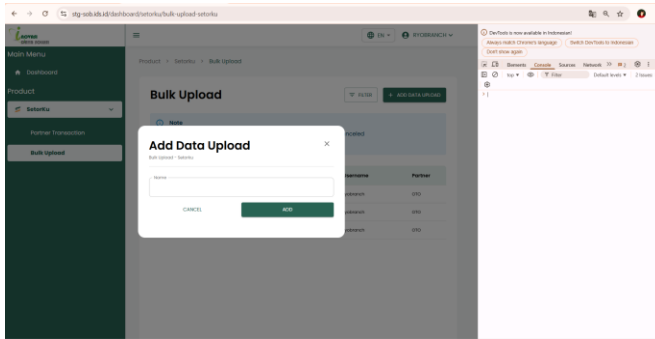
2. Jawaban untuk proses, klik filter data: setelah di periksa tidak ditemukan ada pesan pada console



3. Jawaban untuk proses, Search (proses ketika data filter sudah ter-input semua lalu klik apply.): Setelah diperiksa tidak ditemukan ada pesan pada console.



4. Jawaban untuk proses, klik add data:
Setelah di periksa tidak ditemukan ada pesan pada console.



5. Jawaban untuk proses, apply add data (proses ketika data add data upload sudah ter-input semua datanya lalu klik apply.)
Setelah di periksa ditemukan 2 pesan pada console yaitu:
- `_app-872947ae4c10538e.js:137`

POST `https://stg-sob.ids.id/api/main/setorku/bulk-upload` 404 (Not Found)

(anonymous) @ `_app-872947ae4c10538e.js:137`

xhr @ `_app-872947ae4c10538e.js:137`

eT @ `_app-872947ae4c10538e.js:137`

request @ `_app-872947ae4c10538e.js:137`

(anonymous) @ `_app-872947ae4c10538e.js:137`

(anonymous) @ `_app-872947ae4c10538e.js:137`

AddBulkUploadList @ `_app-872947ae4c10538e.js:114`

AddBulkUploadList @ `_app-872947ae4c10538e.js:114`

mutationFn @ `5496.2eaea37defb6077d.js:1`

fn @ `_app-872947ae4c10538e.js:121`

u @ `_app-872947ae4c10538e.js:121`

d @ `_app-872947ae4c10538e.js:121`

t.executeMutation @ `_app-872947ae4c10538e.js:121`

(anonymous) @ `_app-872947ae4c10538e.js:121`

Promise.then

t.execute @ `_app-872947ae4c10538e.js:121`

r.mutate @ `_app-872947ae4c10538e.js:121`

(anonymous) @ `_app-872947ae4c10538e.js:121`

(anonymous) @ `5496.2eaea37defb6077d.js:1`

(anonymous) @ `_app-872947ae4c10538e.js:137`

await in (anonymous)

