

```

import pytest
from pathlib import Path
import json
import requests
import random
import datetime
import time

from datetime import datetime, timedelta
from Utilities.utils import user_id_url, token_generation, device_id

BASE_DIR = Path(__file__).resolve().parent
CONFIG_PATH = BASE_DIR / "endpoints.json"

def get_base_endpoint():
    try:
        with CONFIG_PATH.open() as json_file:
            properties = json.load(json_file)
            env = properties["mock_environment"]["mock_env"]
            return properties[env]["base_url"]
    except (FileNotFoundError, KeyError, json.JSONDecodeError) as e:
        raise RuntimeError(f"Failed to load the base endpoint: {e}")

@pytest.fixture(scope="session", autouse=True)
def mock_base_url():
    return get_base_endpoint()

def get_crm_base_endpoint():
    try:
        with CONFIG_PATH.open() as json_file:
            properties = json.load(json_file)
            env = properties["crm_stage_endpoint"]["crm_base_url"]
            return env
    except (FileNotFoundError, KeyError, json.JSONDecodeError) as e:
        raise RuntimeError(f"Failed to load the base endpoint: {e}")

@pytest.fixture(scope="session", autouse=True)
def call_crm_base_url():
    return get_crm_base_endpoint()

def get_userid():
    fetch_token = token_generation()
    send_url = user_id_url()
    url = str(send_url)
    userid_header = {
        "Authorization": fetch_token
    }

```

```

send_request = requests.get(url, headers=userid_header)
response = send_request.json()
capture_id = response["userId"]
return capture_id

@pytest.fixture(scope="session", autouse=True)
def fetch_userid():
    return get_userid()

@pytest.fixture(scope="function")
def custom_header_crm():
    fetch_token = token_generation()
    return {
        "Authorization": fetch_token,
        "User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36",
        "accept": "application/json",
        "Content-Type": "application/json",
        "host": "zencast.nixy.stg-drove.phonepe.nb6"
    }

@pytest.fixture(scope="function")
def crm_inbox_payload():
    send_userid = str(get_userid())
    user_id = send_userid
    fetch_device_id = device_id()
    random_integer = random.randint(120, 9000)
    random_message_id = random.randint(105, 945)
    random_group_id = random.randint(143, 865)
    req_id = f"reqShru{random_integer}"
    mess_id = f"msgShru{random_message_id}"
    group_id = f"msgShru{random_group_id}"
    current_epoch_milliseconds = int((datetime.now() +
timedelta(minutes=5)).timestamp() * 1000)
    one_day_ago = datetime.now() - timedelta(days=1)
    one_day_ago_epoch = int(time.mktime(one_day_ago.timetuple()) * 1000)

    return {
        "requestId": req_id,
        "profileId": "CONSUMER_TRANSACTIONAL",
        "enableRetry": "false",
        "entityId": user_id,
        "deviceId": fetch_device_id,
        "customTags": {
            "tag1": "karthik-test-campa631",
            "tag2": "1",
            "tag3": user_id,
            "tag4":
"HE@CM22041319515587550137252@1@d675e16c-8285-455a-8649-1046d3098c50",

```

```

        "tag5": ""},
    "appMessage": {
        "messageId": mess_id,
        "groupingKey": group_id,
        "source": {
            "type": "PHONEPE"
        },
        "data": {
            "placements": [
                {
                    "scope": "INBOX",
                    "expiresAt": current_epoch_milliseconds,
                    "template": {
                        "templateId": "ICON_TITLE_SUBTITLE",
                        "templateParams": {
                            "value": {
                                "title": "myinboxstagetest",
                                "subTitle": "Click here",
                                "iconURL": "https:October was a memorable
month for us. We hit the 250 million user mark in India."
                            }
                        },
                        "nav": {
                            "key": "action_redirection",
                            "params": {
                                "deepLinkIOS":
"phonepestage://dg?providerId=SAFEGOLD"
                            }
                        }
                    },
                    "deferment": {
                        "type": "NONE",
                        "minutes": 1,
                        "future": current_epoch_milliseconds,
                        "retryWindowInSec": 0
                    }
                },
                {
                    "scope": "DRAWER",
                    "expiresAt": current_epoch_milliseconds,
                    "template": {
                        "templateId": "TITLE_MESSAGE_IMAGE",
                        "templateParams": {
                            "value": {
                                "title": "Shrudrawernotif",
                                "message": "Click here",
                                "image":
"https://hexcode.in/wp-content/uploads/phonepe-refer-earn-offer.jpg"
                            }
                        }
                    }
                }
            ]
        }
    }
}

