

# CONTOH IMPLEMENTASI HOST TO HOST

## SMART BILLING BPI (BSI PEMBAYARAN INSTITUSI)



Version 2.10

## KEBUTUHAN APLIKASI:

1. Apache 2.4 dengan PHP 7.0+
2. Modul PHP MySQL, PHP JSON
3. Webserver yang dapat diakses melalui IP Publik
4. Mengizinkan akses pemanggilan dari IP 202.74.236.178 dan 103.219.249.4 ke webservice Anda

## CARA INSTALASI CEPAT:

1. Buat sebuah database dengan nama "db\_perusahaan"
2. Beri akses user "root" dengan password "root" untuk mengakses "db\_perusahaan"
3. Import file "tagihan\_pembayaran.sql" ke dalam database "db\_perusahaan"
4. Letakkan file **config.php**, **inquiry.php**, **payment.php**, dan **reversal.php** pada sebuah direktori yang dapat diakses oleh publik
5. Ubah isi file **config.php**:

```
$biller_name      = 'PERUSAHAAN XYZ';  
$secret_key      = 'SECRET-KEY';  
$mysql_host      = 'localhost:3306';  
$mysql_username  = 'root';  
$mysql_password  = 'root';  
$mysql_dbname    = 'db_perusahaan';
```

6. Tentukan sendiri **SECRET-KEY** untuk konfigurasi di sistem SmartBilling BPI
7. Pastikan berkas **inquiry.php**, **payment.php**, dan **reversal.php** diletakkan di suatu webserver yang dapat diakses melalui jaringan internet menggunakan browser.

Contoh URL Host-to-Host:

- **inquiry.php** dapat diakses pada **URL Host to Host Inquiry:**  
<https://server-saya.com/ws/inquiry.php>
- **payment.php** dapat diakses pada **URL Host to Host Payment:**  
<https://server-saya.com/ws/payment.php>
- **reversal.php** dapat diakses pada **URL Host to Host Reversal:**  
<https://server-saya.com/ws/reversal.php>

Apabila webservice diakses menggunakan browser, akan didapatkan respon sbb:

```
{"rc": "ERR-PARSING-MESSAGE", "msg": "Invalid Message Format"}
```



# KONFIGURASI SANDBOX:

1. Dapatkan akses username & password Institusi Anda untuk masuk ke **SANDBOX BPI** (<https://sandbox.bpi.co.id/>). Apabila belum mendapatkan akses, dapat gunakan username: **demo@bpi.co.id**, password **DemoBPI2021**
2. Masuk ke menu Konfigurasi -> Konfigurasi Umum

The screenshot shows the 'Konfigurasi Institusi' page for '1111 - DEMO H2H'. The interface is divided into several sections:

- Mode Transaksi:** Includes 'Permisian tagihan untuk setiap transaksi' (Setiap tagihan merupakan antrian yang diurutkan berdasarkan waktu jatuh te...), 'Tampilan Nominal Pada Field Nama' (Nominal Bayar & Tagihan Sebelah Kanan (cth: BUDIYANTO B-7000 T:10000)), 'Format Tampilan Pada Field Keterangan (Punya di Channel B2C)' (Format Keterangan Standard (cth: FAKULTAS ABC)), and 'Skenario Alur Transaksi SmartBilling/Host (Biller)' (Inquiry & Payment response dari Biller (Alur Normal)).
- Notifikasi Email:** Includes 'Notifikasi ke Pelanggan' (checked), 'Notifikasi ke Institusi' (checked), and 'Email Tujuan Notifikasi Institusi' (kahfi.cyber@gmail.com).
- Notifikasi WhatsApp:** Includes 'Notifikasi ke Pelanggan' (unchecked), 'Notifikasi ke Institusi' (unchecked), and 'Tujuan Nomor WhatsApp Institusi'.
- Notifikasi HTTP:** Includes 'Status & Spesifikasi' (Non Aktif), 'Token Notifikasi HTTP (Secret Key)' (\*\*\*\*\*), and 'URL Notifikasi Pembayaran'.
- Host to Host:** Includes 'Status & Spesifikasi' (Non Aktif), 'Token Host to Host (Secret Key)' (\*\*\*\*\*), and 'URL Host to Host Inquiry', 'URL Host to Host Payment', 'URL Host to Host Advice (Optional)', and 'URL Host to Host Reversal (Optional)'. The 'Token Host to Host (Secret Key)' field is highlighted with a red circle.
- Rekonsiliasi:** Includes 'Kirim Berkes Rekonsiliasi ke Email Institusi' (unchecked), 'Email Tujuan Rekonsiliasi' (support@yacamt.com), 'Panggil URL Webservice Custom Report Institusi' (unchecked), 'Token Custom Report', and 'URL Webservice Custom Report Rekonsiliasi'.

