QUALITY COUNCIL OF INDIA
QCI Creating an Ecosystem for Quality

NPNT Test Tool | User Documentation

Table of Contents

About

No Permission No Take-off test tool, henceforth known as NPNT test tool is an initiative by QCI to check NPNT-compliance of an UAV through a Web Application. The Application will allow QCI to enable Certification Bodies to conduct verification of NPNT functionality and help them track and maintain test records.

The whole workflow

A tester user shall login to the portal, and generate a new test record. In this new test record, the user fills in details provided by the UAV manufacturer. Once this form is submitted, a bundle of Permission Artifacts will be generated. The tester will be able to download these files and upload it into a drone and activate it. Then the tester shall retrieve a log file from the drone and upload it into the portal. On upload, the user shall receive a report of compliance.

Test Cases for the NPNT functionality

NPNT is implemented by the Registered Flight Module (RFM) of the RPAS. Two sets of tests have been defined for under NPNT function compliance checking.

Arm Test

For the first set, permission artifacts are fed to the RPAS and the response of RPAS is noted and fed to the test tool. No flight happens. The observed and expected output of RPAS are to match for each permission artifact.

The tool may generate multiple permission artifacts randomly from the below cases and the RPAS output should always match the expected output for that permission artifact.

1. Signature on permission is invalid : RPAS should report invalid signature
   → Bad Signature.
2. DroneID does not match: RPAS should report incorrect permission
   → Bad Drone Id
3. Permission valid but current time is out of range: RPAS should report out of time limit
   → Bad Time Limit
4. Permission valid but current location is out of geo-fence: RPAS should report out of Geo-fence
   → Bad Geo Limit
5. Permission valid but UIN is incorrect: RPAS should report incorrect UIN.
   → Bad UIN
6. Permission valid but Operator ID is incorrect: RPAS should report incorrect Operator ID.
   → Bad Operator ID
7. Permission artifact is valid. RPAs should arm.
   → Valid Permission

Flight Test

For the second set, the testing tool will generate valid permission artifacts for checking the log validity part. After loading the permission artifact, there will be a flight and the log file generated for that permission artifact would be uploaded to the test tool.

Flights to be performed:

1. Make 2 consecutive flights without any breach; that is no time limit and geo-fence breach
2. Make a flight with time limit breach
3. Make a flight with geo fence breach

The tool checks for formats of the RPAS log file and checks if specific case outputs match the expected output. Casewise check details:

1. Log Format Verification

a. Performed on all logs
b. Log Schema is as described in the Guidance Manual.
c. Entry types are as per the manual (Case/ Format sensitive):
   i. TAKEOFF/ARM
   ii. GEOFENCE_BREACH
   iii. TIME_BREACH
   iv. LAND/DISARM
d. Logging frequency is at a rate greater than or equal to 1Hz (1 second per log) in case of below entry types
   i. GEOFENCE_BREACH
   ii. TIME_BREACH

2. Log Validation

a. No breach log 1 & No breach log 2
   i. Signature is verified
   ii. No breach log 2 contains previous log hash of No breach log 1
   iii. Entry type GEOFENCE_BREACH and TIME_BREACH are not present
   iv. Entry type TAKEOFF/ARM and LAND/DISARM are present
   v. All timestamps are within time range of permission artefact
   vi. All geo data (Longitude, Latitude and Altitude) is within geo data entered during test case generation.
b. Geo Fence breach log
   i. Signature is verified
   ii. All timestamps are within range of permission artefact
   iii. Entry type TAKEOFF/ARM is present and has valid timestamp and geo location data
   iv. Entry type GEOFENCE_BREACH has valid timestamp and invalid geo location data
   v. Logging frequency for GEOFENCE_BREACH is greater than or equal to 1Hz
   vi. Entry type GEOFENCE_BREACH is present
   vii. Entry type TIME_BREACH is not present
   viii. Entry type LAND/DISARM is present has valid timestamp and valid geo location data
   ix. Remark: Breach the fence only once for the test

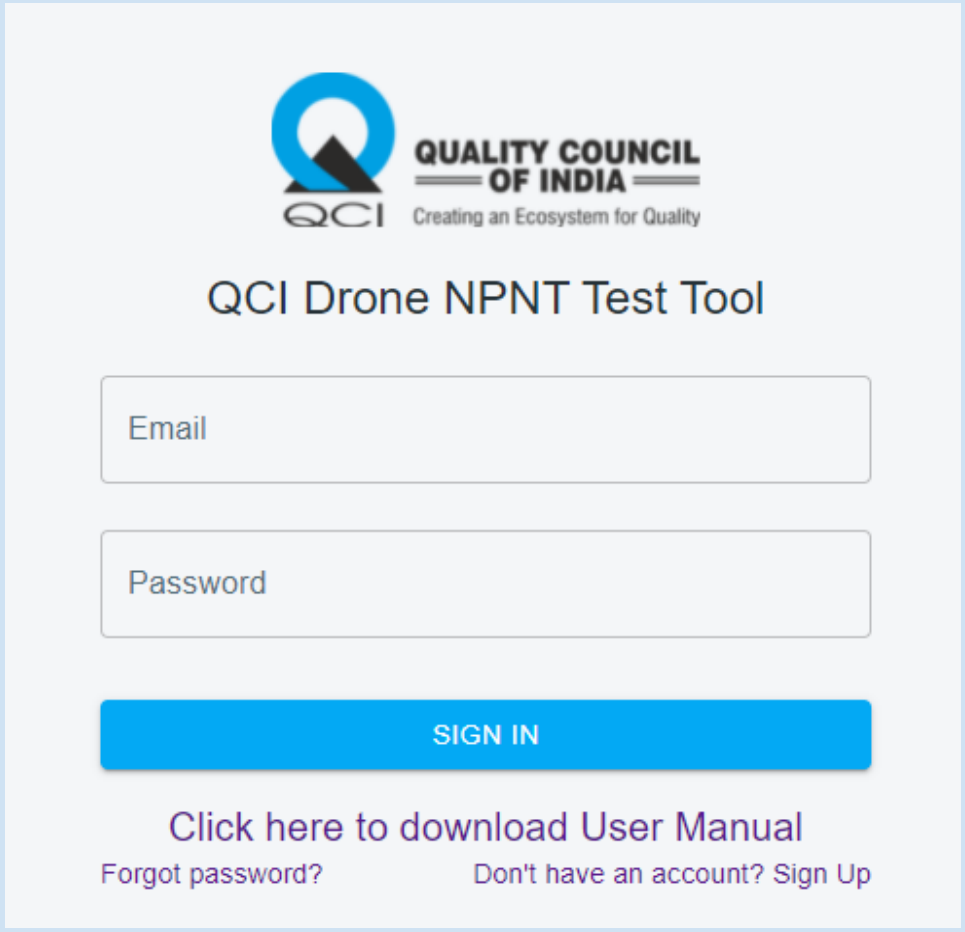
c. Time limit breach log
    i. Signature is verified
    ii. All geo location data are within range of permission artefact
    iii. Entry type TAKEOFF/ARM is present and has valid timestamp and geo location data
    iv. Entry type TIME_BREACH has invalid timestamp and valid geo location data
    v. Logging frequency for TIME_BREACH is greater than or equal to 1Hz
    vi. Entry type TIME_BREACH is present
    vii. Entry type GEOFENCE_BREACH is not present
    viii. Entry type LAND/DISARM is present and has invalid timestamp and valid geo location data

For validating signatures, corresponding public keys of DigitalSky and RPAS are provided to the Tool. Formats for permission and Log files would be as described in the Guidance Manual.