eU @ `framework-aec4381329cec0e4.js:9`

eH @ `framework-aec4381329cec0e4.js:9`

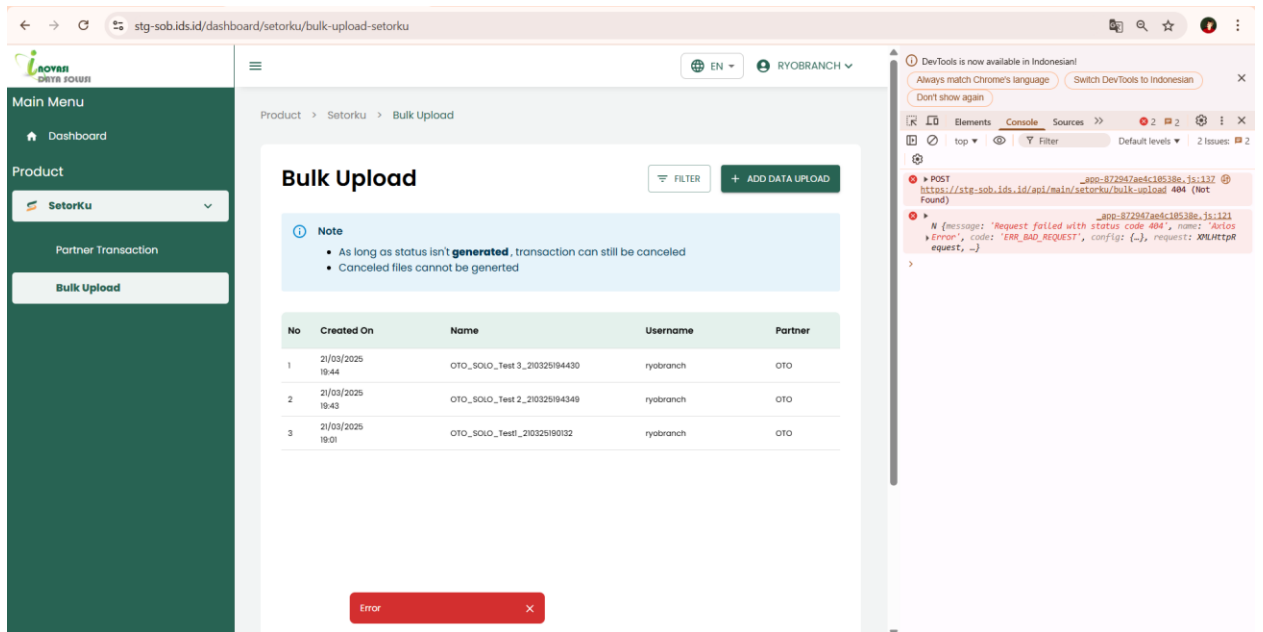
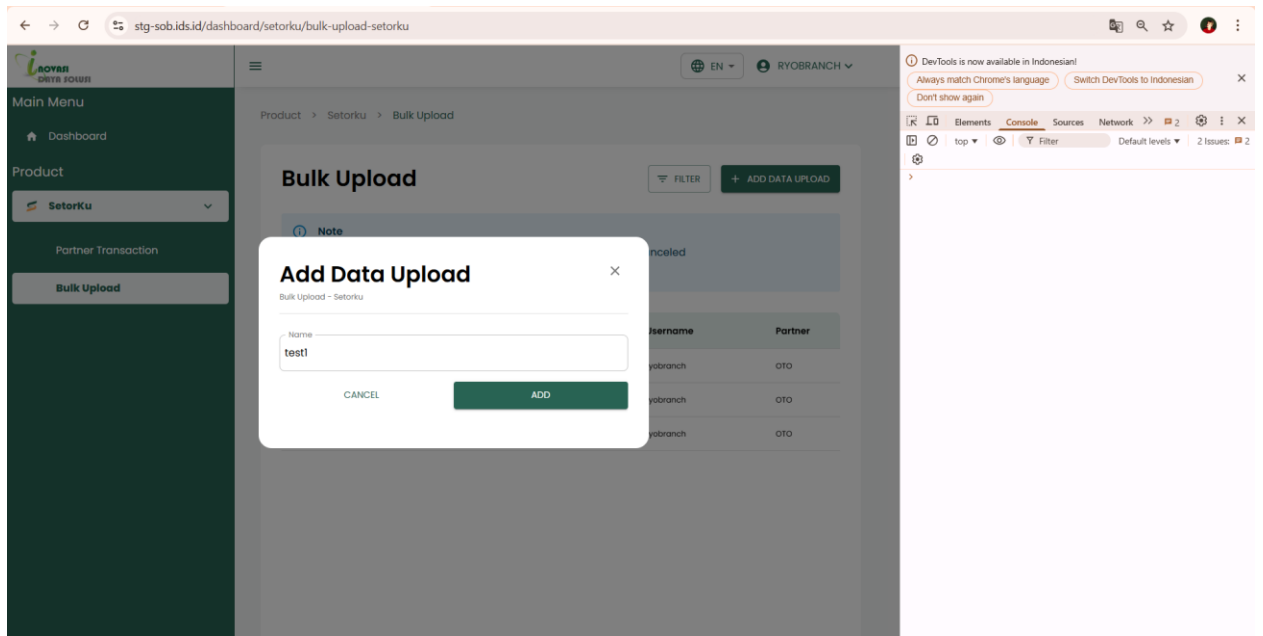
(anonymous) @ `framework-aec4381329cec0e4.js:9`

```

re @ framework-aec4381329cec0e4.js:9
rn @ framework-aec4381329cec0e4.js:9
(anonymous) @ framework-aec4381329cec0e4.js:9
oP @ framework-aec4381329cec0e4.js:9
eF @ framework-aec4381329cec0e4.js:9
ro @ framework-aec4381329cec0e4.js:9
nU @ framework-aec4381329cec0e4.js:9
nD @ framework-aec4381329cec0e4.js:9
_app-872947ae4c10538e.js:121 N {message: 'Request failed with status code 404', name:
'AxiosError', code: 'ERR_BAD_REQUEST', config: {...}, request: XMLHttpRequest, ...}code:
"ERR_BAD_REQUEST"config: {transitional: {...}, adapter: Array(2), transformRequest:
Array(1), transformResponse: Array(1), timeout: 0, ...}message: "Request failed with status
code 404"name: "AxiosError"request: XMLHttpRequest {onreadystatechange: null,
readyState: 4, timeout: 0, withCredentials: false, upload:
XMLHttpRequestUpload, ...}response: {data: {...}, status: 404, statusText: "", headers: eg,
config: {...}, ...}stack: "AxiosError: Request failed with status code 404\n  at https://stg-
sob.ids.id/_next/static/chunks/pages/_app-872947ae4c10538e.js:137:36355\n  at
XMLHttpRequest.d (https://stg-sob.ids.id/_next/static/chunks/pages/_app-
872947ae4c10538e.js:137:36495)"[[Prototype]]: Error
(anonymous) @ _app-872947ae4c10538e.js:121
Promise.catch
t.execute @ _app-872947ae4c10538e.js:121
r.mutate @ _app-872947ae4c10538e.js:121
(anonymous) @ _app-872947ae4c10538e.js:121
(anonymous) @ 5496.2eaea37defb6077d.js:1
(anonymous) @ _app-872947ae4c10538e.js:137
await in (anonymous)
eU @ framework-aec4381329cec0e4.js:9
eH @ framework-aec4381329cec0e4.js:9
(anonymous) @ framework-aec4381329cec0e4.js:9
re @ framework-aec4381329cec0e4.js:9
rn @ framework-aec4381329cec0e4.js:9
(anonymous) @ framework-aec4381329cec0e4.js:9
oP @ framework-aec4381329cec0e4.js:9
eF @ framework-aec4381329cec0e4.js:9
ro @ framework-aec4381329cec0e4.js:9
nU @ framework-aec4381329cec0e4.js:9
nD @ framework-aec4381329cec0e4.js:9
• {
  "message": "Request failed with status code 404",
  "name": "AxiosError",