```

```

        },
        "nav": {
            "key": "action_redirection",
            "params": {
                "deepLinkIOS":
"phonepestage://dg?providerId=SAFEGOLD"
            }
        },
        "deferment": {
            "type": "NONE",
            "minutes": 1,
            "future": current_epoch_milliseconds,
            "retryWindowInSec": 0
        }
    },
    ],
    "campaignId": "karthik-test-campa631",
    "communicationIntent": "TRANSACTIONAL",
    "syncSections": [
        {
            "type": "all",
            "params": {
                "testKey1": "testValue1",
                "testKey2": "testValue2"
            }
        }
    ],
    "utm_source": "Push",
    "utm_medium": "Hawkeye",
    "utm_campaign":
"HAWKEYE@OWRkYmRmMmNiYzdZThiMWZmYjNkNmNjMDg3Y2Q1MTA3OTNlNjc3OTM3MzdjMGQ2MGJlYj
M3MDZmMTUxMGQ4MThlOGQ3NzJjM2EwZjRhYTI0NTk5NGQ1ZGQ0YzcxM2VjNTkwN2IzMjRmOWIzYjM0Y
jF1OGVlMjMxMDliYTNmNGE3Y2RlOWI4NTgzYWU2Zjc5MDU5ZjZkNDY2NThiYTNjOGY3MmUyNTE1Zjk4
YmJmNWU6ZTNjMzZlMzQ5YWFkYzZhOWUwMTA2ZTkxODJhNTE0NjY"
    },
    "properties": {
        "editable": True,
        "deletable": True,
        "override": True,
        "unsubscribeSupported": False
    },
    "sentAt": one_day_ago_epoch,
    "expiresAt": current_epoch_milliseconds
}
}

@pytest.fixture(scope="function")
def crm_cta_payload():

```

```

send_userid = str(get_userid())
user_id = send_userid
random_integer = random.randint(10, 900)
random_group_integer = random.randint(25, 980)
fetch_device_id = device_id()
request_id = f"Requeestb{random_integer}cbcb"
message_id = f"Requeestb22cbcb{random_integer}"
grouping_id = f"Requeestb22cbcb{random_group_integer}"
current_epoch_milliseconds = int((datetime.now() +
timedelta(minutes=5)).timestamp() * 1000)
one_day_ago = datetime.now() - timedelta(days=1)
one_day_ago_epoch = int(time.mktime(one_day_ago.timetuple()) * 1000)

    return {
"requestId": request_id,
"userId": user_id,
"profileId": "CONSUMER_PROMOTIONAL",
"deviceId": fetch_device_id,
"appMessage": {
    "version": 2,
    "messageId": message_id,
    "groupingKey": grouping_id,
    "source": {
        "type": "PHONEPE"
    },
},
"data": {
    "placements": [
        {
            "scope": "DRAWER",
            "template": {
                "templateId": "TITLE_MESSAGE_IMAGE",
                "templateParams": {
                    "value": {
                        "image":
"https://imgstatic.phonepe.com/images/notifications/900/480/notification-15-04-22-health999.png",
                        "title": "Test",
                        "message": "Starting at just ₹999*. Secure
yourself against rising medical costs right away!",
                        "headerSubText": "Test",
                        "largeIconUrl":
"https://imgstatic.phonepe.com/images/notifications/900/480/notification-15-04-22-health999.png",
                        "fontColor": "red",
                        "bgColor": "#F00000"
                    }
                },
            },
        },
    ],
    "nav": {
        "key": "action_redirection",