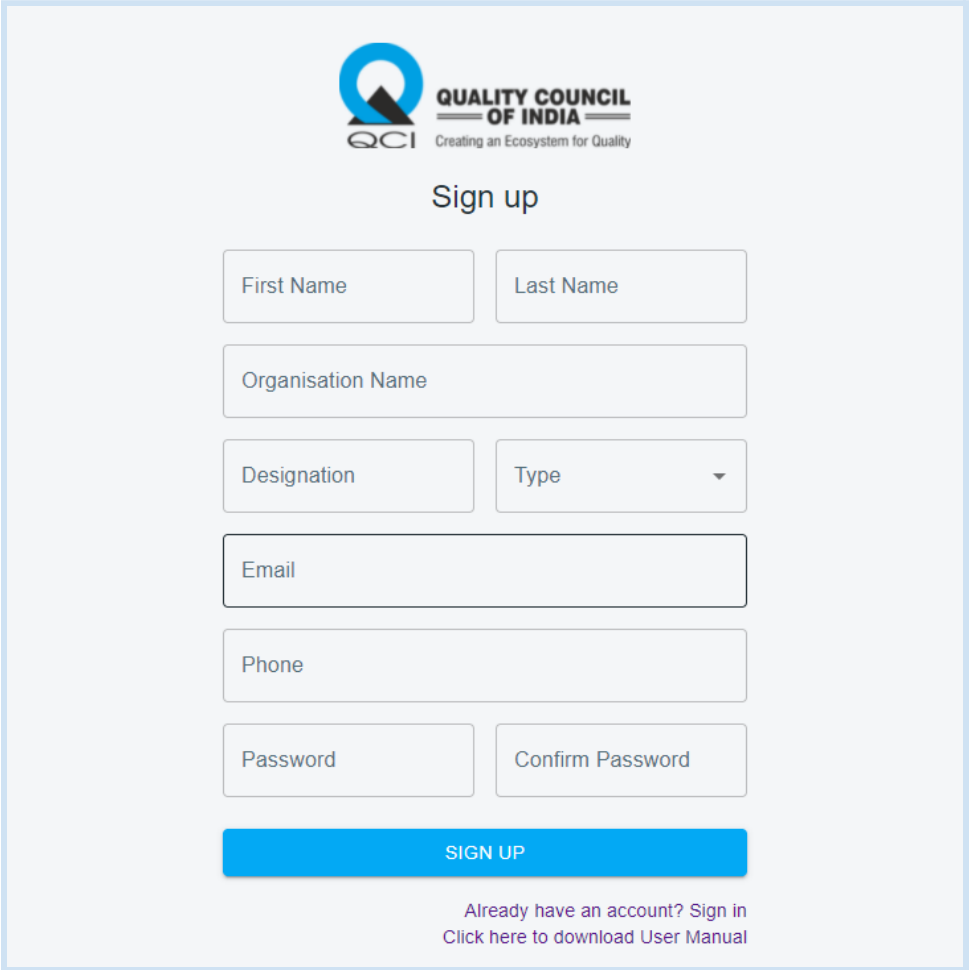
Standard Operating Procedure

1. Sign in/ Sign up

1.1. Visit url NPNT-Tool. Users shall be able to see on screen as shown here in the image.



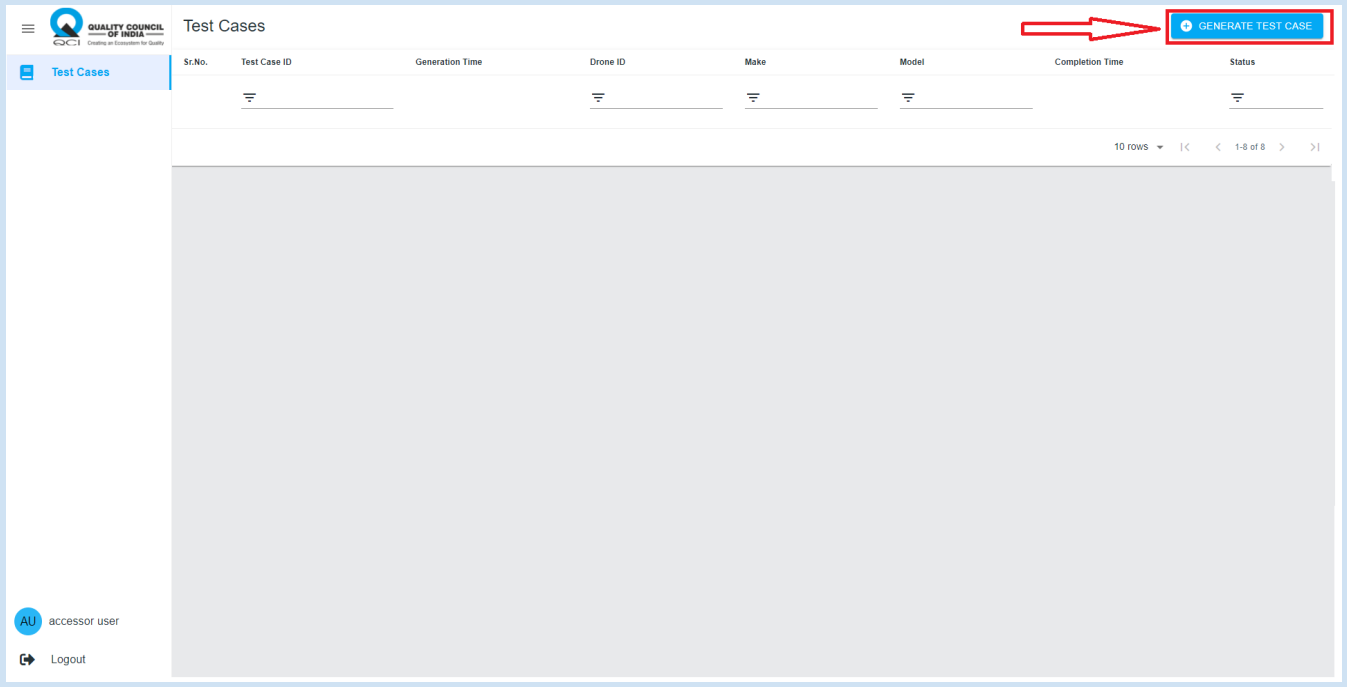
1.2. Click on “**Sign Up**” to create a new account.
Users shall be able to see on screen as shown here in the image



1.3. Login using credentials used to create an account.

2. Generate Test Case

2.1. Click the **Generate test case button** on the **right top corner** to generate a test case for your drone.

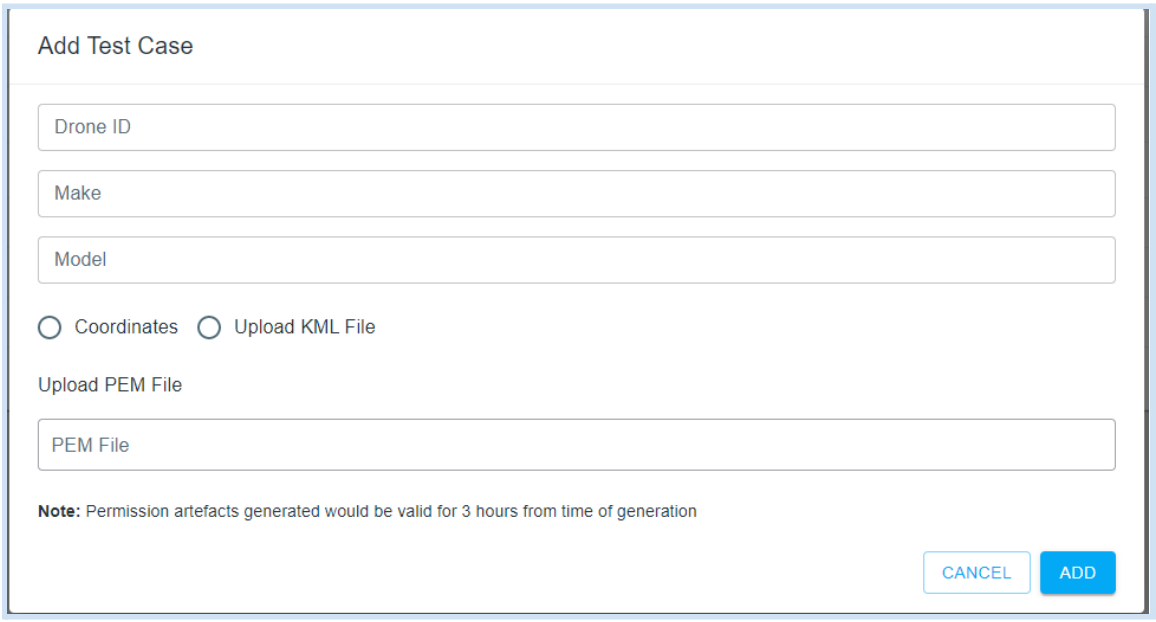


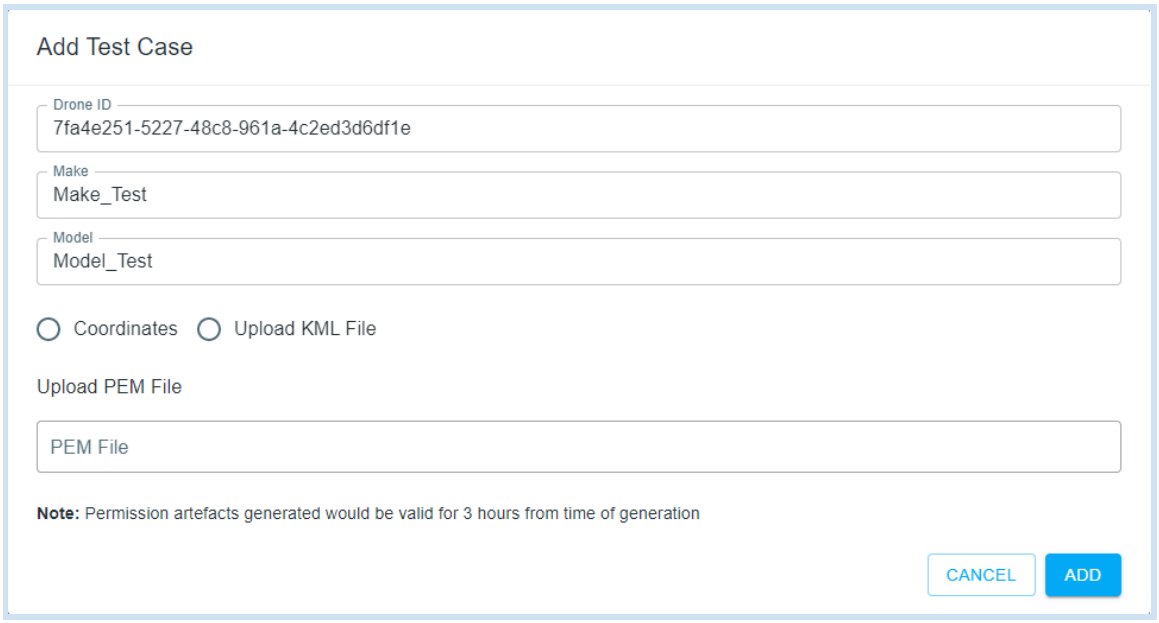
2.2. A form will open. Fill drone information in first three tabs:

