Validate IMEI/ESN API and Responses

URL : `https://www.stage.textnow.com/api/wireless/byod/v2/validate`

Method : `POST`

Body

```
{
    "body": "{"device_id": "[Number]"}"
}
```

Client side validation

Check for a valid IMEI/ESN before submitting.

```
export const validation = (values) => {
    const regexESNHex = /^[0-9a-fA-F]{8}$/; // Hex ESN (eight digit mixed case hex)
    const regexESNDec = /^[0-9]{11}$/; // Decimal ESN candidate (eleven decimal digits further validation is possible)
    const regexIMEINoCheckDec = /^[0-9]{14}$/; // Decimal IMEI with no check digit (fourteen digit decimal)
    const regexMEIDNoCheckHex = /^[0-9a-fA-F]{14}$/; // Hex MEID with no check digit (fourteen digit mixed case hex)
    const regexIMEIWithCheckDec = /^[0-9]{15}$/; // Decimal IMEI with check digit (fifteen digit decimal)
    const regexMEIDWithCheckHex = /^[0-9a-fA-F]{15}$/; // Hex MEID with with check digit (fifteen digit mixed case hex)
    const regexMEIDDec = /^[0-9]{18}$/; // Decimal MEID (eighteen digit decimal)

    if (values) {
        const cleanValues = values.replace(/[\s-]+/g, '');
        if (
            !(
                regexESNHex.test(cleanValues) ||
                regexESNDec.test(cleanValues) ||
                regexIMEINoCheckDec.test(cleanValues) ||
                regexMEIDNoCheckHex.test(cleanValues) ||
                regexIMEIWithCheckDec.test(cleanValues) ||
                regexMEIDWithCheckHex.test(cleanValues) ||
                regexMEIDDec.test(cleanValues)
            )
        ) {
            return 'Invalid IMEI/MEID/ESN';
        }
    } else {
        return 'You must enter an IMEI/MEID/ESN';
    }
    return '';
};
```

Success Response

Condition : If the device is valid you will get a 200 response. This is the response you will get for both incompatible and compatible devices.

Code : `200`

Content example

```
{
  "carrier": "CARRIER_TMOBILE",
  "sim_product_id": 392, // The product ID to link to in the store.
  "supported_plans": [],
  "supports_voice": false,
  "validation_result": "VALIDATION_SIM_REQUIRED"
}
```

Possible `validation_result` values:

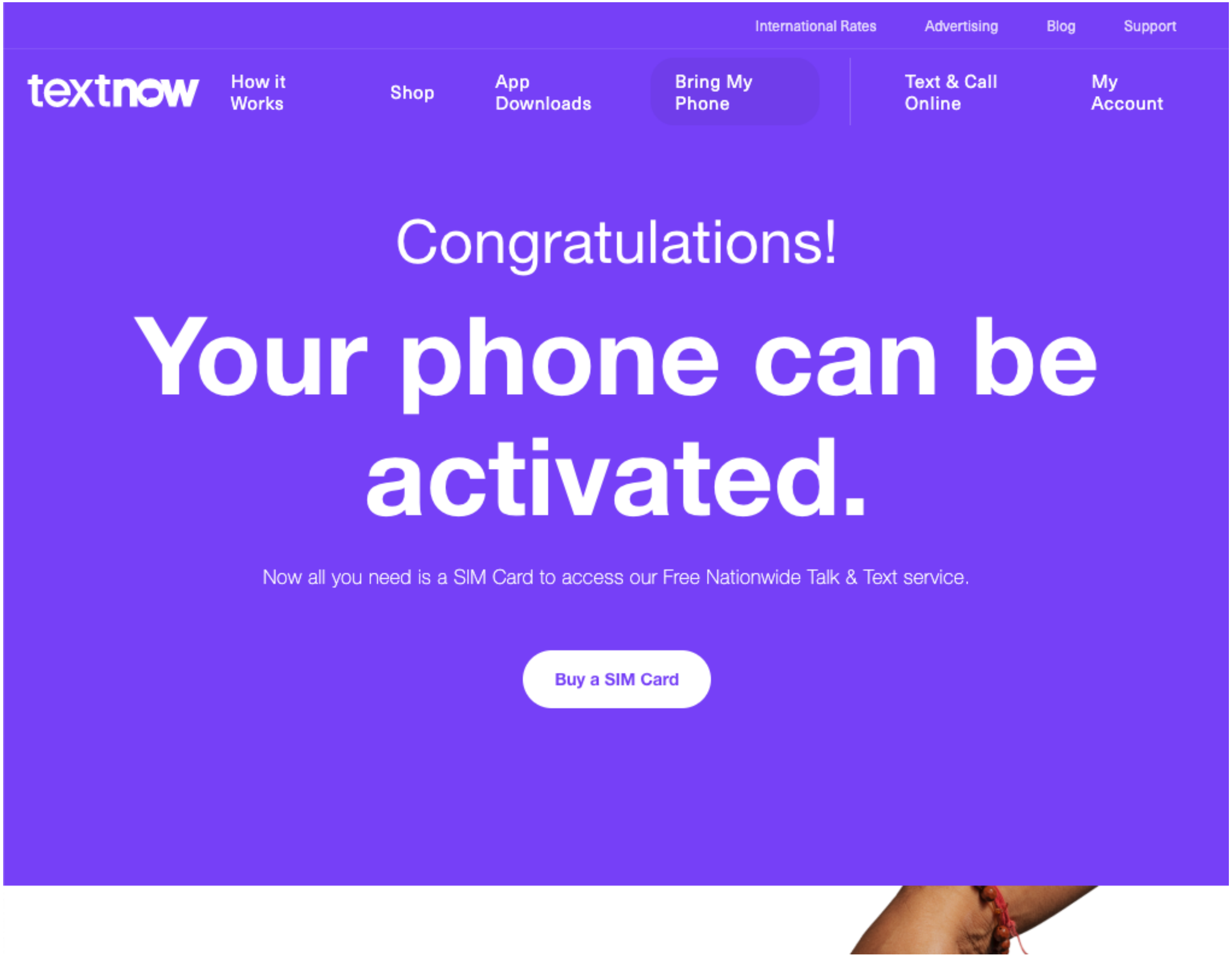
- `VALIDATION_SIM_REQUIRED`: Displays compatible screen.
- `VALIDATION_CANNOT_ACTIVATE`: Displays incompatible screen.

Buy a SIM URL:

- wireless/checkout?plan_id=0&product_id=392

Compatible screen

This is what the user will currently see when their device is compatible.