```

```
"stack": "AxiosError: Request failed with status code 404\n  at https://stg-sob.ids.id/_next/static/chunks/pages/_app-872947ae4c10538e.js:137:36355\n  at XMLHttpRequest.d (https://stg-sob.ids.id/_next/static/chunks/pages/_app-872947ae4c10538e.js:137:36495)",
"config": {
  "transitional": {
    "silentJSONParsing": true,
    "forcedJSONParsing": true,
    "clarifyTimeoutError": false
  },
  "adapter": [
    "xhr",
    "http"
  ],
  "transformRequest": [
    null
  ],
  "transformResponse": [
    null
  ],
  "timeout": 0,
  "xsrfCookieName": "XSRF-TOKEN",
  "xsrfHeaderName": "X-XSRF-TOKEN",
  "maxContentLength": -1,
  "maxBodyLength": -1,
  "env": {},
  "headers": {
    "Accept": "application/json, text/plain, */*",
    "Content-Type": "application/json"
  },
  "baseUrl": "/api",
  "method": "post",
  "url": "/main/setorku/bulk-upload",
  "data": "{\"partnerId\":\"OTO\",\"branchId\":\"5010008\",\"filename\":\"test1\"}"
},
"code": "ERR_BAD_REQUEST",
"status": 404
}
```