2.2.1. Drone ID - UUID of the drone

2.2.2. Manufacturer Name

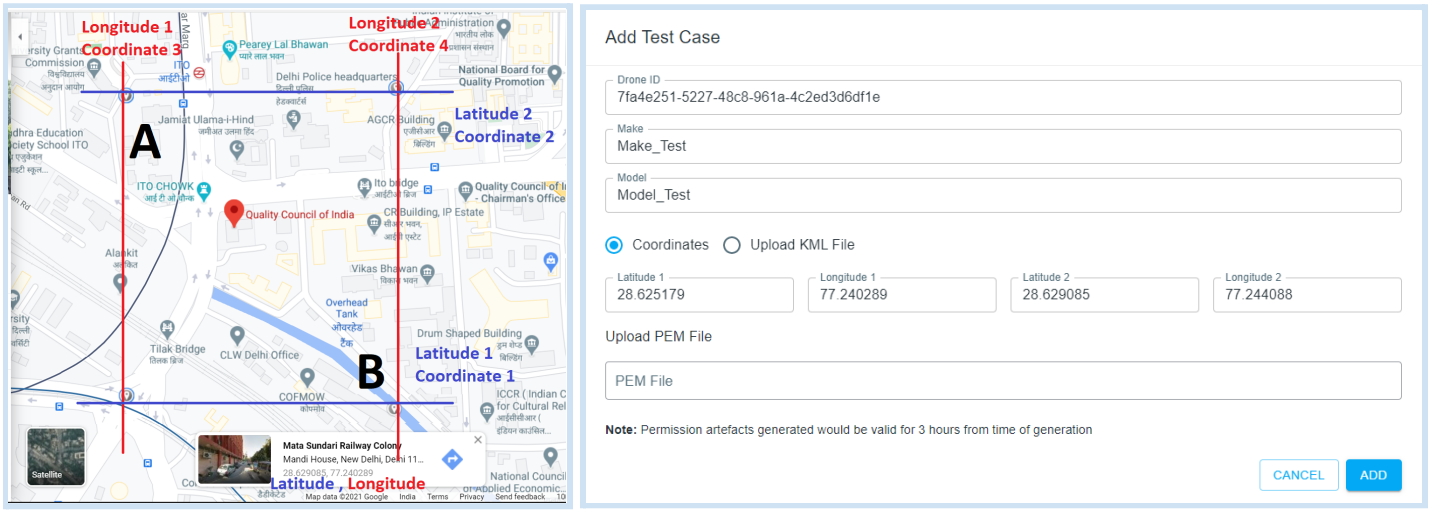
2.2.3. Model Name



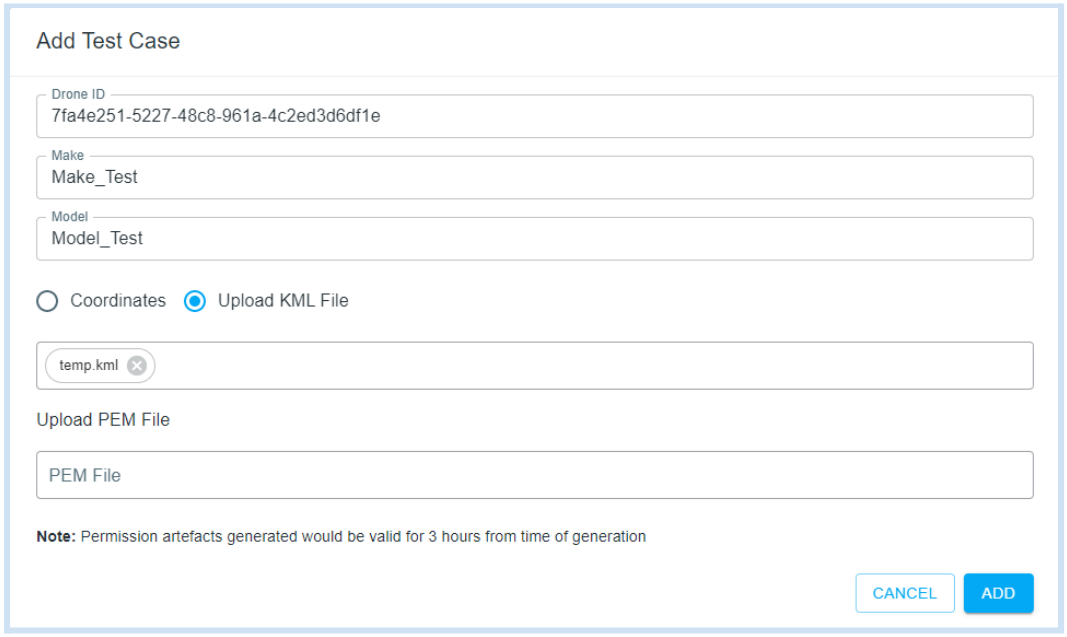


2.3. Fill data corresponding to the location in which the test is to be conducted. This would act as a **Geo Fence** boundary for the test. There are 2 options to do it.

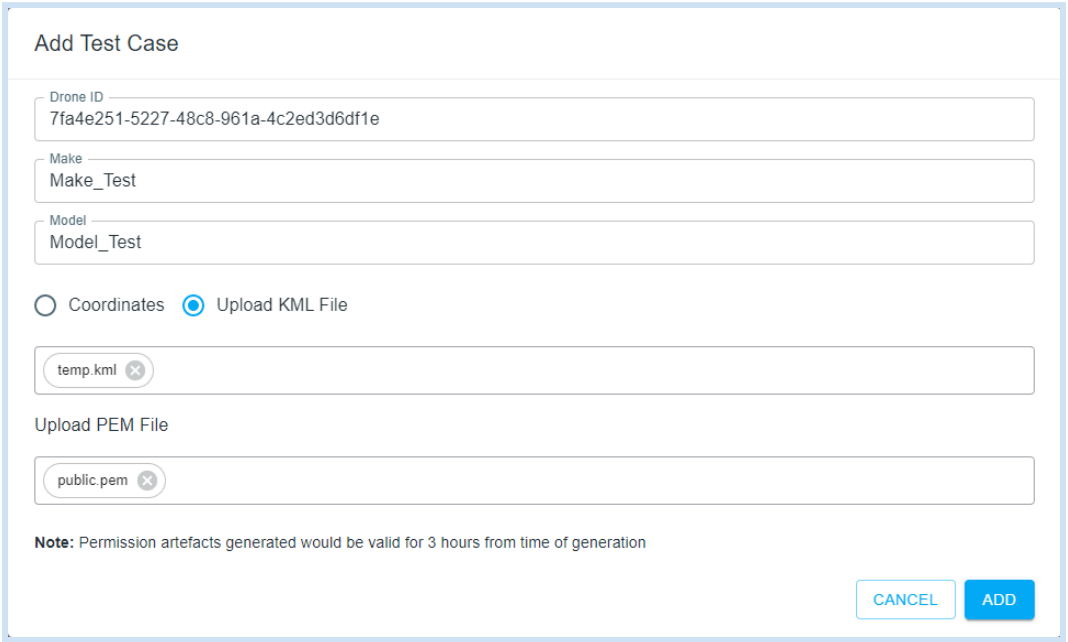
2.3.1. Option1: Coordinates : Latitude and longitude data via maps (eg: Google Maps)
Eg:
Latitude 1: 28.625179
Latitude 2: 28.629085
Longitude 1:77.240289
Longitude 2: 77.244088