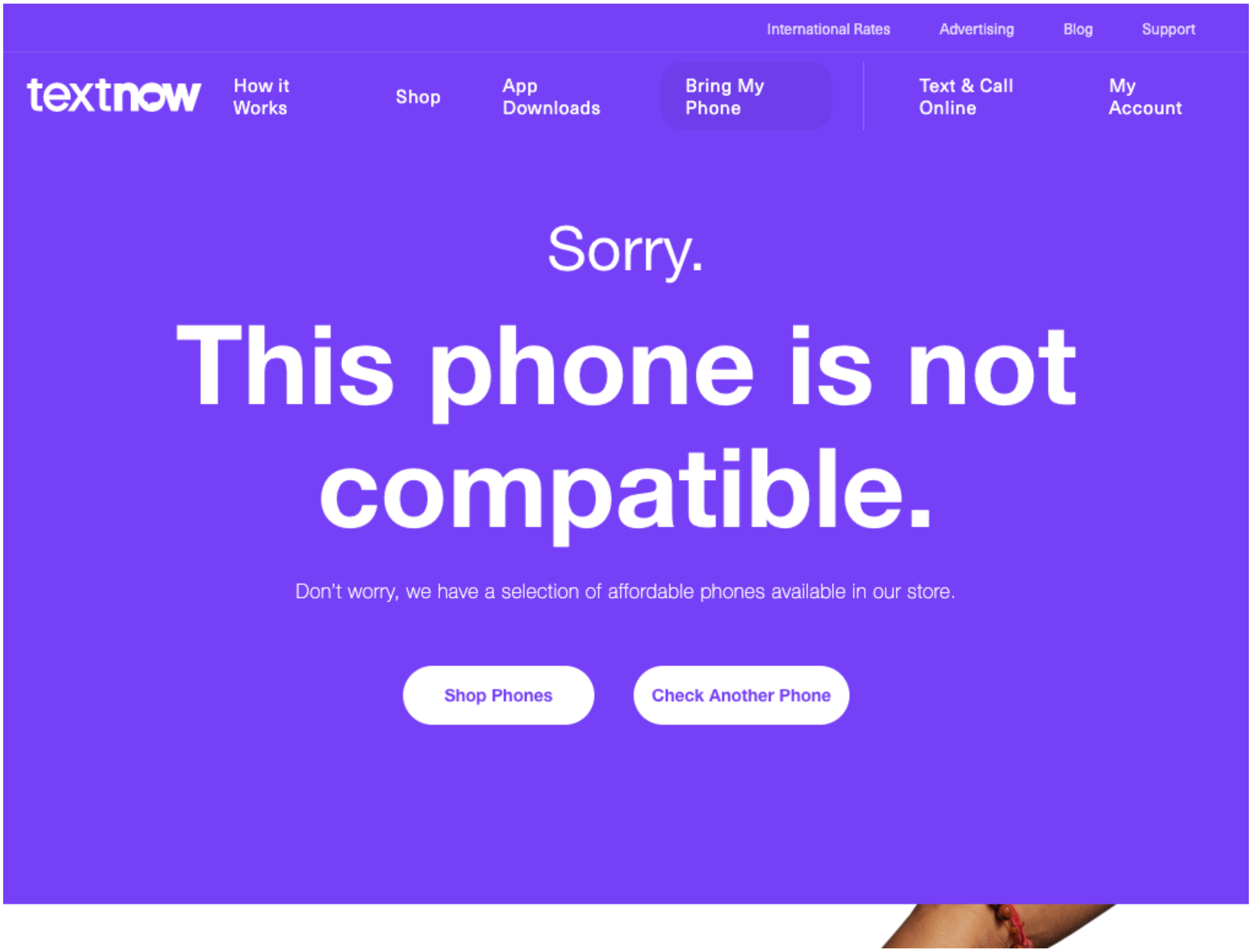
Congratulations!

Your phone can be activated.

Now all you need is a SIM Card to access our Free Nationwide Talk & Text service.

Incompatible screen

This is what the user will currently see when their device is incompatible.



Sorry.

This phone is not compatible.

Don't worry, we have a selection of affordable phones available in our store.