6. Jawaban untuk proses, klik detail pada salah satu data yang di tampilkan bulk upload:
Setelah di periksa tidak ditemukan ada pesan pada console.

The screenshot displays the iOYAN DINTAS FOKUS application interface. The top navigation bar includes the logo, a hamburger menu, and the user profile 'RYOBRANCH'. The main menu on the left lists 'Dashboard', 'Setorku', 'Partner Transaction', and 'Bulk Upload'. The 'Bulk Upload' section is active, showing a 'Bulk Upload' screen with a table of data. The table has columns: Partner, Branch, Total Transaction, and Amount. The data is as follows:

Partner	Branch	Total Transaction	Amount
OTO	SOLO	0	Rp 0
OTO	SOLO	0	Rp 0
OTO	SOLO	0	Rp 0

Below the table, there is a 'Bulk Upload Detail' screen. It shows the 'File Name' as 'OTO_SOLO_Test_3_210325194430'. Below this, there is a table with 5 columns: No, Merchant Transaction ID, Customer ID, Payment Code, and User Code. The table is currently empty, with a message 'Filter data first to show the data.' and a search icon. The bottom of the screen shows 'Rows per page: 5' and '0-0 of 0'.

The browser address bar shows the URL: stg-sob.ids.id/dashboard/setorku/bulk-upload-setorku/17011. The DevTools console is open, showing no messages.

- Tuliskan endpoint API beserta header dan parameter yang dikirim saat melakukan klik di semua tombol page tersebut

1. Jawaban untuk proses, Membuka submenu upload “bluk” pada menu Setorku:

Endpoint API	
GET	https://stg-sob.ids.id/api/main/setorku/bulk-upload?fromDate=23-02-2025&toDate=23-03-2025&pageNumber=1&pageSize=10&branchId=5010008&partnerId=OTO

Header	
Content-Type	application/json; charset=utf-8
User-Agent	Mozilla/5.0
Accept	application/json, text/plain, */*

Query String Parameter	
fromDate	23-02-2025
toDate	23-02-2025
pageNumber	1
pageSize	10
branchId	5010008
partnerID	OTO

2. Jawaban untuk proses, Search (proses ketika data filter sudah ter-input semua data lalu klik apply.):

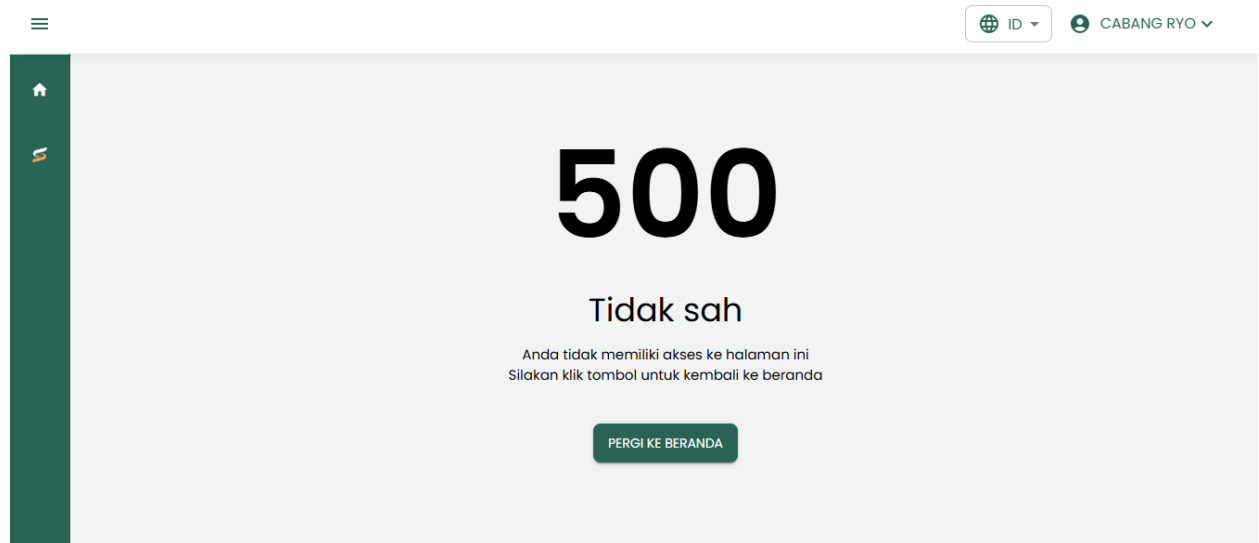
Endpoint API	
GET	https://stg-sob.ids.id/api/main/setorku/bulk-upload?fromDate=21-03-2025&toDate=23-03-2025&pageNumber=1&pageSize=10&branchId=5010008&partnerId=OTO&status=0