2.3.2. Option2: Upload kml file
Drag and drop or click and select to upload a Kml file.

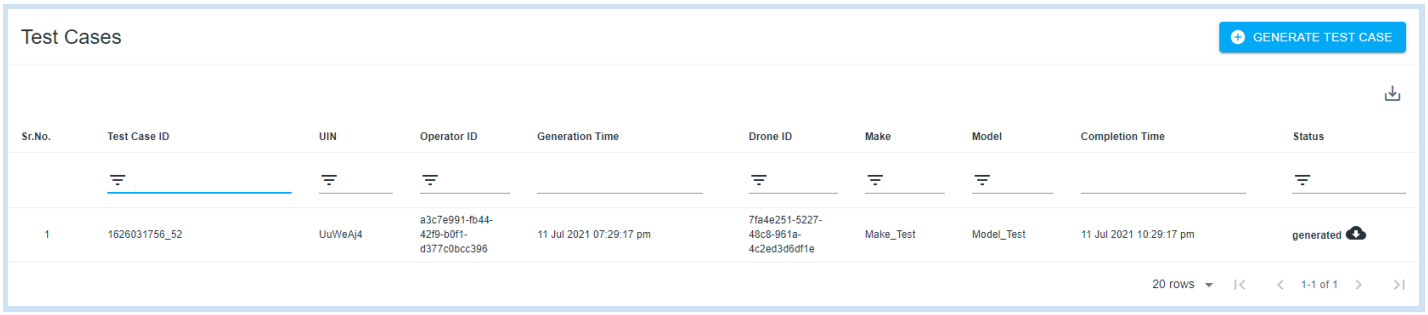


2.4. Drag and drop or click and select to upload drone public key (.pem file)



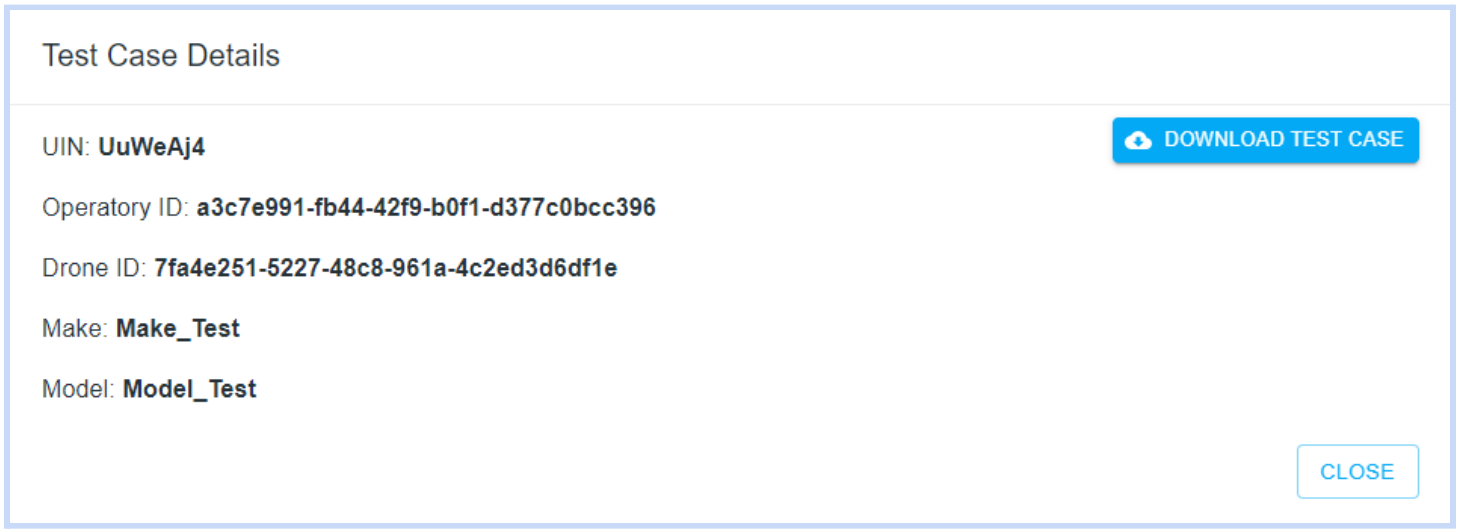
2.5. Click ADD to generate a test case.
Remark: Test case files generated are valid for 3 hrs only from time of generation.

2.6. A test case with a dummy UIN and Operator ID meant only for NPNT test will be generated.
2.6.1. The user must update the UIN and Operator ID in the drone manually.
2.6.2. Use the **Generated** button to download permission artefacts.



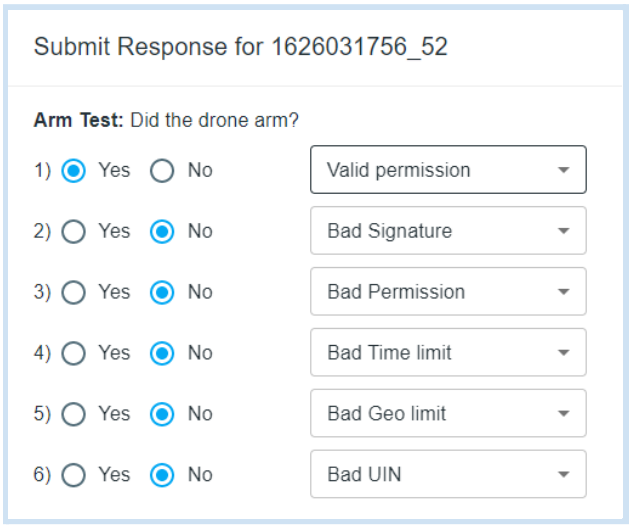
2.7. A folder by the name containing test case id generated containing 7 files should be downloaded.
2.7.1. Folder Name: (test_cases_<Test case ID>.xml)
→test_cases_1626031756_52.zip
File Names:
→permission_artifact_1.xml
→permission_artifact_2.xml
→permission_artifact_3.xml
→permission_artifact_4.xml
→permission_artifact_5.xml
→permission_artifact_6.xml
→permission_artifact_breach.xml

2.8. To redownload the test case files or check complete details click on the Test Case ID



3. Conduct Arm Tests

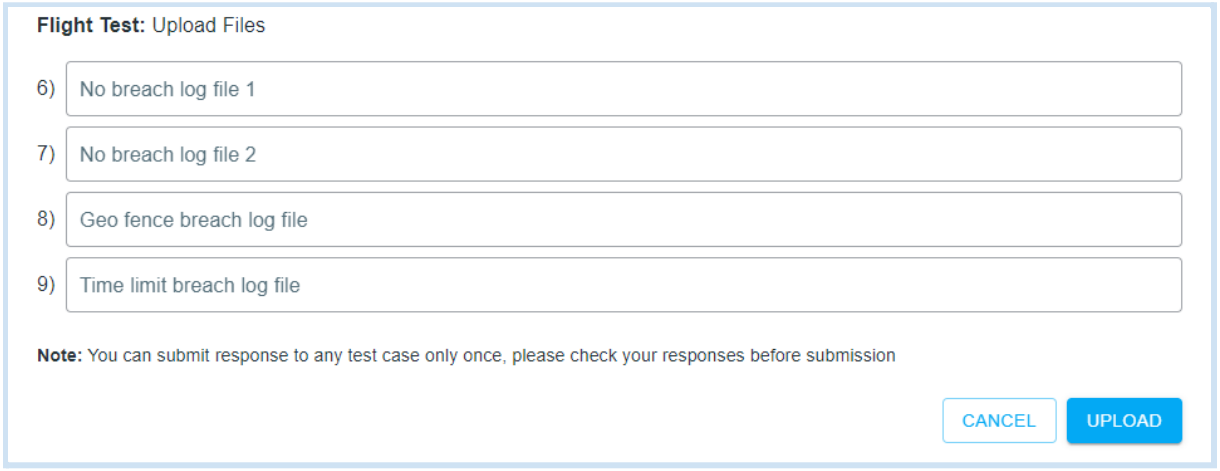
3.1. Files to be used : Permission_artifact_1.xml to permission_artifact_6.xml
3.2. One by one upload permission artefacts file into the drone and check whether the drone ARMs for flight (turns on) or not.
3.3. Expected response:
  3.3.1. Yes/ No: Select Yes if the drone was able to Arm, else No
  3.3.2. DropDown: Select the error reported by the DroneFrom the drop down