Error Response: 400

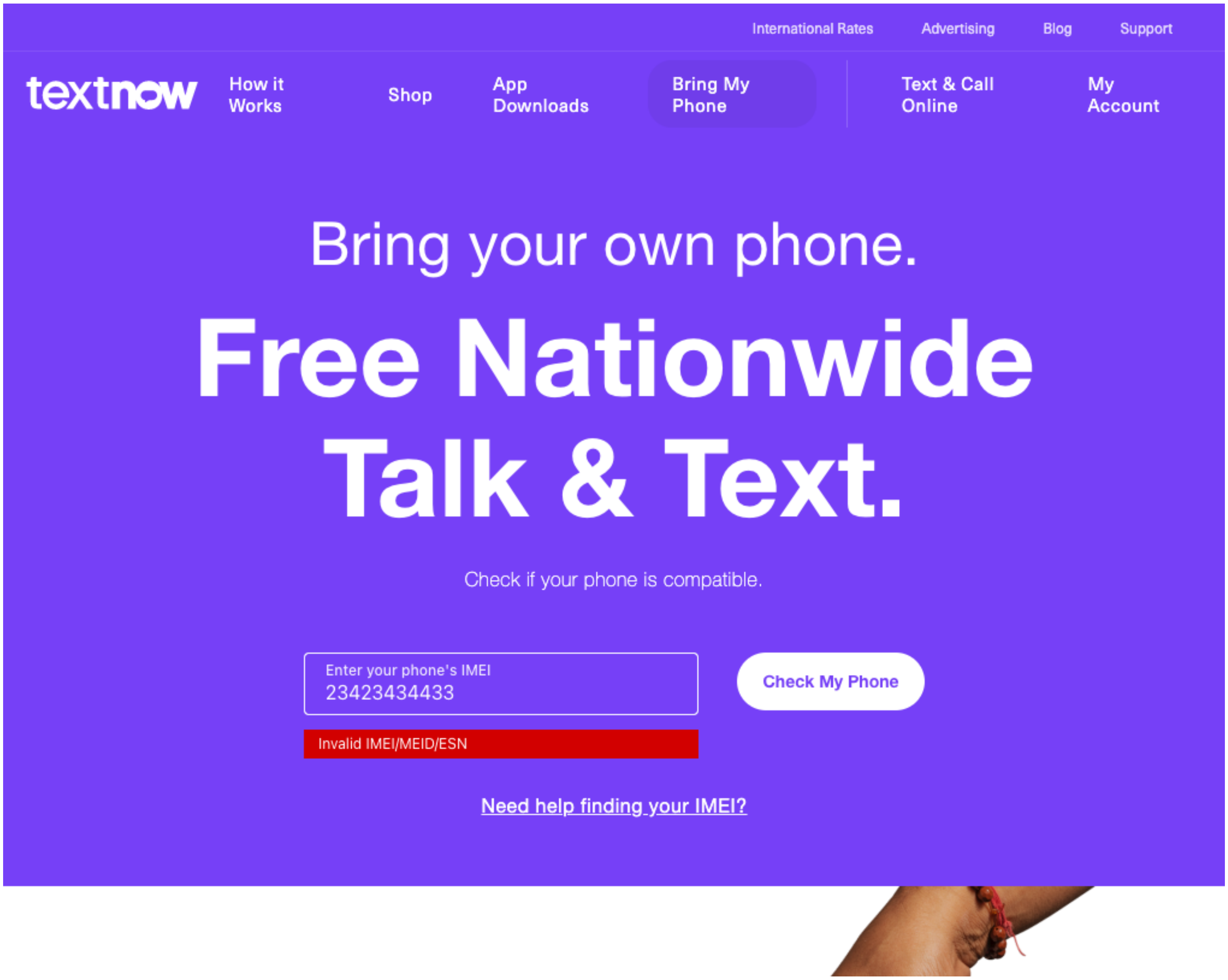
Condition : If IMEI/ESN is invalid.

Code : `400`

Content example

```
{
  "error_code": "DEVICE_ID_INVALID",
  "result": null
}
```

Error Screen



Bring your own phone.

Free Nationwide Talk & Text.

Check if your phone is compatible.