```

```
        "params": {
            "deepLinkIOS":
"phonepe://sachetInsurance?category=HEALTH_INSURANCE&product=SIMPLE_HEALTH",
            "action_nav": "sachetInsuranceHome",
            "redirection_data": {
                "data": [
                    {
                        "key": "category",
                        "value": "HEALTH_INSURANCE",
                        "isEncoded": False
                    },
                    {
                        "key": "productType",
                        "value": "SIMPLE_HEALTH",
                        "isEncoded": False
                    }
                ]
            }
        },
        "callToAction": [
            {
                "text": "test",
                "action_nav": "sachetInsuranceHome",
                "redirection_data": {
                    "data": [
                        {
                            "key": "category",
                            "value": "HEALTH_INSURANCE",
                            "isEncoded": False
                        },
                        {
                            "key": "productType",
                            "value": "SIMPLE_HEALTH",
                            "isEncoded": False
                        }
                    ]
                }
            }
        ]
    },
    "fallbackStrategy": "NONE",
    "deferment": {
        "type": "NONE",
        "future": current_epoch_milliseconds,
        "retryWindowInSec": 0
    },
    "nav": {
        "key": "action_redirection",
```

```
        "params": {
            "deepLinkIOS":
"phonepe://sachetInsurance?category=HEALTH_INSURANCE&product=SIMPLE_HEALTH",
            "action_nav": "sachetInsuranceHome",
            "redirection_data": {
                "data": [
                    {
                        "key": "category",
                        "value": "HEALTH_INSURANCE",
                        "isEncoded": False
                    },
                    {
                        "key": "productType",
                        "value": "SIMPLE_HEALTH",
                        "isEncoded": False
                    }
                ]
            }
        },
        "callToAction": [
            {
                "text": "PAY",
                "action_nav": "sachetInsuranceHome",
                "redirection_data": {
                    "data": [
                        {
                            "key": "category",
                            "value": "HEALTH_INSURANCE",
                            "isEncoded": False
                        },
                        {
                            "key": "productType",
                            "value": "SIMPLE_HEALTH",
                            "isEncoded": False
                        }
                    ]
                }
            }
        ]
    },
    "campaignId": "testCampaignBETAAutomation",
    "syncSections": [],
    "communicationIntent": "PROMOTIONAL",
    "utm_source": "marketing",
    "utm_medium": "push",
    "utm_campaign": "test123"
},
```

```

        "properties": {
            "editable": True,
            "deletable": True,
            "override": True,
            "unsubscribeSupported": False
        },
        "customParams": {
            "campaignId": "C121212",
            "interventionId": "12",
            "userId": user_id
        },
        "sentAt": one_day_ago_epoch,
        "expiresAt": current_epoch_milliseconds
    }
}

```

```
@pytest.fixture(scope="function")
```

```

def crm_non_push():
    send_userid = str(get_userid())
    user_id = send_userid
    random_integer = random.randint(15, 90)
    random_message = random.randint(10, 80)
    random_grouping = random.randint(18, 98)
    fetch_device_id = device_id()
    random_request_id = f"test-apr-14-1st{random_integer}"
    random_message_id = f"test-apr-14-1st{random_message}"
    random_grouping_id = f"test-apr-14-1st{random_grouping}"
    one_day_ago = datetime.now() - timedelta(days=1)
    one_day_ago_epoch = int(time.mktime(one_day_ago.timetuple()) * 1000)
    current_epoch_milliseconds = int((datetime.now() +
timedelta(minutes=5)).timestamp() * 1000)

```

```

    return {
        "requestId": random_request_id,
        "userId": user_id,
        "deviceId": fetch_device_id,
        "profileId": "CONSUMER_PROMOTIONAL",
        "appMessage": {
            "version": 2,
            "messageId": random_message_id,
            "groupingKey": random_grouping_id,
            "source": {
                "type": "PHONEPE"
            },
        },
        "data": {
            "syncSections": [
                {
                    "type": "all"
                }
            ]
        }
    }