3.4. Remark:
For only one of the permission artifacts, the drone will arm, rest would give error at random.

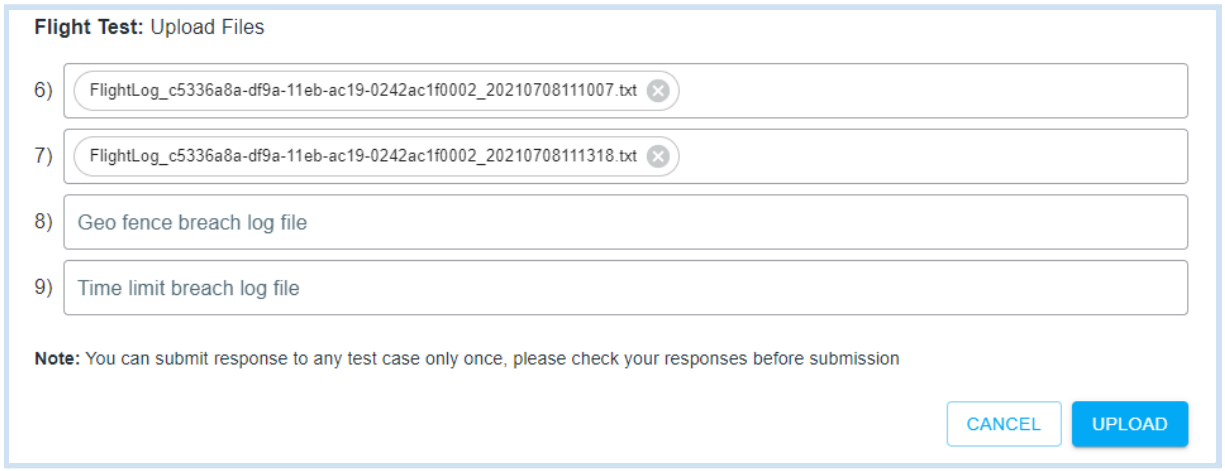
4. Conduct Flight Tests

4.1. File to be used : Permission_artifact_breach.xml
4.2. Flights to be conducted: 4
4.3. Remark: This permission artefact is valid for 3 hours from time of generation.
4.4. Logs to be uploaded as specified in below sections.



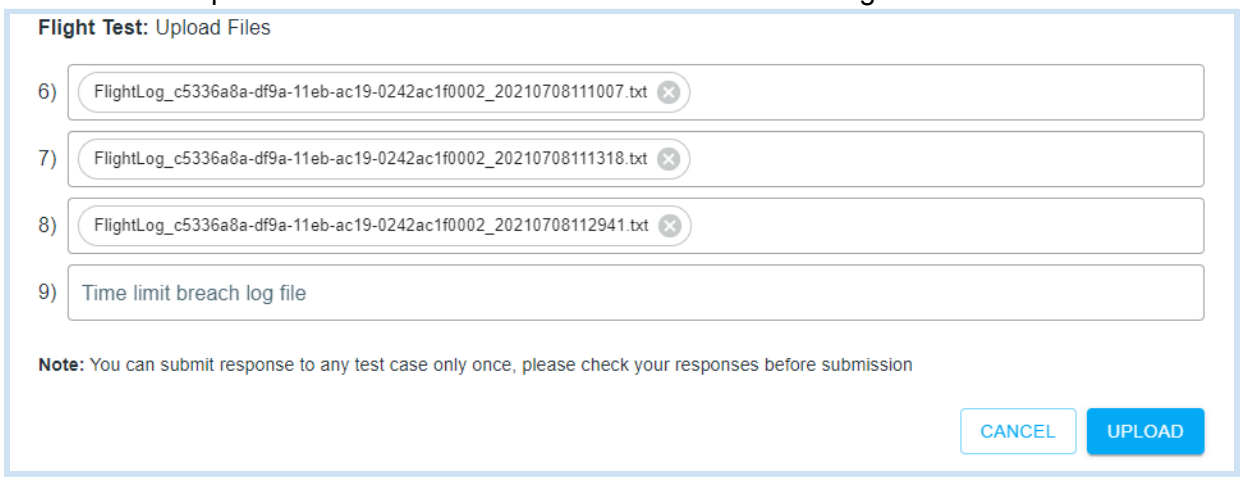
4.5. No Breach Test

4.5.1. Fly in the location marked earlier in the valid time period and return to home position.
4.5.2. Perform this action twice.
4.5.3. Extract both logs. These logs should be consecutive.
4.5.4. Upload first file in field labeled: No breach log 1
4.5.5. Upload second file in field labeled: No breach log 2



4.6. Geo Fence Breach Test

4.6.1. Start the flight in the location marked earlier in the valid time period.
4.6.2. Fly outside the geo fence.
4.6.3. Come back to home/ inside geo fence.
4.6.4. Extract the log.
4.6.5. Upload first file in field labeled: Geo fence breach log file



4.7. Time Limit Breach Test

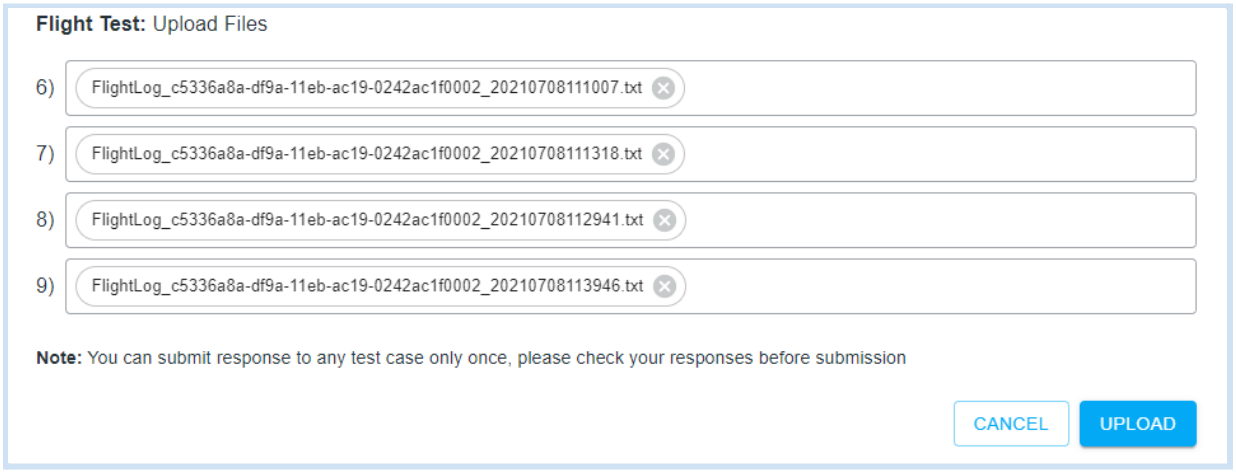
4.7.1. Start the flight in the location marked earlier in the valid time period.
Recommendation:
Start the flight 5-10 minutes before end_time. If a permission artifact is generated at 8am, it is valid till 11 am (3 hrs + generation time = end_time).

4.7.2. Fly beyond the end_time as per permission artifact .

4.7.3. Come back to home/ inside geo fence.

4.7.4. Extract the log.

4.7.5. Upload first file in field labeled: Time Limit breach log file



5. Submit Response

5.1. Click “Upload” to submit responses and upload logs for validation.

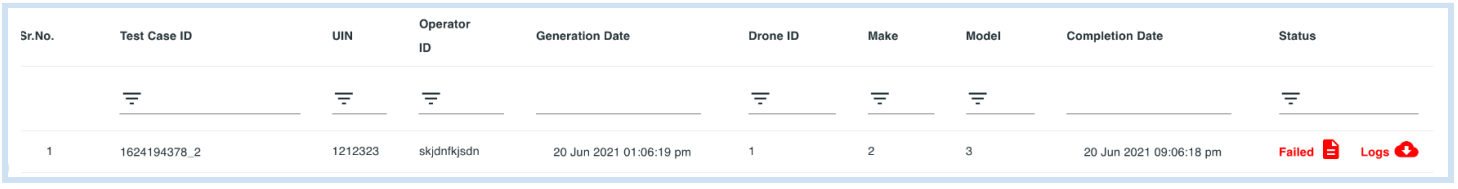
5.2. Remark: Users can upload the responses only once.

5.3. An email confirmation will also be sent to the user’s email id.

5.4. Depending on the checks the user shall receive either a Success report or a Failure report with Error log.

5.5. Click the Report Icon to download the report.

5.6. Click the Log Icon to download the Error log.



Sr.No.	Test Case ID	UIN	Operator ID	Generation Date	Drone ID	Make	Model	Completion Date	Status
1	1624194378_2	1212323	skjdnfkjsdn	20 Jun 2021 01:06:19 pm	1	2	3	20 Jun 2021 09:06:18 pm	Failed Logs

6. Annexure

6.1. Sample Permission Artifact

<?xml version="1.0"?>

<UAPermission txnId="f115f60a-8b4d-4bd8-93e1-329be89a6349" ttl="" pilotPinHash="" permissionArtifactId="91d9abfc-e7fd-11eb-9104-0242ac1f0002" lastUpdated="">

<!--UAPermission comments{"lastUpdated": "Latest update date of the Permission Artefact - Optional","permissonArtifactId": "Permission Artefact id generated by DigitalSky" }-->

-<Permission>

-<Owner pilotName="" pilotID="" operatorName="" operatorId="44537394-b702-4722-849d-b37fe75d4364">

<!--Owner comments{"OperatorId": "Temporary OperatorId generated by NPNT tool. For testing purpose only" }-->