Header	
Content-Type	application/json; charset=utf-8
User-Agent	Mozilla/5.0
Accept	application/json, text/plain, */*

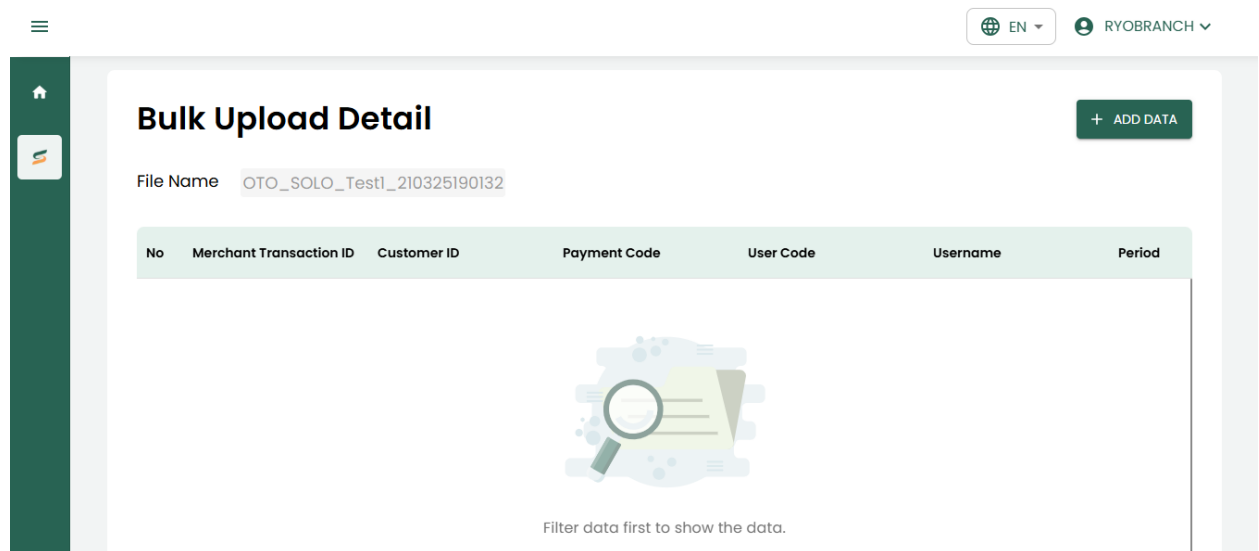
Query String Parameter	
fromDate	21-03-2025
toDate	23-03-2025
pageNumber	1
pageSize	10
branchId	5010008
partnerID	OTO
Status	0

3. Jawaban untuk proses, apply add data (proses ketika data add data upload sudah ter-input semua datanya lalu klik apply.)