```



```

        ],
        "placements": [
            {
                "scope": "DRAWER",
                "template": {
                    "templateId": "TITLE_MESSAGE_IMAGE",
                    "templateParams": {
                        "value": {
                            "title": "Teja-auto-14-1st",
                            "message": "Teja-auto-14-1st"
                        }
                    }
                },
                "nav": {
                    "key": "action_redirection",
                    "params": {
                        "action_nav": "prepaidMobile",
                        "redirection_data": {
                            "type": "STATIC",
                            "content": "{\\"data\\":[]}"
                        }
                    }
                }
            },
            {
                "deferment": {
                    "type": "NONE",
                    "minutes": 1
                }
            }
        ],
        "campaignId": "testcampaign1",
        "communicationIntent": "PROMOTIONAL",
        "utm_source": "Push",
        "utm_medium": "Hawkeye",
        "utm_campaign": "testcampaign"
    },
    "properties": {
        "editable": True,
        "deletable": True,
        "override": True,
        "unsubscriptionSupported": False
    },
    "sentAt": one_day_ago_epoch,
    "expiresAt": current_epoch_milliseconds
}

@pytest.fixture(scope="function")
def crm_PN_and_inbox_Android():
    send_userid = str(get_userid())

```

```

user_id = send_userid
random_integer = random.randint(11,98)
random_message_id = random.randint(14,88)
random_grouping_id = random.randint(17,93)
fetch_device_id = device_id()
request_id = f"RSF-Ssatyam-G{random_integer}G4"
message_id = f"M1G-5483GP-{random_message_id}TG"
grouping_key = f"G2G-5483GP-{random_grouping_id}TG"
current_epoch_milliseconds = int((datetime.now() +
timedelta(minutes=5)).timestamp() * 1000)
one_day_ago = datetime.now() - timedelta(days=1)
one_day_ago_epoch = int(time.mktime(one_day_ago.timetuple()) * 1000)
return {
    "requestId": request_id,
    "userId": user_id,
    "deviceId": fetch_device_id,
    "profileId": "CONSUMER_TRANSACTIONAL",
    "appMessage": {
        "messageId": message_id,
        "groupingKey": grouping_key,
        "authorisedFor": user_id,
        "source": {
            "type": "PHONEPE"
        },
        "data": {
            "placements": [
                {
                    "scope": "DRAWER",
                    "template": {
                        "templateId": "TITLE_MESSAGE_IMAGE",
                        "templateParams": {
                            "value": {
                                "image":
"https://imgstatic.phonepe.com/images/notifications/900/480/notification-15-04-22-health999.png",
                                "title": "Hi Satyam- Hit The Pay
Button",
                                "message": "Starting at just ₹999*.
Secure yourself against rising medical costs right away!",
                                "headerSubText": "{{templatePush}}",
                                "largeIconUrl":
"https://imgstatic.phonepe.com/images/notifications/900/480/notification-15-04-22-health999.png",
                                "fontColor": "red",
                                "bgColor": "#f00000"
                            }
                        }
                    },
                    "nav": {
                        "key": "action_redirection",