<Pilot validTo="" uaplNo=""/>

</Owner>

-<FlightDetails>

-<UADetails uinNo="Uv9iQwu" UUID="abcd">

<!--UADetails comments{"UUID": "Drone Id (Identifier) of the RPAS provided by Manufacturer","uinNo": "Temporary UIN generated by NPNT tool. For testing purpose only" }-->

</UADetails>

-<FlightPurpose shortDesc="Provisional Test" frequency="10">

<!--FlightPurpose comments{"shortDesc": "Description of the flight given during flight plan creation","frequency": "Total number of permitted take offs given during flight plan creation" }-->

</FlightPurpose>

-<PayloadDetails payloadWeight="Not specified" payloadDetails="Not specified">

<!--PayloadDetails comments{"payloadWeight": "Maximum all up weight (in KG) for current flight plan given during flight plan creation","payloadDetails": "Payload details given during flight plan creation" }-->

</PayloadDetails>

-<FlightParameters maxAltitude="200" frequenciesUsed="" flightStartTime="2021-07-18T19:23:02" flightEndTime="2021-07-19T03:23:02">

<!--FlightParameters comments{"flightStartTime": "Start date and time for the flight given during flight plan creation","flightEndTime": "End date and time for the flight given during flight plan creation","frequenciesUsed": "Reserved for real-time frequency for communication","maxAltitude": "Maximum permissible altitude (in Feet) given during flight plan creation" }-->

-<Coordinates>

<Coordinate longitude="73.0281571986705" latitude="19.10828123526791"/>

<Coordinate longitude="73.02838956779853" latitude="19.10746192989367"/>

<Coordinate longitude="73.0296929274969" latitude="19.10782320005443"/>

<Coordinate longitude="73.02936092527486" latitude="19.10866514680841"/>

<Coordinate longitude="73.0281571986705" latitude="19.10828123526791"/>

<!--Coordinate comments{"latitude": "Latitude of the flight plan area coordinate given during flight plan creation","longitude": "Longitude of the flight plan area coordinate given during flight plan creation" }-->

</Coordinates>

</FlightParameters>

</FlightDetails>

</Permission>

-<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">

-<SignedInfo>

<CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>

<SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>

-<Reference URI="">

-<Transforms>

<Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>

<Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>

</Transforms>

<DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<DigestValue>IYUSpM3iX+REXYeU4zWU2giUZeqTZnGloV5zCs/qbfs=</DigestValue>

</Reference>

</SignedInfo>

<SignatureValue>Sv7kT6ojwO2tg3z0a5ZAw/qdPTmqW1sjz7+HuV7VncivkYUyPdDJ/bvAakIs8ml4yVwt
Y2b+IhdCXJ8HqiLAdjHt7dIzzB7rYjgw9mviuGD32wA324d2Gq8t9qtntz94Viwg8BwMOVNgReIM19Rzq
V3HNFee4OuKP95G8BIgaN0=</SignatureValue>

-<KeyInfo>

-<X509Data>

<X509Certificate>MIICujCCAaICCQDGxXPQwumdKDANBgkqhkiG9w0BAQsFADAfMQswCQYDVQQ
GEwJp bjEQMA4GA1UECgwHZGF0YW9yYzAeFw0yMTA3MDIxMTIzMjVaFw0yMzA2MjIxMTIz
MjVaMB8xCzAJBgNVBAYTAmluMRAwDgYDVQQKDAdkYXRhb3JjMIIBIjANBgkqhkiG
9w0BAQEFAAOCAQ8AMIIBCgKCAQEA1pEqdzM106GTx0IHF2J5Tia8rEnXi2tlp4Th
wY0Y8viLnxPX/EGlueEePXWQWw+Dc9usVliWKUDRBHEUZ4FLcxl3QBRkTfgul2S0
6FLLdUfVbeMiSq2gANsjGxNW/+hjkbPsot32fUT0DqboQJLMZzh9s6IEJIe42s/l
tYhziXOln1+oE6M+S7j7r6QHeU25kKMHIBsKD72S3dGBqGimz6ZGJTU3FOmx3m+B
863b9bICAw29nlE4XbkxRZHwGvAfB8yS8Y6+YFRc6lPrVEOOJR/YjnOAtaBhLCnM
t9/Yj3p9fw3QeptZaSNgJ5itnxu7if/9PNuPuUxIZA0n+mgeRwIDAQABMA0GCSqG
SIb3DQEBCwUAA4IBAQDEuxlb8jUop78g2mwaYbpTAgbpBgJRUVT4VyWCkGRSi4Bc
dHm0zhyZrVuf3GqSwMc+LnRn+/N8AWGHbEvMI7RHSu98biO7T1S6kidsPxKz9CL9
UaE5Wv+RbXxKDhg0MJpHD/WOWbQC3UPs8Ec6sy34QrAhjKTQP1m6zKKDlicmL+29
ku2h/TQv9uEeGa8foL6/bzHAtyYJJC2jovnsBstK9P7nJ1exL0kdqSWQ92xvRqno
sYkLo5k31+gK7KRKwKoi/y+TQDjYqvuZfAm/LmsS1h2TefgMtcbMxF774V3ATqhh
Bt4lFdOhSjgC8E0BhP6TyP+SF5mNAaJ1zhjIYBUS </X509Certificate>

</X509Data>

</KeyInfo>

</Signature>

</UAPermission>

6.2. Permission Artifact Data formats

Sr.	Tag and Attribute	Attribute Format	Comments
1	UAPermission.lastUpdated	String	Latest update date of the Permission Artefact - Optional
2	UAPermission.permissionArtifactId	UUID	Permission Artefact id generated by NPNT Tool
3	UAPermission.pilotPinHash	Empty	Unused
4	UAPermission.ttl	Empty	Unused
5	UAPermission.txnId	Empty	Unused
6	Permission.Owner.operatorId	UUID	Business identifier of the operator, temporarily generated by NPNT Tool
7	Permission.Owner.operatorName	String	Unused
8	Permission.Owner.Pilot.pilotId	UUID	Unused
9	Permission.Owner.Pilot.pilotName	String	Unused
10	Permission.Owner.Pilot.validUpto	Empty	Unused
11	Permission.FlightDetails.UADetails.UUID	String	Drone Id (Identifier) of the RPAS provided by Manufacturer
12	Permission.FlightDetails.UADetails.uinNo	String	UIN of the RPAS, temporarily generated by NPNT Tool
13	Permission.FlightDetails.FlightPurpose.frequency	Integer number	Total number of permitted take offs given during flight plan creation
14	Permission.FlightDetails.FlightPurpose.shortDesc	String	Unused
15	Permission.FlightDetails.PayloadDetails.payloadDetails	String	Unused
16	Permission.FlightDetails.PayloadDetails.payloadWeight	Float	Unused
17	Permission.FlightDetails.FlightParameters.flightEndTime	Date and time in UTC (YYYY-MM-DDTHH:mm:ss)	End date and time for the flight given during flight plan creation
18	Permission.FlightDetails.FlightParameters.flightStartTime	Date and time in UTC (YYYY-MM-DDTHH:mm:ss)	Start date and time for the flight given during flight plan creation
19	Permission.FlightDetails.FlightParameters.frequenciesUsed	Empty	Reserved for real-time frequency for communication
20	Permission.FlightDetails.FlightParameters.maxAltitude	Integer number	Maximum permissible altitude (in Feet) given during flight plan creation
21	Permission.FlightDetails.FlightParameters.Coordinates.Coordinate.latitude	String	Latitude of the flight plan area coordinate given during flight plan creation
22	Permission.FlightDetails.FlightParameters.Coordinates.Coordinate.longitude	String	Longitude of the flight plan area coordinate given during flight plan creation
24	Signature.SignedInfo.CanonicalizationMethod.Algorithm	String	http://www.w3.org/2001/10/xml-exc-c14n#

25	Signature.SignedInfo.CanonicalizationMethod.SignatureMethod	String	http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
26	Signature.SignedInfo.Reference.Transforms.Transform	String	http://www.w3.org/2000/09/xmldsig#enveloped-signature
27	Signature.SignedInfo.Reference.DigestMethod	String	http://www.w3.org/2001/04/xmlenc#sha256
28	Signature.SignedInfo.Reference.DigestValue	String	Digest generated
29	Signature.SignatureValue	String	Digital Signature Private key: NPNT Tool Certificate: NPNT Tool
30	KeyInfo.X509Data.X509Certificate	String	Drone public key