NOTE: Error saat menambahkan data baru



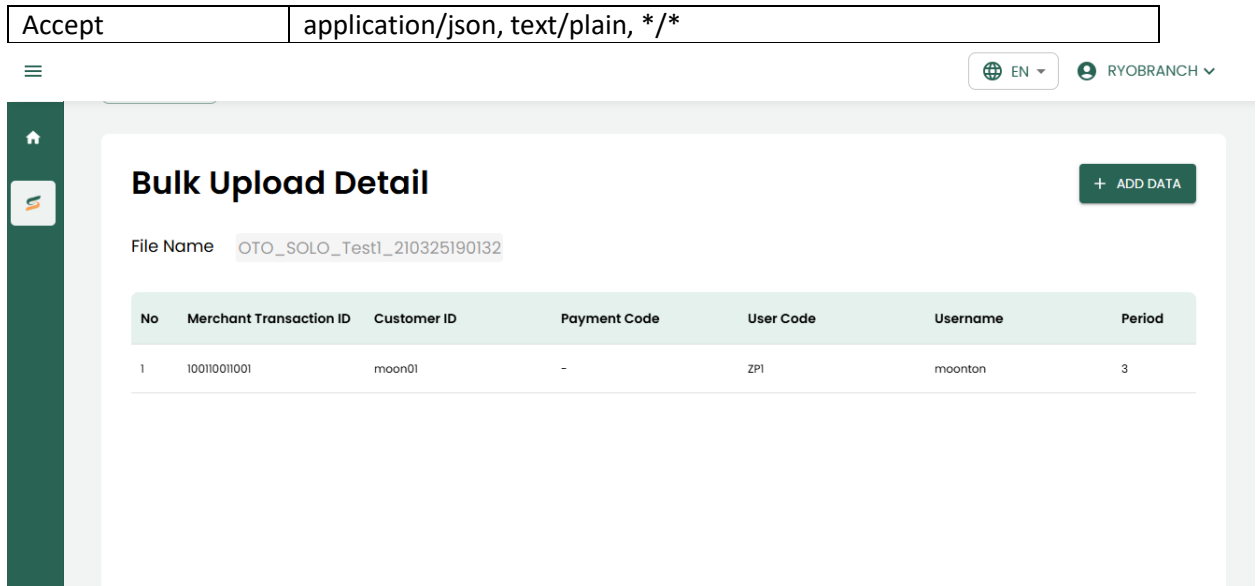
4. Jawaban untuk proses, klik detail pada salah satu data yang di tampilkan bulk upload: Setelah di periksa tidak ditemukan ada pesan pada console. Error tidak menampilkan detail dari data, apakah harus ditambahkan secara manual?



Jika iya, berikut preview setelah menambahkan data secara manual

Endpoint API	
POST	https://stg-sob.ids.id/api/main/setorku/bulk-upload/17009

Header	
Content-Type	application/json; charset=utf-8
User-Agent	Mozilla/5.0



- Buatlah dokumen Test Case
jawaban untuk dokumen test case saya buat di file berbeda agar lebih rapih tabelnya yaitu pada file Test Case IDS.xlsx digithub <https://github.com/syarifahhani/Testing-IDS/blob/main/Test%20Case%20IDS.xlsx>

- Buatlah selenium script untuk menjalankan test case yang dibuat pada poin sebelumnya dan capture hasil/bug reportnya

Jawaban:

Kode Untuk Menguji halaman Login.

sebelum jalan kan kodingan ini jangan lupa untuk: pip install selenium webdriver-manager

#kodingan ini akan mengecek halaman login

import time

import logging

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from webdriver_manager.chrome import ChromeDriverManager

from selenium.webdriver.chrome.options import Options

from selenium.webdriver.common.by import By

from selenium.common.exceptions import NoSuchElementException

Konfigurasi Logging

logging.basicConfig(

 filename="login_test.log",

 level=logging.INFO,

 format="%(asctime)s - %(levelname)s - %(message)s"