```

```

        "params": {
            "deepLinkIOS":
"phonepe://sachetInsurance?category=HEALTH_INSURANCE&product=SIMPLE_HEALTH",
            "deepLinkAndroid":
"phonepe://externalapp?externalAppMeta=ewogICAgICAgICJwYWNRyYwdlTmFtZSI6ImNvbS5p
bmRlcy5hcHBzdG9yZSIsCiAgICAgICAgImRlZmFlbHRSZWZWRpcmVjdGlvb1VyYbCI6Imh0dHBzOi8vaWF
zLm9uZWxpbmsubWUvYajN6by85cWJxaTFyOSIsCiAgICAgICAgImRlZXBsaW5rIjoiaHR0cHM6Ly9pYX
Mub25lbGluay5tZS9qM3pvLzlxYnFpMXI5IgoKfQ==",
            "action_nav": "sachetInsuranceHome",
            "redirection_data": {
                "data": [
                    {
                        "key": "category",
                        "value": "HEALTH_INSURANCE",
                        "isEncoded": False
                    },
                    {
                        "key": "productType",
                        "value": "SIMPLE_HEALTH",
                        "isEncoded": False
                    }
                ]
            },
            "callToAction": [{
                "text": "test",
                "action_nav": "sachetInsuranceHome",
                "redirection_data": {
                    "data": [
                        {
                            "key": "category",
                            "value": "HEALTH_INSURANCE",
                            "isEncoded": False
                        },
                        {
                            "key": "productType",
                            "value": "SIMPLE_HEALTH",
                            "isEncoded": False
                        }
                    ]
                }
            }]
        },
        "fallbackStrategy": "NONE",
        "deferment": {
            "type": "FIXED_DELAY",
            "minutes": 3
        },
    },

```

```
        "expiresAt": current_epoch_milliseconds,
        "nav": {
            "key": "action_redirection",
            "params": {
                "deepLinkIOS":
"phonepe://sachetInsurance?category=HEALTH_INSURANCE&product=SIMPLE_HEALTH",
                "deepLinkAndroid":
"phonepe://externalapp?externalAppMeta=ewogICAgICAgICJwYWNRyYWdlTmFtZSI6ImNvbS5p
bmR1cy5hcHBzdG9yZSIsCiAgICAgICAgImRlZmF1bHRSZWRpcmVjdGlvb1VyYyB1bG90dHBzOi8vd3d
3LmluZHVzYXBwc3RvcuUuY29tLyIsCiAgICAgICAgImRlZXBsaW5rIjoiaHR0cHM6Ly9pYXMub251bG
luay5tZS9qM3pvLzlxYnFpMXI5IgoKfQ==",
                "action_nav": "sachetInsuranceHome",
                "redirection_data": {
                    "data": [
                        {
                            "key": "category",
                            "value": "HEALTH_INSURANCE",
                            "isEncoded": False
                        },
                        {
                            "key": "productType",
                            "value": "SIMPLE_HEALTH",
                            "isEncoded": False
                        }
                    ]
                }
            },
            "callToAction": [{
                "text": "PAY",
                "action_nav": "sachetInsuranceHome",
                "redirection_data": {
                    "data": [
                        {
                            "key": "category",
                            "value": "HEALTH_INSURANCE",
                            "isEncoded": False
                        },
                        {
                            "key": "productType",
                            "value": "SIMPLE_HEALTH",
                            "isEncoded": False
                        }
                    ]
                }
            }
        ],
        "properties": {
            "visibilityImportance": "URGENT"
        }
    }
```

```

    },
    {
      "category": "ANNOUNCEMENT",
      "scope": "INTERSTITIAL",
      "restrictions": {
        "maxAttempts": 2,
        "coolOffDuration": {
          "timeSpec": "MINUTE",
          "units": 1
        }
      },
      "template": {
        "templateId": "HTML",
        "templateParams": {
          "value": {
            "url":
"https://www.phonepe.com/apollo/emailers/2023/june/05/central-mapper-popup/en.h
tml",
            "dismissable": False
          }
        }
      },
      "fallbackStrategy": "NONE",
      "deferment": {
        "type": "NONE"
      },
      "expiresAt": current_epoch_milliseconds,
      "nav": {
        "key": "action_redirection",
        "params": {
          "deepLinkIOS":
"phonepe://sachetInsurance?category=HEALTH_INSURANCE&product=SIMPLE_HEALTH",
          "action_nav": "sachetInsuranceHome",
          "redirection_data": {
            "data": [
              {
                "key": "category",
                "value": "HEALTH_INSURANCE",
                "isEncoded": False
              },
              {
                "key": "productType",
                "value": "SIMPLE_HEALTH",
                "isEncoded": False
              }
            ]
          }
        }
      },
      "callToAction": [