6.3. Sample Flight Log

6.3.1. Sample 1


```
{
  "signature":
"Nb4mgoxImQh8MbtMGZNvj3yKsWTTTWwcaSnLJROGRu2QBJ8aJ8TBZIah2GSUpgr9
ZDpRBnbwG8TuvHqMIvOn/pCi0Els03x2sm9w7EzZhW9eFeyKHNmxTyc4pltAmwIeubT
L4vF6S5gFIbphwXb9W23rBr8y6b8g/umNXa3B9NP3IBuuz8bx59Rq9yuwLS3VBmfdTQ
KtZ4CRBT25jaM7vpB6csQPATXtrXHrGRdt00Kv8huOtzLYBYDI2cVKwsQ98kiUFFICKp
CORQk+O8TiFySaczZkscdfYMzVAYJ3uYNqWzOkOpy2z1onJcCc1XxPq8v4PD1cwD8a
Re8UIHJsCQ==",
  "flightLog": {
    "permissionArtefact": "c5336a8a-df9a-11eb-ac19-0242ac1f0002",
    "previousLogHash":
"iz+gAlzsPWJwy1lebjGdkoaat3VQmuTR1ySEveH0Fm3mZIM2OCY6AzJD891feB7MT1
Uqu7ZAtYBnq4rQmK0hSUBd8vikzrTh+/ddpNxQM6N+TvSR/rtPee2YueXaRGxL7aWJNJ
aTg5XW04yL9cD3xoJgBDHu8Ih1G6fhvaLqItKeMwARdnDGV4YEAA1bkOiMrJozLxpCu
FFjEeTflTTTU/oYgpY2tymUYZZBOeozP+BsGKbomLkL7C+ItTvaMFAIfF93VekLJuw8l5d
bQV3VXE5SgLl6QJM9GKWAkIkqGKPe7jo5lYiLJtlWt6IhuGqBdQzjpHO3Q33OGEj1rTra
MA==",
    "logEntries": [
      {
        "entryType": "TAKEOFF/ARM",
        "timeStamp": 1625742798,
        "longitude": 73.028351,
        "latitude": 19.10783,
        "altitude": -0.4
      },
      {
        "entryType": "LAND/DISARM",
        "timeStamp": 1625742920,
        "longitude": 73.028343,
        "latitude": 19.107821,
        "altitude": 0.2
      }
    ]
  }
}
```

6.3.2. Sample 2


```
{
  "signature":
"ouiqeUWNX3X2xZDYRv+yE9rIFpZ9GlntnjbnSWkarU9grg4lT5HzmUO/hjo6JWL/0P80
B3evGr05YGG3t30yvKPLCDyGOikreAwCaISlnH3Y1saemW6dFhXbTx08pRxig/Duw/x/n
dZqeOipBULNNoExjsFjv7KuydmxznVO0XOexSacMOjtDXRoiKjMVKT/kddElvFjCCN1IFI
cSkk8nSuyFN/B1OYmEfMY3Fqvc2Mjv+KkQ90pM1dr5XbWoIxqWG5dhPWl1RExG/yC2j
h6JY7sCGFZHLZNtDNpqpdOvTLsP+2VqB6oKY+o+i6jYTzJXd8ylN6sViVM9fKTrxKFQ=
=",
  "flightLog": {
    "permissionArtefact": "c5336a8a-df9a-11eb-ac19-0242ac1f0002",
    "previousLogHash":
"Nb4mgoxImQh8MbtMGZNvj3yKsWTTTWwcaSnLJROGRu2QBJ8aJ8TBZIah2GSUpgr9
ZDpRBnbwG8TuvHqMIvOn/pCi0Els03x2sm9w7EzZhW9eFeyKHNmxTyc4pltAmwIeubT
L4vF6S5gFIbphwXb9W23rBr8y6b8g/umNXa3B9NP3IBuuz8bx59Rq9yuwLS3VBmfdTQ
KtZ4CRBT25jaM7vpB6csQPATXtrXHrGRdt00Kv8huOtzLYBYDI2cVKwsQ98kiUFFICKp
CORQk+O8TiFySaczZkscdfYMzVAYJ3uYNqWzOkOpy2z1onJcCc1XxPq8v4PD1cwD8a
Re8UIHJsCQ==",
    "logEntries": [
      {
        "entryType": "TAKEOFF/ARM",
        "timeStamp": 1625743781,
        "longitude": 73.028366,
        "latitude": 19.107832,
        "altitude": -0.2
      },
      {
        "entryType": "GEOFENCE_BREACH",
        "timeStamp": 1625743879,
        "longitude": 73.028259,
        "latitude": 19.107826,
        "altitude": 59.5
      },
      {
        "entryType": "GEOFENCE_BREACH",
        "timeStamp": 1625743880,
        "longitude": 73.028236,
        "latitude": 19.107824,
        "altitude": 59.5
      },
      {
        "entryType": "GEOFENCE_BREACH",
        "timeStamp": 1625743881,
        "longitude": 73.028221,
        "latitude": 19.107824,
        "altitude": 59.400002
      },
      {
        "entryType": "GEOFENCE_BREACH",
        "timeStamp": 1625743882,
```

```
  "longitude": 73.028206,
  "latitude": 19.107821,
  "altitude": 59.299999
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743883,
  "longitude": 73.028206,
  "latitude": 19.107821,
  "altitude": 59.299999
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743884,
  "longitude": 73.028221,
  "latitude": 19.107822,
  "altitude": 59.400002
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743885,
  "longitude": 73.028229,
  "latitude": 19.107824,
  "altitude": 59.400002
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743886,
  "longitude": 73.028236,
  "latitude": 19.107826,
  "altitude": 59.400002
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743887,
  "longitude": 73.028244,
  "latitude": 19.107826,
  "altitude": 59.400002
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743888,
  "longitude": 73.028244,
  "latitude": 19.107826,
  "altitude": 59.400002
},
{
  "entryType": "GEOFENCE_BREACH",
  "timeStamp": 1625743889,
  "longitude": 73.028252,
  "latitude": 19.107828,
```

```
      "altitude": 59.400002
    },
    {
      "entryType": "GEOFENCE_BREACH",
      "timeStamp": 1625743890,
      "longitude": 73.028252,
      "latitude": 19.10783,
      "altitude": 59.400002
    },
    {
      "entryType": "GEOFENCE_BREACH",
      "timeStamp": 1625743891,
      "longitude": 73.028259,
      "latitude": 19.107832,
      "altitude": 59.299999
    },
    {
      "entryType": "GEOFENCE_BREACH",
      "timeStamp": 1625743892,
      "longitude": 73.028259,
      "latitude": 19.107832,
      "altitude": 59.299999
    },
    {
      "entryType": "GEOFENCE_BREACH",
      "timeStamp": 1625743893,
      "longitude": 73.028267,
      "latitude": 19.107832,
      "altitude": 59.299999
    },
    {
      "entryType": "LAND/DISARM",
      "timeStamp": 1625743985,
      "longitude": 73.028358,
      "latitude": 19.10783,
      "altitude": 0.9
    }
  ]
 }
}
```

6.3.3. Sample 3


```
{
  "signature":
"E0N/vbV/mUbaw5eTSC9YRkcLTtwiJ/24vYJrfMtnDM+tixDO3OqCTjv6VTEjUKCHRvMz
UkdBC/81qoI+X83sTjYCimQZrKtE3+g9mZZCRN1TZTSYBMZ6TYDhvtimEW2gIrWDoH
erZarsGGdVNuMtpZMkRyJ7leQjDui/eCZVDmARsTzI9EJG6Wz9t6QF6rMr+oFPCcUxx1
OpCJbfCTRYFHzkjVeoWH7qzxLHHzwBLDtbVEKleJJ0VuTHayDM9Zw4+cUwyMP8BTb
Eg4M9Sd/3HQkcoke5J7IB8AidfNf9KuQ1M7XO7FzyxEB0cu1VCx42g7Ol7M4aAPe16C1
C078CAA==",
  "flightLog": {
    "permissionArtefact": "c5336a8a-df9a-11eb-ac19-0242ac1f0002",
    "previousLogHash":
"ouiqeUWNX3X2xZDYRv+yE9rIFpZ9Glntnjbn+SWkarU9grg4lT5HzmUO/hjo6JWL/0P80
B3evGr05YGG3t30yvKPLCDyGOikreAwCaISlnH3Y1saemW6dFhXbTx08pRxig/Duw/x/n
dZqeOipBULNNoExjsFjv7KuydmxznVO0XOexSacMOjtDXRoiKjMVKT/kddElvFjCCN1IFI
cSkk8nSuyFN/B1OYmEfMY3Fqvc2Mjv+KkQ90pM1dr5XbWoIxqWG5dhPWI1RExG/yC2j
h6JY7sCGFZHLZNtDNpqpdOvTLsP+2VqB6oKY+o+i6jYTzJXd8ylN6sViVM9fKTrxKFQ=
=",
    "logEntries": [
      {
        "entryType": "TAKEOFF/ARM",
        "timeStamp": 1625744386,
        "longitude": 73.028381,
        "latitude": 19.10784,
        "altitude": -0.1
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744740,
        "longitude": 73.028366,
        "latitude": 19.107836,
        "altitude": 50
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744741,
        "longitude": 73.028374,
        "latitude": 19.107836,
        "altitude": 50
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744742,
        "longitude": 73.028374,
        "latitude": 19.107836,
        "altitude": 50
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744743,
        "longitude": 73.028374,
```

```
        "latitude": 19.107834,
        "altitude": 49.900002
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744744,
        "longitude": 73.028374,
        "latitude": 19.107832,
        "altitude": 50
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744745,
        "longitude": 73.028374,
        "latitude": 19.10783,
        "altitude": 49.900002
      },
      {
        "entryType": "TIME_BREACH",
        "timeStamp": 1625744746,
        "longitude": 73.028374,
        "latitude": 19.10783,
        "altitude": 49.900002
      },
      {
        "entryType": "LAND/DISARM",
        "timeStamp": 1625744804,
        "longitude": 73.028374,
        "latitude": 19.107828,
        "altitude": 1.5
      }
    ]
  }
}
```

