

スマートフォン上のアプリケーションにおける利用者情報の
取扱いに係る技術的検証等の諸問題に係る実証調査研究

静的解析システムソースコード（フロントエンド処理）

平成 29 年 2 月 1 日

改訂履歴

版数	日付	変更理由	変更内容	変更者
1.0	H29/2/1	新規作成	1.0 版作成	

目次

1	はじめに	5
2	ソースファイル一覧	6
3	画面ソースコード	8
3.1	W0001 タイトル画面.....	8
3.2	W1001 Android アプリ解析依頼画面	11
3.3	W1002 Android アプリ解析依頼内容確認ダイアログ	19
3.4	W1003 Android アプリ解析依頼受付拒否ダイアログ	22
3.5	Android アプリ解析依頼画面用 JavaScript.....	24
3.6	CAPTCHA 画像生成	37
3.7	W1004 解析依頼完了画面.....	38
3.8	解析依頼完了画面用 JavaScript	41
3.9	W2001 統計情報画面.....	42
3.10	文言定義ファイル.....	46
4	Ajax リクエスト処理ソースコード	51
4.1	解析状況確認.....	51
4.1	apk ファイルアップロード	57

5	利用クラスソースコード	63
5.1	DB 接続クラス	63
5.2	セッションクラス	82
5.3	ファイルクラス	84
6	CSS ファイル	87

1 はじめに

本書は、平成 28 年度の総務省施策である「スマートフォン上のアプリケーションにおける利用者情報の取扱いに係る実証調査研究の請負」における静的解析システムのフロントエンド処理のソースコードである。

2 ソースファイル一覧

公開サーバー側システムのうち、画面を構成するソースファイルを「表 2-1 画面構成ソースファイル一覧」に示す。

画面 ID は、静的解析システム仕様書の「3.1 画面一覧」に示されたものである。

表 2-1 画面構成ソースファイル一覧

項番	名称	画面 ID	ソースコードファイル名	備考
1	タイトル	W0001	index.php	
2	Android アプリ解析依頼画面	W1001	apkform.php	
3	Android アプリ解析依頼内容確認ダイアログ	W1002	dialog.php	W1001 Android アプリ解析依頼画面 表示時に読み込まれる
4	Android アプリ解析依頼受付拒否ダイアログ	W1003	dialogReject.php	W1001 Android アプリ解析依頼画面 表示時に読み込まれる
5	Android アプリ解析画面用 JavaScript		js/pp.js	W1001 Android アプリ解析依頼画面 表示時に読み込まれる
6	CAPTCHA 画像生成		captcha/captcha.php	W1001 Android アプリ解析依頼画面 表示時に実行され画像を出力する。
7	解析依頼完了画面	W1004	result.php	
8	解析依頼完了画面用 JavaScript		js/ppResult.js	W1004 解析依頼完了画面 表示時に読み込まれる
9	統計情報画面	W2001	status.php	
10	文言定義ファイル		wording.php	画面で使用する文言の宣言ファイル（共通）

公開サーバー側システムで、画面からの Ajax リクエスト処理のためのソースファイルを「表 2-2 Ajax 呼び出されるソースファイル」に

示す。

表 2-2 Ajax 呼び出されるソースファイル

項番	名称	ソースコードファイル名	備考
1	解析状況確認	send1st.php	W1001 Android アプリ解析依頼画面から「依頼」ボタンをクリックした時に呼び出される
2	apk ファイルアップロード	apkupload.php	W1002 Android アプリ解析依頼内容確認ダイアログからアップロード時に呼び出される

公開サーバー側システムで利用されるクラスを定義しているソースファイルを「表 2-3 クラスを定義しているソースファイル」に示す。

表 2-3 クラスを定義しているソースファイル

項番	名称	ソースコードファイル名	備考
1	DB 接続クラス	ppDb.php	DB を扱うクラス
2	セッションクラス	ppSession.php	セッションを扱うクラス
3	ファイルクラス	ppFiles.php	ディレクトリ内のファイル一覧を取得するクラス