Buttons for 'ULANGI' and 'SUBMIT' are visible at the bottom.

- Masukkan **Token Host to Host**  
Cth: SECRET-KEY
- Masukkan **URL Host to Host Inquiry**  
cth: <https://server-saya.com/ws/inquiry.php>
- Masukkan **URL Host to Host Payment**  
cth: <https://server-saya.com/ws/payment.php>
- Masukkan **URL Host to Host Reversal**  
cth: <https://server-saya.com/ws/reversal.php>
- Checklist **Host to Host (S&K Berlaku)**
- Klik Submit

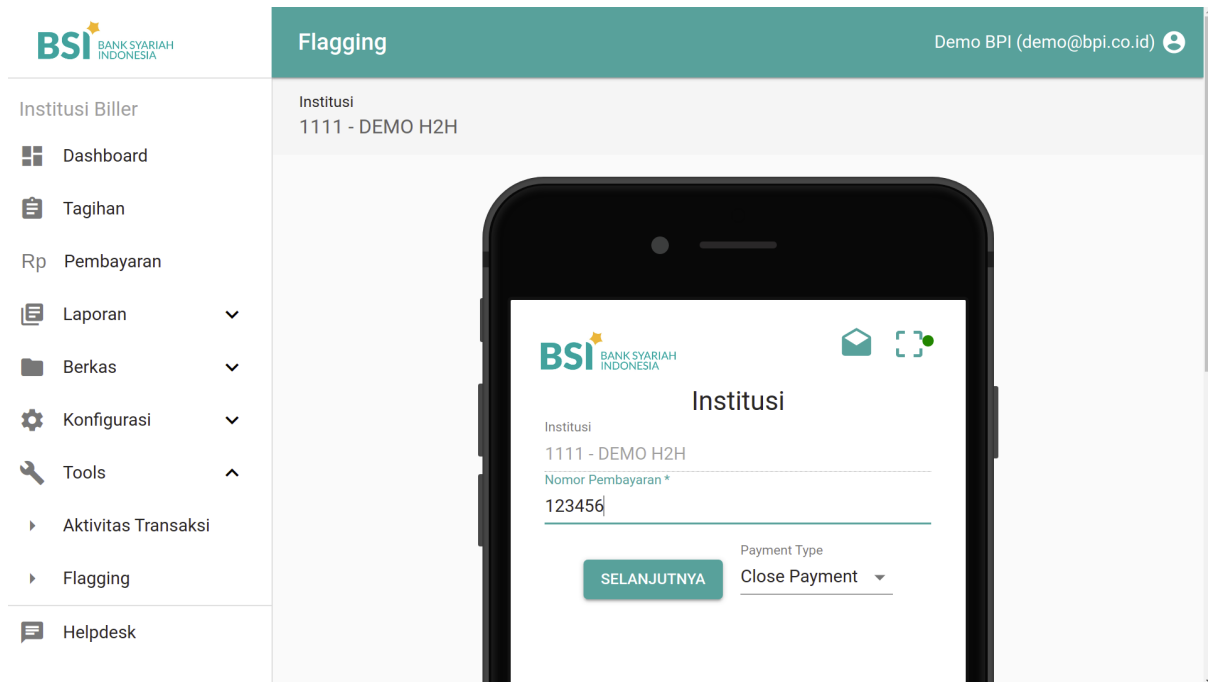
The screenshot shows the 'Konfigurasi Institusi' page for '0099 - UJICOPA WEBSERVICE'. The interface is similar to the previous one but with different values:

- Mode Transaksi:** 'Format Keterangan Standard (cth: FAKULTAS ABC)' and 'Skenario Alur Transaksi SmartBilling/Host (Biller)' (Inquiry & Payment response dari Biller (Alur Normal)).
- Notifikasi Email:** 'Notifikasi ke Pelanggan' (checked), 'Notifikasi ke Institusi' (unchecked), 'Email Tujuan Notifikasi Institusi'.
- Notifikasi WhatsApp:** 'Notifikasi ke Pelanggan' (unchecked), 'Notifikasi ke Institusi' (unchecked), 'Tujuan Nomor WhatsApp Institusi'.
- Notifikasi HTTP:** 'Status & Spesifikasi' (Non Aktif), 'Token Notifikasi HTTP (Secret Key)' (\*\*\*\*\*), 'URL Notifikasi Pembayaran'.
- Host to Host:** 'Status & Spesifikasi' (Standard BSI - Host-to-Host (Inquiry, Payment, Advice)), 'Token Host to Host (Secret Key)' (SECRETKEY), 'URL Host to Host Inquiry' (<https://webservice.tes.ac.id/bsi/inquiry.php>), 'URL Host to Host Payment' (<https://webservice.tes.ac.id/bsi/payment.php>), 'URL Host to Host Advice (Optional)', and 'URL Host to Host Reversal (Optional)'. The 'Token Host to Host (Secret Key)' field is highlighted with a red circle.
- Rekonsiliasi:** 'Kirim Berkes Rekonsiliasi ke Email Institusi' (unchecked), 'Email Tujuan Rekonsiliasi', 'Panggil URL Webservice Custom Report Institusi' (unchecked), 'Token Custom Report', and 'URL Webservice Custom Report Rekonsiliasi'.

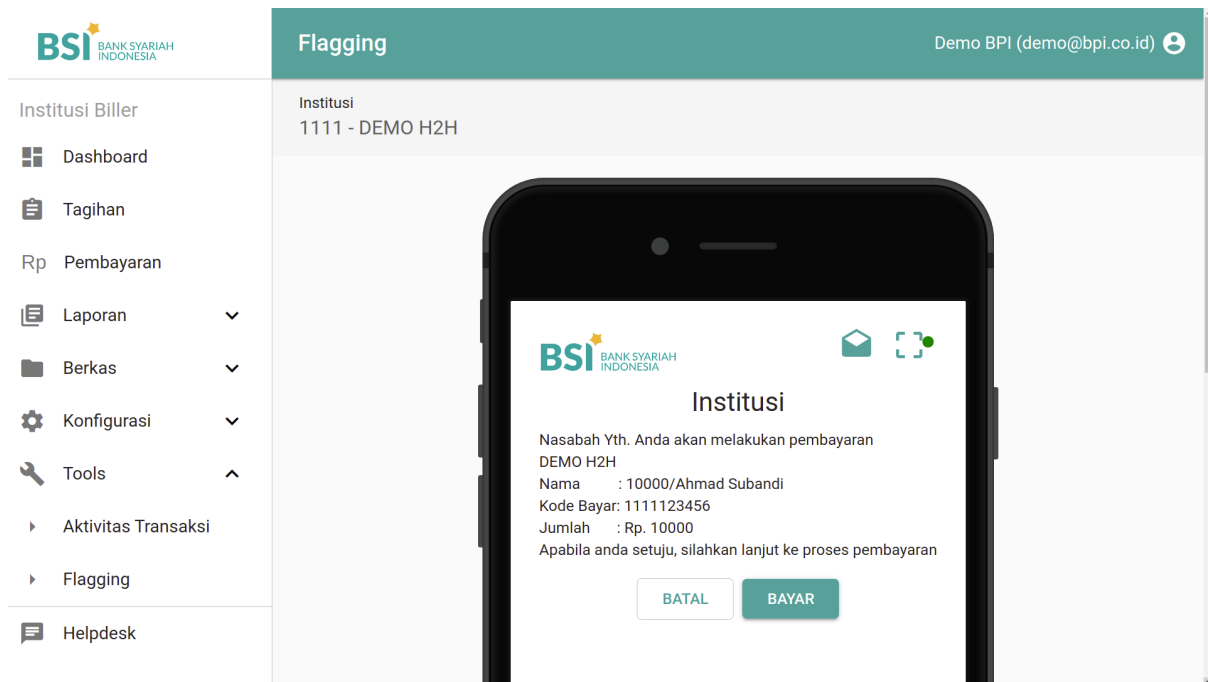
Buttons for 'ULANGI' and 'SUBMIT' are visible at the bottom.