6.4. Flight Log Data Formats

Sr.	Attribute	Attribute Format	Comments
1	Signature	String	Digital Signature Private key: Drone Certificate: Drone
2	flightLog.permissionArtefact	String	Permission Artefact id generated by NPNT Tool (UUID)
3	flightLog.previousLogHash	String	SHA 256 of flight log content
4	flightLog.logEntries.entryType	String	"TAKEOFF/ARM", "GEOFENCE_BREACH", "TIME_BREACH", “LAND/DISARM",
5	flightLog.logEntries.timeStamp	Date and time in IST (YYYY-MM-DDTHH:mm:ss)	Timestamp during logging event (IST)
6	flightLog.logEntries.longitude	Float	Longitude during logging event (00.000000)
7	flightLog.logEntries.latitude	Float	Latitude during logging event (00.000000)
8	flightLog.logEntries.altitude	Float	Altitude during logging event (in meters)

6.5. Sample Pass Report

Test Report

Test Case Details

Date	July 11, 2021
User Email	
Test ID	1626031756_52
Drone ID	7fa4e251-5227-48c8-961a-4c2ed3d6df1e
Drone Model	Model_Test
Manufacturer Name	Make_Test
Test Result	Pass

Result Details

Test	Responses	Expected	Given	Result
Arm1	Test case: Test case result:	bad_sign False	bad_sign False	Pass
Arm2	Test case: Test case result:	valid_pa True	valid_pa True	Pass
Arm3	Test case: Test case result:	bad_drone_id False	bad_drone_id False	Pass
Arm4	Test case: Test case result:	bad_time False	bad_time False	Pass
Arm5	Test case: Test case result:	bad_geo False	bad_geo False	Pass
Arm6	Test case: Test case result:	bad_uin False	bad_uin False	Pass

Log Format Verification Details

Filename	Log Schema	Logging Frequency	Entry Type	Result
No breach log file 1	True	True	True	Pass
No breach log file 2	True	True	True	Pass
Geo breach log file	True	True	True	Pass
Time breach log file	True	True	True	Pass

Log Validation Details

Test	Responses	Expected	Given	Result
NO BREACH	Signature: Log Validation:	True True	True True	Pass
GEO BREACH	Signature: Log Validation:	True True	True True	Pass
TIME BREACH	Signature: Log Validation:	True True	True True	Pass

6.6. Sample Fail Report

Test Report

Test Case Details

Date	July 11, 2021
User Email	
Test ID	1626031756_52
Drone ID	7fa4e251-5227-48c8-961a-4c2ed3d6df1e
Drone Model	Model_Test
Manufacturer Name	Make_Test
Test Result	Fail

Result Details

Test	Responses	Expected	Given	Result
Arm1	Test case: Test case result:	bad_sign False	bad_sign False	Pass
Arm2	Test case: Test case result:	valid_pa True	valid_pa True	Pass
Arm3	Test case: Test case result:	bad_drone_id False	bad_drone_id False	Pass
Arm4	Test case: Test case result:	bad_time False	bad_time False	Pass
Arm5	Test case: Test case result:	bad_geo False	bad_geo False	Pass
Arm6	Test case: Test case result:	bad_uin False	bad_uin False	Pass

Log Format Verification Details

Filename	Log Schema	Logging Frequency	Entry Type	Result
No breach log file 1	True	True	True	Pass
No breach log file 2	True	False	True	Fail
Geo breach log file	True	False	True	Fail
Time breach log file	True	False	True	Fail

Log Validation Details

Test	Responses	Expected	Given	Result
NO BREACH	Signature: Log Validation:	True True	True True	Pass
GEO BREACH	Signature: Log Validation:	True True	False False	Fail
TIME BREACH	Signature: Log Validation:	True True	False False	Fail

6.7. Sample Error Log


```
{
   "no_breach_log_file_1": {
      "geo_breach_errors": [],
      "time_frequency_errors": [
         {
            "row": 1,
            "error_message": "logging frequency is above 1 seconds current 2021-07-08 11:11:50+00:00 -
previous 2021-07-08 11:10:07+00:00"
         }
      ],
      "entry_type_errors": [],
      "time_breach_errors": [
         {
            "row": 0,
            "error_message": "logged timestamp 2021-07-08 11:10:07+00:00 is outside time limit 2021-07-11
13:54:45.648191+00:00 - 2021-07-11 17:54:45.648191+00:00"
         },
      ]
   },
   "no_breach_log_file_2": {
      "log_hash_error": "Expected previous hash :
87c642569d133f68478fa4e4fcc173a8f1220f65ee58401f18342dd703e0442e but found
iz+gAlzsPWJwy1lebjGdkoaat3VQmuTR1ySEveH0Fm3mZIM2OCY6AzJD891feB7MT1Uqu7ZAtYBnq4rQmK0h
SUBd8vikzrTh+/ddpNxQM6N+TvSR/rtPee2YueXaRGxL7aWJNJaTg5XW04yL9cD3xoJgBDHu8Ih1G6fhvaLqIt
KeMwARdnDGV4YEAA1bkOiMrJozLxpCuFFjEeTflTTTU/oYgpY2tymUYZZBOeozP+BsGKbomLkL7C+ItTvaM
FAIfF93VekLJuw8l5dbQV3VXE5SgLl6QJM9GKWAkIkqGKPe7jo5lYiLJtlWt6IhuGqBdQzjpHO3Q33OGEj1rTra
MA== in file",
      "geo_breach_errors": [],
      "time_frequency_errors": [ ],
      "entry_type_errors": [],
      "time_breach_errors": [
         {
            "row": 0,
            "error_message": "logged timestamp 2021-07-08 11:13:18+00:00 is outside time limit 2021-07-11
13:54:45.648191+00:00 - 2021-07-11 17:54:45.648191+00:00"
         },
         {
            "row": 1,
            "error_message": "logged timestamp 2021-07-08 11:15:20+00:00 is outside time limit 2021-07-11
13:54:45.648191+00:00 - 2021-07-11 17:54:45.648191+00:00"
         }
      ]
```

```
  },
  "geo_breach_log_file": {
    "time_frequency_errors": [
      {
        "row": 1,
        "error_message": "logging frequency is above 1 seconds current 2021-07-08 11:31:19+00:00 -
previous 2021-07-08 11:29:41+00:00"
      },
      {
        "row": 16,
        "error_message": "logging frequency is above 1 seconds current 2021-07-08 11:33:05+00:00 -
previous 2021-07-08 11:31:33+00:00"
      }
    ],
    "entry_type_errors": [],
    "time_breach_errors": [
      {
        "row": 16,
        "error_message": "logged timestamp 2021-07-08 11:33:05+00:00 is outside time limit 2021-07-11
13:54:45.648191+00:00 - 2021-07-11 17:54:45.648191+00:00 for entry type LAND/DISARM"
      }
    ],
    "geo_breach_errors": []
  },
  "time_breach_log_file": {
    "time_frequency_errors": [
      {
        "row": 1,
        "error_message": "logging frequency is above 1 seconds current 2021-07-08 11:45:40+00:00 -
previous 2021-07-08 11:39:46+00:00"
      },
    ],
    "entry_type_errors": [],
    "time_breach_errors": [
      {
        "row": 0,
        "error_message": "logged timestamp 2021-07-08 11:39:46+00:00 is outside time limit 2021-07-11
13:54:45.648191+00:00 - 2021-07-11 17:54:45.648191+00:00 for entry type TAKEOFF/ARM"
      }
    ],
    "geo_breach_errors": []
777   }
}
```

6.8. Queries

In case of any doubts and queries please email to csrpas@qcin.org. To help us respond to your queries please maintain the below format.

6.9. Subject:
NPNT Test | Test Case Id | (Issue in 2-3 words)

6.10. Share data in email:

6.10.1. Make/ model/ UUID of drone

6.11. Attach data on a zip folder named <Test case ID.zip>:

6.11.1. Drone public key

6.11.2. Location kml file

6.11.3. Generated permission artifacts

6.11.4. Generated Flight logs

6.11.5. Generated report

6.11.6. Generated error log

Let us be compliant! We wish you a happy flying experience!