)

```

# Setup WebDriver
options = Options()
options.add_argument("--start-maximized") # Membuka browser dalam mode full screen
service = Service(ChromeDriverManager().install())
driver = webdriver.Chrome(service=service, options=options)

try:
    # 1. Buka Halaman Login
    url = "https://stg-sob.ids.id/auth/login"
    driver.get(url)
    logging.info("Berhasil membuka halaman login: " + url)

    # 2. Input username
    try:
        username = driver.find_element(By.NAME, "username") # Sesuaikan dengan atribut elemen
        username.send_keys("ryobbranch")
        logging.info("Berhasil memasukkan username")
    except NoSuchElementException:
        logging.error("Kolom username tidak ditemukan!")
        driver.save_screenshot("bug_username_not_found.png")
        raise

    # 3. Input password
    try:
        password = driver.find_element(By.NAME, "password") # Sesuaikan dengan atribut elemen
        password.send_keys("Test_1234")
        logging.info("Berhasil memasukkan password")
    except NoSuchElementException:
        logging.error("Kolom password tidak ditemukan!")
        driver.save_screenshot("bug_password_not_found.png")
        raise

    # 4. Klik tombol login
    try:
        login_button = driver.find_element(By.XPATH, "//button[@type='submit']") # Sesuaikan
        dengan XPath tombol login
        login_button.click()
        logging.info("Berhasil mengklik tombol Login")
    except NoSuchElementException:
        logging.error("Tombol login tidak ditemukan!")
        driver.save_screenshot("bug_login_button_not_found.png")

```

raise

5. Verifikasi hasil login

time.sleep(3) # Tunggu 3 detik agar halaman load sepenuhnya

try:

 dashboard_element = driver.find_element(By.XPATH, "//h1[contains(text(),'Dashboard')]") #

Sesuaikan elemen yang muncul setelah login

 logging.info("Login berhasil! Dashboard ditemukan.")

except NoSuchElementException:

 logging.error("Login gagal! Dashboard tidak ditemukan.")

 driver.save_screenshot("bug_dashboard_not_found.png")

raise

except Exception as e:

 logging.error("Error tidak terduga: " + str(e))

 driver.save_screenshot("bug_unexpected_error.png")

finally:

 time.sleep(3) # Tunggu sebelum browser ditutup

 driver.quit()

 logging.info("Test selesai, browser ditutup.")

print("Test login selesai! Cek file 'login_test.log' dan screenshot jika ada bug.")

Kode Untuk Menguji Tombol:

1. **POST** <https://stg-sob.ids.id/api/main/setorku/bulk-upload/17009>,
2. **GET** <https://stg-sob.ids.id/api/main/setorku/bulk-upload?fromDate=21-03-2025&toDate=23-03-2025&pageNumber=1&pageSize=10&branchId=5010008&partnerId=OTO&status=0>,
3. **GET** <https://stg-sob.ids.id/api/main/setorku/bulk-upload?fromDate=23-02-2025&toDate=23-03-2025&pageNumber=1&pageSize=10&branchId=5010008&partnerId=OTO>

import time

import logging

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from webdriver_manager.chrome import ChromeDriverManager