# UJICOBA TRANSAKSI:

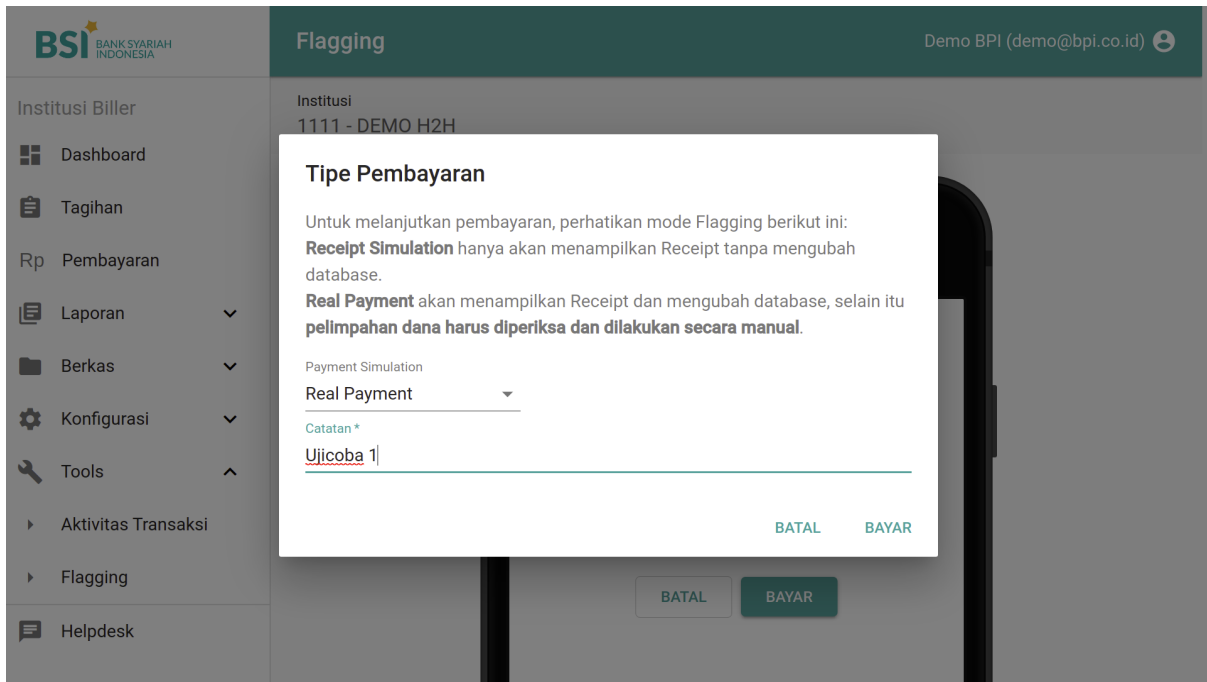
## 1. Masuk ke menu Tools -> Flagging



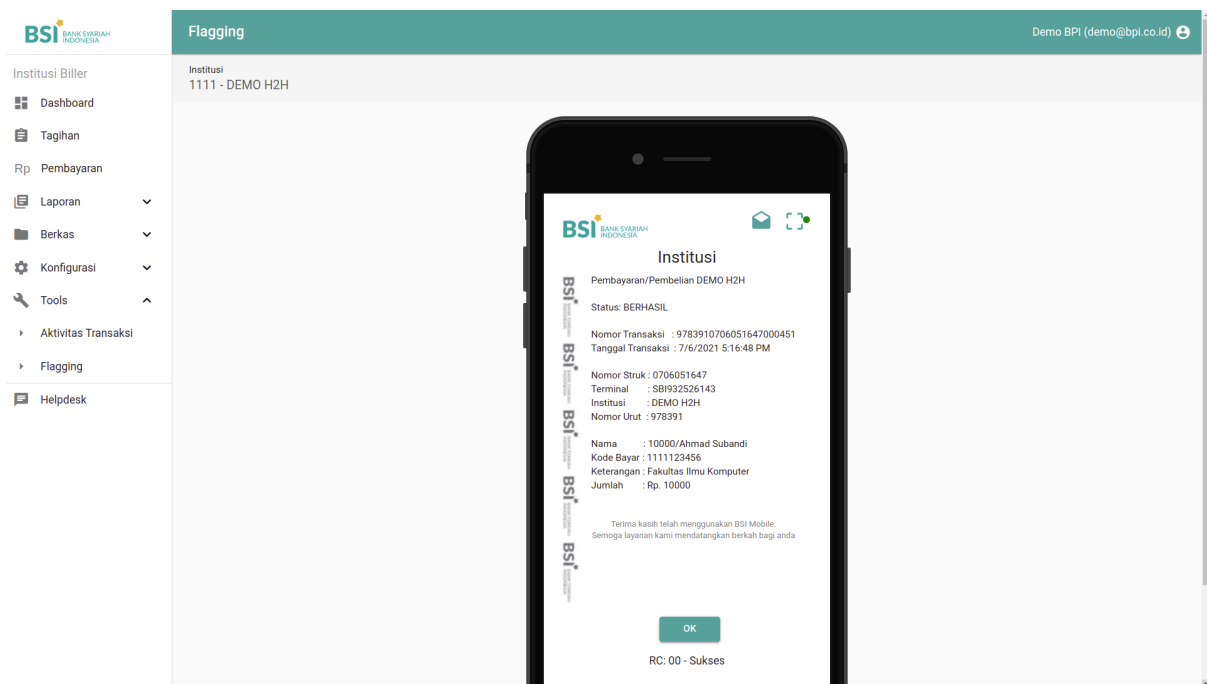
## 2. Lakukan ujicoba INQUIRY dengan memasukkan nomor Pembayaran (cth: 123456)