```

```

        {
            "text": "PAY",
            "action_nav": "sachetInsuranceHome",
            "redirection_data": {
                "data": [
                    {
                        "key": "category",
                        "value": "HEALTH_INSURANCE",
                        "isEncoded": False
                    },
                    {
                        "key": "productType",
                        "value": "SIMPLE_HEALTH",
                        "isEncoded": False
                    }
                ]
            }
        }
    ],
    {
        "scope": "INBOX",
        "template": {
            "templateId": "ICON_TITLE_SUBTITLE",
            "templateParams": {
                "value": {
                    "title": "SATYAM_INTERVENTION",
                    "subTitle": "Click here",
                    "iconURL":
"https://imgstatic.phonepe.com/images/app-icons/wealth-management/mutual-funds/
providers/{height}/{width}/SUPER_FUND_AGGRESSIVE.png"
                }
            },
            "nav": {
                "key": "action_redirection",
                "params": {
                    "deepLinkIOS":
"phonepestage://dg?providerId=SAFEGOLD",
                    "deepLinkAndroid":
"phonepe://externalapp?externalAppMeta=ewogICAgICAgICJwYWNRyWdlTmFtZSI6ImNvbS5w
aG9uZXBlLnN0b2NrYnJva2luZyIsCiAgICAgICAgImRlZmF1bHRSZWRRpcmVjdGlvb1VyYyCI6Imh0dHB
zOi8vc2hhcmVtYXJrZXQub25lbGluay5tZS91TUhaL2Rpd2FsaTIzbXVodXJhdCI6CiAgICAgICAgICAgIm
RlZXBsaw5rIjoiaHR0cHM6Ly9zaGFyZW1hcmtldC5vbmlld3VNSFovZG13YWxpMjNtdWh1c
mF0IgoKfQ==",
                    "action_nav": "genReactContainer",
                    "redirection_data": {
                        "data": [
                            {

```

```

        "key": "bundleName",
        "isEncoded": "false",
        "value": "brickbatbridge"
    },
    {
        "key": "config",
        "isEncoded": "false",
        "value": "brickbatbridge"
    }
]
}
}
},
"deferment": None
}
],
"campaignId": "CM2204131951551550343789",
"syncSections": [],
"communicationIntent": "TRANSACTIONAL",
"utm_source": "Push",
"utm_medium": "Hawkeye",
"utm_campaign":
"HE@CM2204131951551550343789@1@d675e16c-8285-455a-8649-1046d3098c50"
},
"properties": {
    "editable": True,
    "deletable": True,
    "override": True,
    "unsubscribeSupported": False
},
"customParams": {
    "customTag2": "1",
    "customTag1": "CM2204131951551550343789",
    "customTag4":
"HE@CM2204131951551550343789@1@d675e16c-8285-455a-8649-1046d3098c50",
    "customTag3": user_id,
    "customTag5": ""
},
"sentAt": one_day_ago_epoch,
"expiresAt": current_epoch_milliseconds,
"version": 2,
"destination": {
    "mailboxName": "CONSUMER_U2302161303194695860743@zencast",
    "type": "USER_RESTRICTED"
}
}
}