from selenium.webdriver.chrome.options import Options

from selenium.webdriver.common.by import By

from selenium.common.exceptions import NoSuchElementException, TimeoutException

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected_conditions as EC

Konfigurasi Logging

```

logging.basicConfig(
    filename="button_test.log",
    level=logging.INFO,
    format="%(asctime)s - %(levelname)s - %(message)s"
)

# Setup WebDriver
options = Options()
options.add_argument("--start-maximized") # Membuka browser dalam mode full screen
service = Service(ChromeDriverManager().install())
driver = webdriver.Chrome(service=service, options=options)

try:
    # 1. Login ke website
    url = "https://stg-sob.ids.id/auth/login"
    driver.get(url)
    logging.info("Berhasil membuka halaman login: " + url)

    # Masukkan username
    WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.NAME,
"username"))).send_keys("ryobbranch")
    logging.info("Berhasil memasukkan username")

    # Masukkan password
    WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.NAME,
"password"))).send_keys("Test_1234")
    logging.info("Berhasil memasukkan password")

    # Klik tombol login
    WebDriverWait(driver, 10).until(EC.element_to_be_clickable((By.XPATH,
"//button[@type='submit']"))).click()
    logging.info("Berhasil mengklik tombol Login")

    # Tunggu sampai halaman utama/dasbor terbuka
    time.sleep(3)

    # 2. Membuka submenu "Upload Bulk" pada menu Setorku
    try:
        bulk_upload_button = WebDriverWait(driver, 10).until(
            EC.element_to_be_clickable((By.XPATH, "//button[contains(text(),'Upload Bulk')])) #
Sesuaikan dengan teks tombol
        )

```

```

bulk_upload_button.click()
logging.info("Berhasil membuka submenu Upload Bulk")

# Cek apakah halaman berhasil dimuat dengan memeriksa elemen tertentu
time.sleep(3)
WebDriverWait(driver, 10).until(
    EC.presence_of_element_located((By.XPATH, "//h1[contains(text(),'Bulk Upload')]")) #
    Sesuaikan dengan teks di halaman
)
logging.info("Halaman Bulk Upload terbuka dengan benar")
except TimeoutException:
    logging.error("Gagal membuka submenu Upload Bulk!")
    driver.save_screenshot("bug_upload_bulk.png")

# 3. Melakukan pencarian dengan filter dan klik apply
try:
    from_date = driver.find_element(By.NAME, "fromDate") # Sesuaikan dengan atribut elemen
    from_date.clear()
    from_date.send_keys("21-03-2025")

    to_date = driver.find_element(By.NAME, "toDate")
    to_date.clear()
    to_date.send_keys("23-03-2025")

    # Klik tombol Apply
    apply_button = WebDriverWait(driver, 10).until(
        EC.element_to_be_clickable((By.XPATH, "//button[contains(text(),'Apply')]"))
    )
    apply_button.click()
    logging.info("Berhasil mengklik tombol Apply untuk melakukan pencarian")

    # Cek apakah hasil muncul
    time.sleep(3)
    result_data = driver.find_elements(By.XPATH, "//table/tbody/tr")
    if len(result_data) > 0:
        logging.info(f"Ditemukan {len(result_data)} hasil pencarian")
    else:
        logging.warning("Tidak ada data yang muncul setelah pencarian")
        driver.save_screenshot("bug_no_search_result.png")
except TimeoutException:
    logging.error("Gagal melakukan pencarian dengan filter!")
    driver.save_screenshot("bug_search_filter.png")

```



```

# 4. Klik detail pada salah satu data bulk upload
try:
    detail_button = WebDriverWait(driver, 10).until(
        EC.element_to_be_clickable((By.XPATH,
            "//table/tbody/tr[1]/td/button[contains(text(),'Detail')]"))
    )
    detail_button.click()
    logging.info("Berhasil mengklik tombol Detail pada data pertama")

    # Cek apakah halaman detail berhasil terbuka
    time.sleep(3)
    WebDriverWait(driver, 10).until(
        EC.presence_of_element_located((By.XPATH, "//h1[contains(text(),'Detail Bulk Upload')]")) #
    )
    logging.info("Halaman Detail Bulk Upload terbuka dengan benar")
except TimeoutException:
    logging.error("Gagal membuka halaman Detail Bulk Upload!")
    driver.save_screenshot("bug_detail_not_found.png")

except Exception as e:
    logging.error("Error tidak terduga: " + str(e))
    driver.save_screenshot("bug_unexpected_error.png")

finally:
    time.sleep(3) # Tunggu sebelum browser ditutup
    driver.quit()
    logging.info("Test selesai, browser ditutup.")

print("Test selesai! Cek file 'button_test.log' dan screenshot jika ada bug.")

```