### 3. Lakukan ujicoba PAYMENT dengan klik tombol **BAYAR**, kemudian pilih **Real Payment**



### 4. Klik tombol BAYAR untuk melihat hasil transaksi



## TROUBLESHOOT:

Apabila Anda mendapatkan pesan error, atau tidak terjadi skenario ujicoba yang diinginkan, maka yang dapat Anda lakukan adalah:

- Periksa log file pada webserver Anda, pastikan bahwa tidak ada error pada koneksi Database, atau adanya syntaks error pada source code **config.php**, **inquiry.php**, **payment.php**, dan **reversal.php**.
- Periksa direktori logs dan baca debug file yang dihasilkan oleh webservice. Pastikan pula direktori logs sudah dibuat dan dapat ditulis oleh user yang menjalankan Apache / PHP.
- Pastikan **URL Host to Host Inquiry, Payment, dan Reversal** dapat diakses dari jaringan Internet

## config.php

```
<?php

$billier_name      = 'PERUSAHAAN XYZ';      // UBAH VARIABEL INI
$secret_key        = 'SECRET-KEY';         // UBAH VARIABEL INI

$mysql_host        = 'localhost:3306';     // UBAH VARIABEL INI
$mysql_username    = 'root';               // UBAH VARIABEL INI
$mysql_password    = 'root';               // UBAH VARIABEL INI
$mysql_dbname      = 'db_perusahaan';     // UBAH VARIABEL INI

$allowed_collecting_agents = array('BSM');
$allowed_channels      = array('TELLER', 'IBANK', 'ATM', 'MBANK', 'FLAGGING');
$log_directory         = './logs/';        // Direktori ini harus bisa ditulis oleh Apache
/ PHP user

function debugLog($o) {
    // Fungsi ini untuk menulis seluruh log ke File

    $file_debug = $GLOBALS['log_directory'] . 'debug-h2h-' . date("Y-m-d") . '.log.txt';
    ob_start();
    var_dump(date("Y-m-d H:i:s"));
    var_dump($o);
    $c = ob_get_contents();
    ob_end_clean();

    $f = fopen($file_debug, "a");
    fputs($f, "$c\n");
    fflush($f);
    fclose($f);
}

debugLog('REQUEST: ');
$request = file_get_contents('php://input');
debugLog($request);
$_JSON = json_decode($request, true);
?>
```

## inquiry.php

```
<?php
require 'config.php';

debugLog($_JSON);

// PARAMETER DI BAWAH INI ADALAH VARIABEL YANG DITERIMA DARI BSI
$kodeBank          = $_JSON['kodeBank'];
$kodeChannel       = $_JSON['kodeChannel'];
$kodeBiller        = $_JSON['kodeBiller'];
$kodeTerminal      = $_JSON['kodeTerminal'];
$nomorPembayaran   = $_JSON['nomorPembayaran'];
$tanggalTransaksi  = $_JSON['tanggalTransaksi'];
$idTransaksi       = $_JSON['idTransaksi'];
$totalNominalInquiry = $_JSON['totalNominalInquiry'];

// PERIKSA APAKAH SELURUH PARAMETER SUDAH LENGKAP
if (empty($kodeBank) || empty($kodeChannel) || empty($kodeTerminal) ||
empty($nomorPembayaran) || empty($tanggalTransaksi) || empty($idTransaksi)) {
    $response = json_encode(array(
```

```

        'rc' => 'ERR-PARSING-MESSAGE',
        'msg' => 'Invalid Message Format'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH KODE BANK DIIZINKAN MENGAKSES WEBSERVICE INI
if (!in_array($kodeBank, $allowed_collecting_agents)) {
    $response = json_encode(array(
        'rc' => 'ERR-BANK-UNKNOWN',
        'msg' => 'Collecting agent is not allowed by '.$biller_name
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH KODE CHANNEL DIIZINKAN MENGAKSES WEBSERVICE INI
if (!in_array($kodeChannel, $allowed_channels)) {
    $response = json_encode(array(
        'rc' => 'ERR-CHANNEL-UNKNOWN',
        'msg' => 'Channel is not allowed by '.$biller_name
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH CHECKSUM VALID
if (shal($_JSON['nomorPembayaran'].$secret_key.$_JSON['tanggalTransaksi']) !=
$_JSON['checksum']) {
    $response = json_encode(array(
        'rc' => 'ERR-SECURE-HASH',
        'msg' => 'H2H Checksum is invalid'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

$conn = mysqli_init();
$conn->options(MYSQLI_OPT_CONNECT_TIMEOUT, 3);
$conn->options(MYSQLI_OPT_READ_TIMEOUT, 7);
$conn->real_connect($mysql_host, $mysql_username, $mysql_password, $mysql_dbname);

$sql = $conn->prepare("SELECT * FROM tagihan_pembayaran WHERE nomor_siswa = ? order by
tanggal_invoice desc limit 1");
$sql->bind_param('s', $nomorPembayaran);
$sql->execute();
$result_cek = $sql->get_result();
$data_cek_available = $result_cek->fetch_array(MYSQLI_ASSOC);
debugLog($data_cek_available);

if($data_cek_available['nama'] == '') {
    // APABILA NAMA TIDAK DITEMUKAN
    $response = json_encode(array(
        'rc' => 'ERR-NOT-FOUND',
        'msg' => 'Nomor Tidak Ditemukan'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