```

Json file

```
{
  "mocking_stage_endpoint": {
    "base_url": "http://userservice.nixy.stg-drove.phonepe.nb6"
  },
  "mock_environment": {
    "mock_env": "mocking_stage_endpoint"
  },
  "crm_stage_endpoint": {
    "crm_base_url": "http://zencast.nixy.stg-drove.phonepe.nb6/"
  }
}
```

```
import pytest
import requests
import time
from Utilities.utils import device_id

@pytest.mark.crm
def test_crm_push_notification(call_crm_base_url, custom_header_crm,
                              crm_cta_payload, fetch_userid):
    url = f"{call_crm_base_url}v1/communication/send/push/unicast"
    print(crm_cta_payload)
    send_request = requests.post(url, headers=custom_header_crm,
                                  json=crm_cta_payload)
    response = send_request.json()
    request_id = crm_cta_payload.get("requestId")
    grouping_key = crm_cta_payload.get("appMessage", {}).get("groupingKey")
    sent_at = crm_cta_payload.get("appMessage", {}).get("sentAt")
    expires_at = crm_cta_payload.get("appMessage", {}).get("expiresAt")
    print(response)
    response_message_id = response.get("data", {}).get("messageId")
    time.sleep(10)
    capture_deviceid = device_id()
    payload = {
        "requestId": request_id,
        "entityId": fetch_userid,
        "deviceId": capture_deviceid,
        "profileId": "CONSUMER_PROMOTIONAL",
        "appMessage": {
            "version": 2,
            "messageId": response_message_id,
            "groupingKey": grouping_key,
            "source": {
                "type": "PHONEPE"
            },
        },
    },
```



```

        "data": {
            "syncSections": [
                {
                    "type": "crm_kill_switch",
                    "params": {
                        "createdAt": sent_at,
                        "killSwitch": {
                            "mode": "PERPETUAL",
                            "killSwitchScope": {
                                "campaignId": "testcampaign1",
                                "scope": "CAMPAIGN"
                            }
                        }
                    }
                }
            ],
            "placements": [],
            "campaignId": "testcampaign1",
            "communicationIntent": "PROMOTIONAL",
            "utm_source": "Push",
            "utm_medium": "Hawkeye",
            "utm_campaign": "testcampaign"
        },
        "properties": {
            "editable": True,
            "deletable": True,
            "override": True,
            "unsubscribeSupported": False
        },
        "sentAt": sent_at,
        "expiresAt": expires_at
    }
}

send_kill_request = requests.post(url, headers=custom_header_crm,
json=payload)
kill_response = send_kill_request.json()
print(kill_response)

```

```

import time
import pytest
import requests
from Utilities.utils import device_id
from conftest import fetch_userid

@pytest.mark.androidcrm
def test_android_crm(call_crm_base_url, custom_header_crm,
crm_PN_and_inbox_Android, fetch_userid):
    url = f"{call_crm_base_url}v1/communication/send/push/unicast"

```

```

send_request = requests.post(url, headers=custom_header_crm,
json=crm_PN_and_inbox_Android)
print(crm_PN_and_inbox_Android)
response = send_request.json()
print(send_request)
request_id = crm_PN_and_inbox_Android.get("requestId")
response_message_id = response.get("data", {}).get("messageId")
grouping_key = crm_PN_and_inbox_Android.get("appMessage",
{}).get("groupingKey")
sent_at = crm_PN_and_inbox_Android['appMessage']['sentAt']
expires_at = crm_PN_and_inbox_Android['appMessage']['expiresAt']
print(response)
time.sleep(10)
capture_deviceid = device_id()
user_id = fetch_userid
payload = {
    "requestId": request_id,
    "entityId": user_id,
    "deviceId": capture_deviceid,
    "profileId": "CONSUMER_TRANSACTIONAL",
    "appMessage": {
        "version": 2,
        "messageId": response_message_id,
        "groupingKey": grouping_key,
        "source": {
            "type": "PHONEPE"
        },
        "data": {
            "syncSections": [
                {
                    "type": "crm_kill_switch",
                    "params": {
                        "createdAt": sent_at,
                        "killSwitch": {
                            "mode": "PERPETUAL",
                            "killSwitchScope": {
                                "campaignId": "CM2204131951551550343789",
                                "scope": "CAMPAIGN",
                                "placements": ["DRAWER,INBOX,INTERSTITIAL"]
                            }
                        }
                    }
                }
            ]
        },
        "placements": [],
        "campaignId": "CM2204131951551550343789",
        "communicationIntent": "TRANSACTIONAL",
        "utm_source": "Push",
        "utm_medium": "Hawkeye",

```

