SECURE Challenge

When the team scans the Arrival QR code, the MS instructs the Team to proceed with receiving and analyzing the signals, and determining the 4 Lock Digits. The Team scans the Open QR code. The MS tells the station to enable the photodetector. When the photodetector is enabled, the Team instructs their BRATA to send the Lock Digits as a pulse-encoded sequence via the Android device's LED flash. The station receives the sequence, and determines success or failure.

All messages to the server have the following header field values, unless specified otherwise:

    Content-type: application/json
    Accept: application/json

All content is exchanged in JSON format, unless specified otherwise.

Team | **BRATA** | HTTP Request | **Master Server** | **Pi**

Team → BRATA: Scan Arrival QR Code

HTTP Request:
```
POST start_challenge/<station_id>
{
    "team_passcode":  "<passcode>"
}
```

HTTP Response:
```
{
    "message":  "Attach the mic cord and determine the 4 Lock Digits,
    then scan the Open QR Code"
}
```

BRATA → Team: Return message to Team

HTTP Request:
```
POST start_challenge
{
    "secure_tone_Pattern":  [d, d, d, d, d, d, d, d],
}
```

Turn OFF Red LED
Turn ON Yellow LED

HTTP Response

Team → BRATA: Enable BRATA to Capture Tones

tone patterns

Team + BRATA receives and interprets tones

Generate tones

BRATA → Team: Capture Complete

Team → BRATA: Scan Open QR Code

```
POST open/<station_id>
{
    "team_passcode":  "<passcode>"
}
```

```
{
    "message":  "Photodetector enabled: send Lock Digit pulses now!"
}
```

BRATA → Team: Return message to Team

```
POST open
{
}
```

Turn OFF Yellow LED
Turn ON Green LED
Enter photodetector input mode

Team → BRATA: Enable BRATA to Send Pulses

Team + BRATA sends pulse pattern

light pulse pattern

Receive pulse patterns

```
POST submit/<station_id>
{
    "candidate_answer":  [d, d, d, d],
    "is_correct":  "False",
    "fail_message":  "Incorrect answer provided.
                       <reason for failure>"
}
```

Send pass/fail result to MS

<reason for failure>:
Stop bit timeout
Incorrect code
Start timeout
...
(make them user-friendly)

```
{
    "challenge_complete":  "False"
}
```

Team will try again

Team submits result

Team → BRATA: Scan Secure QR Code

```
POST secure/<station_id>
{
    "team_passcode":  "<passcode>"
}
```

```
{
    "message":  "Correct! Go to the next Challenge!"
}
    or
{
    "message":  "Try again by listening to tones again (optional) and
    scanning Open QR Code to re-enable photodetector"
}
    or
{
    "message":  "Failed! Go to the next Challenge!"
}
```

BRATA → Team: Return message to Team

Try again

Team follows instructions