```

```

}

$sql = $conn->prepare("SELECT * from tagihan_pembayaran where nomor_siswa = ? AND
status_pembayaran is NULL order by tanggal_invoice desc limit 1");
$sql->bind_param('s', $nomorPembayaran);
$sql->execute();
$result_tagihan = $sql->get_result();
$data_tagihan = $result_tagihan->fetch_array(MYSQLI_ASSOC);
debugLog($data_tagihan);

if($data_tagihan['nama'] == '' ) {
    // APABILA tidak ada nama yang bisa diambil berarti semua sudah SUKSES / terbayar
    $response = json_encode(array(
        'rc' => 'ERR-ALREADY-PAID',
        'msg' => 'Sudah Terbayar'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    $conn->close();
    exit();
}

$nama = $data_tagihan['nama'];
$id_tagihan = $data_tagihan['id_invoice'];
$all_info = $data_tagihan['informasi'];
$info1 = substr($all_info, 0, 30);
$info2 = substr($all_info, 30, 30);
$arr_informasi = [
    ['label_key' => 'Info1', 'label_value' => $info1 ],
    ['label_key' => 'Info2', 'label_value' => $info2 ],
];
$nominalTagihan = intval($data_tagihan['nominal_tagihan']);
$arr_rincian = [
    [
        'kode_rincian' => 'TAGIHAN',
        'deskripsi' => 'TAGIHAN',
        'nominal' => $nominalTagihan
    ],
];

$data_inquiry = [
    'rc' => 'OK',
    'msg' => 'Inquiry Succeeded',
    'nomorPembayaran' => $nomorPembayaran,
    'idPelanggan' => $nomorPembayaran,
    'nama' => $nama,
    'totalNominal' => $nominalTagihan,
    'informasi' => $arr_informasi,
    'rincian' => $arr_rincian,
    'idTagihan' => $id_tagihan,
];

$response_inquiry = json_encode($data_inquiry);
debugLog('RESPONSE: ' . $response_inquiry);
header('Content-Type: application/json');
echo $response_inquiry;
exit();

?>