Enter your phone's IMEI
23423434433

Invalid IMEI/MEID/ESN

Error Response: 500

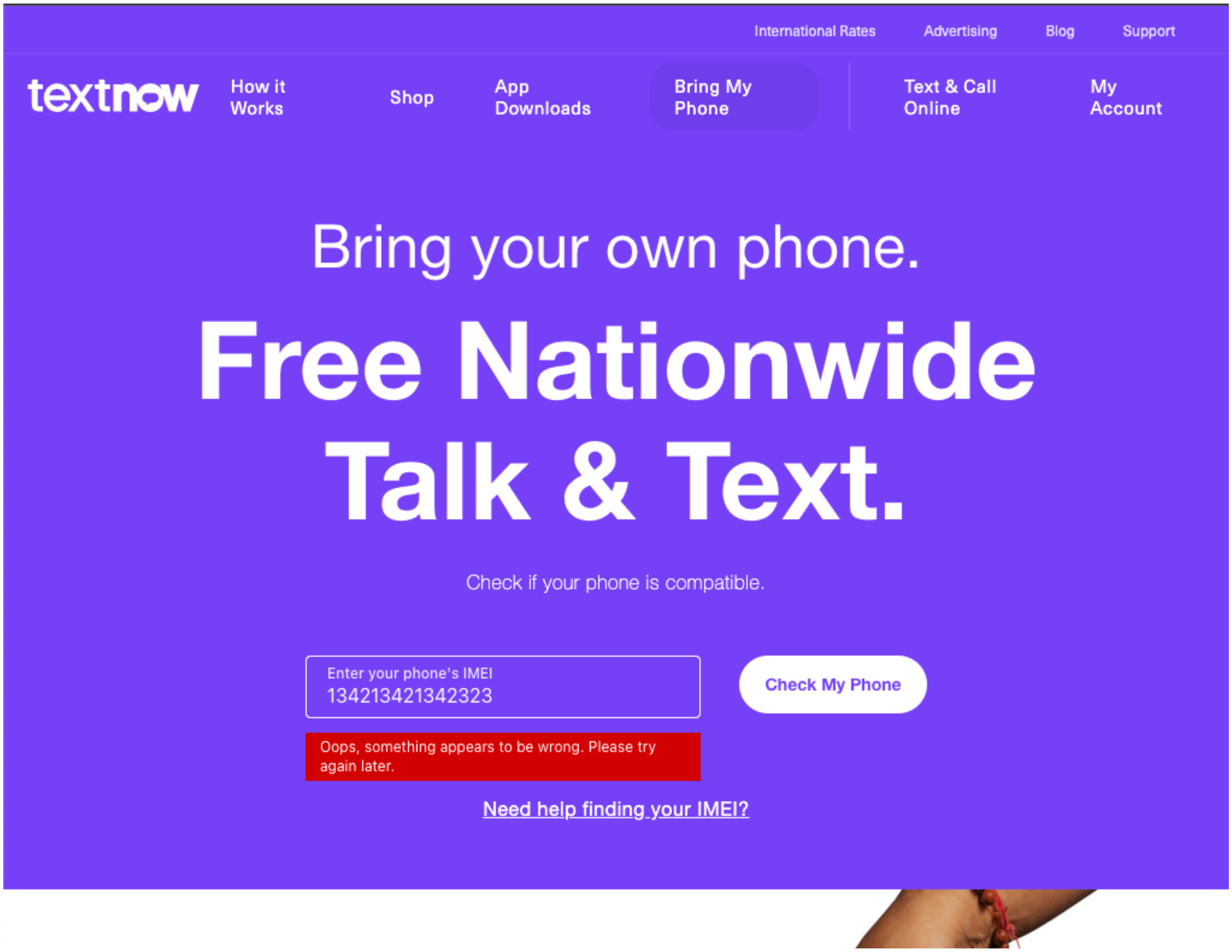
Condition : Server error. Sometimes things don't work. We currently handle this as well.

Code : `500`

Content example

```
{
  "error_code": "INTERNAL_FAILURE"
  "result": null
}
```

Error Screen



Bring your own phone.

Free Nationwide Talk & Text.

Check if your phone is compatible.

Enter your phone's IMEI
134213421342323

Oops, something appears to be wrong. Please try again later.

Cypress integration tests

We have a test suite that we can run against the device checker page. It can be set up as a stand-alone suite and changed to check the new user-facing copy. It can also be run against any test URL.

It can easily be modified to check any new copy being shown to users.

Mock device IDs

Mock Devices