```

# payment.php

```
<?php
require 'config.php';

debugLog($_JSON);

// PARAMETER DI BAWAH INI ADALAH VARIABEL YANG DITERIMA DARI BSI
$kodeBank           = $_JSON['kodeBank'];
$kodeChannel        = $_JSON['kodeChannel'];
$kodeBiller         = $_JSON['kodeBiller'];
$kodeTerminal       = $_JSON['kodeTerminal'];
$nomorPembayaran    = $_JSON['nomorPembayaran'];
$idTagihan          = $_JSON['idTagihan'];
$tanggalTransaksi   = $_JSON['tanggalTransaksi'];
$idTransaksi        = $_JSON['idTransaksi'];
$totalNominal       = $_JSON['totalNominal'];
$nomorJurnalPembukuan = $_JSON['nomorJurnalPembukuan'];

// PERIKSA APAKAH SELURUH PARAMETER SUDAH LENGKAP
if (empty($kodeBank) || empty($kodeChannel) || empty($kodeTerminal) ||
    empty($nomorPembayaran) || empty($tanggalTransaksi) || empty($idTransaksi) ||
    empty($totalNominal) || empty($nomorJurnalPembukuan) ) {
    $response = json_encode(array(
        'rc' => 'ERR-PARSING-MESSAGE',
        'msg' => 'Invalid Message Format'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH KODE BANK DIIZINKAN MENGAkses WEBSERVICE INI
if (!in_array($kodeBank, $allowed_collecting_agents)) {
    $response = json_encode(array(
        'rc' => 'ERR-BANK-UNKNOWN',
        'msg' => 'Collecting agent is not allowed by '.$biller_name
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH KODE CHANNEL DIIZINKAN MENGAkses WEBSERVICE INI
if (!in_array($kodeChannel, $allowed_channels)) {
    $response = json_encode(array(
        'rc' => 'ERR-CHANNEL-UNKNOWN',
        'msg' => 'Channel is not allowed by '.$biller_name
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

// PERIKSA APAKAH CHECKSUM VALID
if
(shal($_JSON['nomorPembayaran'].$secret_key.$_JSON['tanggalTransaksi'].$totalNominal.$nomor
JurnalPembukuan) != $_JSON['checksum']) {
    $response = json_encode(array(
        'rc' => 'ERR-SECURE-HASH',
        'msg' => 'H2H Checksum is invalid'
    ));
    debugLog('RESPONSE: ' . $response);
}
```

```

        echo $response;
        exit();
    }

$conn = mysqli_init();
$conn->options(MYSQLI_OPT_CONNECT_TIMEOUT, 3);
$conn->options(MYSQLI_OPT_READ_TIMEOUT, 7);
$conn->real_connect($mysql_host, $mysql_username, $mysql_password, $mysql_dbname);

///// INQUIRY /////

$sql = $conn->prepare("SELECT * FROM tagihan_pembayaran WHERE nomor_siswa = ? order by
tanggal_invoice desc limit 1");
$sql->bind_param('s', $nomorPembayaran);
$sql->execute();
$result_cek = $sql->get_result();
$data_cek_available = $result_cek->fetch_array(MYSQLI_ASSOC);
debugLog($data_cek_available);

if($data_cek_available['nama'] == '') {
    // APABILA NAMA TIDAK DITEMUKAN
    $response = json_encode(array(
        'rc' => 'ERR-NOT-FOUND',
        'msg' => 'Nomor Tidak Ditemukan'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    exit();
}

$sql = $conn->prepare("SELECT * from tagihan_pembayaran where nomor_siswa = ? AND
status_pembayaran is NULL order by tanggal_invoice desc limit 1");
$sql->bind_param('s', $nomorPembayaran);
$sql->execute();
$result_tagihan = $sql->get_result();
$data_tagihan = $result_tagihan->fetch_array(MYSQLI_ASSOC);
debugLog($data_tagihan);

if($data_tagihan['nama'] == '' ) {
    // APABILA tidak ada nama yang bisa diambil berarti semua sudah SUKSES / terbayar
    $response = json_encode(array(
        'rc' => 'ERR-ALREADY-PAID',
        'msg' => 'Sudah Terbayar'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    $conn->close();
    exit();
}

$nama = $data_tagihan['nama'];
$id_tagihan = $data_tagihan['id_invoice'];
$all_info = $data_tagihan['informasi'];
$info1 = substr($all_info, 0, 30);
$info2 = substr($all_info, 30, 30);
$arr_informasi = [
    ['label_key' => 'Info1', 'label_value' => $info1 ],
    ['label_key' => 'Info2', 'label_value' => $info2 ],
];
$nominalTagihan = intval($data_tagihan['nominal_tagihan']);
$arr_rincian = [
    [
        'kode_rincian' => 'TAGIHAN',
        'deskripsi' => 'TAGIHAN',
        'nominal' => $nominalTagihan
    ]
]

```

```

    ],
];

$data_inquiry = [
    'rc'          => 'OK',
    'msg'         => 'Inquiry Succeeded',
    'nomorPembayaran' => $nomorPembayaran,
    'idPelanggan'  => $nomorPembayaran,
    'nama'         => $nama,
    'totalNominal' => $nominalTagihan,
    'informasi'    => $arr_informasi,
    'rincian'      => $arr_rincian,
    'idTagihan'   => $id_tagihan,
];

if($nominalTagihan != $totalNominal) {
    // APABILA [CLOSE PAYMENT]
    // MAKA NILAI TAGIHAN DI DALAM DATABASE HARUS SAMA DENGAN YANG DIBAYARKAN
    $response = json_encode(array(
        'rc' => 'ERR-PAYMENT-WRONG-AMOUNT',
        'msg' => 'Terdapat kesalahan nilai pembayaran ' . $totalNominal . ' tidak sama
dengan tagihan ' . $nominalTagihan
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    $conn->close();
    exit();
}

///// PAYMENT /////
debugLog("START PAYMENT");
$conn->begin_transaction();
try {
    $sql = $conn->prepare("UPDATE tagihan_pembayaran set
status_pembayaran=?,
nomor_jurnal_pembukuan=?,
waktu_transaksi=?,
channel_pembayaran=?
WHERE id_invoice = ?");

    $status = "SUKSES";
    $waktu_pembayaran = date("Y-m-d H:i:s");

    debugLog("XXX");
    $sql->bind_param('sssss',
        $status,
        $nomorJurnalPembukuan,
        $waktu_pembayaran,
        $kodeChannel,
        $id_tagihan
    );
    debugLog("YYY");
    $sql->execute();
    $conn->commit();
}
catch(Exception $e) {
    $mysqli->rollback();
    $response = json_encode(array(
        'rc' => 'ERR-DB',
        'msg' => 'Error saat Update Transaksi'
    ));
    debugLog('RESPONSE: ' . $response);
    echo $response;
    $conn->close();
}

```

```

        exit();
    }
    debugLog("END PAYMENT");

    $data_payment = $data_inquiry;
    $data_payment['msg'] = 'Payment Succeeded';
    debugLog($data_payment);

    $response_payment = json_encode($data_payment);
    debugLog('RESPONSE: ' . $response_payment);
    header('Content-Type: application/json');
    echo $response_payment;
    exit();

?>

```

## reversal.php

```

<?php
require 'config.php';

debugLog($_JSON);

$kodeBank           = $_JSON['kodeBank'];
$kodeChannel        = $_JSON['kodeChannel'];
$kodeTerminal       = $_JSON['kodeTerminal'];
$nomorPembayaran    = $_JSON['nomorPembayaran'];
$idTagihan          = $_JSON['idTagihan'];
$tanggalTransaksi   = $_JSON['tanggalTransaksi'];
$tanggalTransaksiAsal = $_JSON['tanggalTransaksiAsal'];
$idTransaksi        = $_JSON['idTransaksi'];
$totalNominal       = $_JSON['totalNominal'];
$nomorJurnalPembukuan = $_JSON['nomorJurnalPembukuan'];

// MENOLAK REVERSAL
$response = json_encode(array(
    'rc' => 'ERR-REVERSAL-DENIED',
    'msg' => 'Reversal ditolak. Pembayaran sudah update ke DB di '.$biller_name
));

debugLog('RESPONSE: ' . $response);
echo $response;

```

## MySQL Dump (Sample Tabel)

```

-- phpMyAdmin SQL Dump
-- version 4.9.7
-- https://www.phpmyadmin.net/
--
-- Host: localhost:3306
-- Generation Time: Jul 06, 2021 at 12:19 PM
-- Server version: 5.7.34
-- PHP Version: 7.3.28

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

```

```

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `bsi`
--

-----

--
-- Table structure for table `tagihan_pembayaran`
--

CREATE TABLE `tagihan_pembayaran` (
  `id_invoice` varchar(100) NOT NULL,
  `tanggal_invoice` date NOT NULL,
  `nomor_siswa` varchar(12) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `nominal_tagihan` int(11) NOT NULL,
  `informasi` varchar(60) NOT NULL,
  `nomor_jurnal_pembukuan` varchar(100) DEFAULT NULL,
  `waktu_transaksi` datetime DEFAULT NULL,
  `channel_pembayaran` varchar(20) DEFAULT NULL,
  `status_pembayaran` varchar(10) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `tagihan_pembayaran`
--

INSERT INTO `tagihan_pembayaran` (`id_invoice`, `tanggal_invoice`, `nomor_siswa`, `nama`,
`nominal_tagihan`, `informasi`, `nomor_jurnal_pembukuan`, `waktu_transaksi`,
`channel_pembayaran`, `status_pembayaran`) VALUES
('JUL/001', '2021-07-06', '123456', 'Ahmad Subandi', 10000, 'Fakultas Ilmu Komputer', NULL,
NULL, NULL, NULL),
('JUL/002', '2021-07-06', '876876', 'Yanto Susanto', 20000, 'Fakultas Ilmu Sosial', NULL,
NULL, NULL, NULL);

--
-- Indexes for dumped tables
--

--
-- Indexes for table `tagihan_pembayaran`
--
ALTER TABLE `tagihan_pembayaran`
  ADD PRIMARY KEY (`id_invoice`) USING BTREE;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```