公開サーバー側システムで利用される CSS ソースファイルを以下の「表 2-4 CSS ソースファイル」に示す。

表 2-4 CSS ソースファイル

項番	名称	ソースコードファイル名	備考
1	CSS ファイル	css/pp.css	

3 画面ソースコード

3.1 W0001 タイトル画面

W0001 タイトル画面 のソースコードを以下に示す。

```
<?php
include_once('wording.php');
$title = $WORDING['WD-1001'];
include_once('header.php');

include_once('ppDb.php');

// DB 接続
$db = new ppDB();
if( NULL !== $db->getDbError() ){
    header('Location: dberror.php');
    exit;
}

// 解析状況を取得
$status = $db->getAnalysisStatus();

$db->close();
```



```
?>
<div class="container">
  <div class="title bg-primary">
    <h1><?php echo $WORDING['WD-1002']; ?></h1>
  </div>

  <div class="subsectionTitle">
    <h2><?php echo $WORDING['WD-1003']; ?></h2>
  </div>
  <div class="subsectionBody" id="topMessage">
<?php include_once('message.php'); ?>
  </div>

  <div class="subsectionTitle">
    <h2><?php echo $WORDING['WD-1004']; ?></h2>
  </div>
  <div class="subsectionBody" id="topInfo">
<?php include_once('info.php'); ?>
  </div>

  <div class="subsectionTitle">
    <h2><?php echo $WORDING['WD-1005']; ?></h2>
  </div>
</div>
```

```

<div class="subsectionBody">
  <ul id="titleLinks" class="list-unstyled">
    <li>
<?php if($status['accept']) { ?>
      <a href="apkform.php"><span class="glyphicon glyphicon-circle-arrow-right"></span> <?php echo $WORDING['WD-1006']; ?><
/a>
<?php }else{ ?>
      <span class="glyphicon glyphicon-circle-arrow-right"></span> <?php echo $WORDING['WD-1006']; ?>
<?php } ?>
      <dl class="dl-horizontal topblank8">
        <dt><?php echo $WORDING['WD-7001']; ?></dt>
        <dd>
<?php if($status['accept']) { ?>
          <?php echo $WORDING['WD-7004']; ?> <span><?php echo $status['count']; ?></span> <?php echo $WORDING['WD-7005
']; ?>
<?php }else{ ?>
          <?php echo $WORDING['WD-7006']; ?>
<?php } ?>
        </dd>
        <dt><?php echo $WORDING['WD-7007']; ?></dt>
        <dd><span><?php echo !empty($status['avgtime'])?$status['avgtime']:'--:--:--' ?></span></dd>
      </dl>
    </li>

```

```
<li><a href="status.php"><span class="glyphicon glyphicon-circle-arrow-right"></span> <?php echo $WORDING['WD-1007'];
?></a></li>
<li><a><span class="glyphicon glyphicon-circle-arrow-right"></span> <?php echo $WORDING['WD-1008']; ?></a></li>
</ul>
</div>
</div>

<?php
include_once(' footer.php ');
```

3.2 W1001 Android アプリ解析依頼画面

W1001 Android アプリ解析依頼画面のソースコードを以下に示す。

```
<?php
include_once(' wording.php ');
$title = $WORDING['WD-2001'];

include_once(' ppDb.php ');
/*
 * DB への接続
```

```

* 接続できなかった場合はエラー画面へ
*/
$db = new ppDB();
if( NULL !== $db->getDbError() ){
    header('Location: dberror.php');
    exit;
}

include_once('header.php');
?>
<div class="container">

    <div class="title bg-primary">
        <h1><?php echo $WORDING['WD-2001']; ?></h1>
        <div class="titleback"><a href="."/"><?php echo $WORDING['WD-1009']; ?></a></div>
    </div>
    <div class="titlebackClear"></div>

    <div class="sectionBody">
        <input type="hidden" id="bsKey" value="">
        <div class="input-group">
            <div class="form-group">
                <input type="file" id="apkFile" name="apkFile" class="form-control nodisplay" accept=".apk">

```

```

        <label for="apkFileCover"><?php echo $WORDING['WD-2003']; ?> <span class="require" id="warn-apkFileCover"><?php e
cho $WORDING['WD-2020']; ?></span></label>
        <span class="input-group-btn"><input type="text" id="apkFileCover" class="form-control" placeholder="<?php echo
$WORDING['WD-2004']; ?>">
        <button id="apkFileCoverBtn" type="button" class="btn btn-primary"><?php echo $WORDING['WD-2018']; ?></button></s
pan>
    </div>
</div>

<div class="form-group">
    <label for="appUser"><?php echo $WORDING['WD-2019']; ?> <span class="require" id="warn-appUser"><?php echo $WORDING['W
D-2020']; ?></span></label>
    <select id="appUser" class="form-control">
        <option value="0" selected="selected" class="placeholder"><?php echo $WORDING['WD-2021']; ?></option>
<?php foreach($db->getUserListAll() as $u ) { ?>
        <option value="<?php echo $u['id']; ?>"><?php echo $u['name']; ?></option>
<?php }?>
    </select>
</div>

<div class="form-group">
    <label for="appCount"><?php echo $WORDING['WD-2008']; ?> <span class="require" id="warn-appCount"><?php echo $WORDING
['WD-2020']; ?></span></label>

```

```

        <select id="appCount" class="form-control">
            <option value="0" selected="selected" class="placeholder"><?php echo $WORDING['WD-2022']; ?></option>
<?php foreach($COUNT as $ckey => $cval ) { ?>
            <option value="<?php echo $ckey; ?>"><?php echo $cval; ?></option>
<?php }?>
        </select>
    </div>

    <div class="form-group">
        <label for="appGenre"><?php echo $WORDING['WD-2009']; ?> <span class="require" id="warn-appGenre"><?php echo $WORDING
['WD-2020']; ?></span></label>
        <select id="appGenre" class="form-control">
            <option value="0" selected="selected" class="placeholder"><?php echo $WORDING['WD-2023']; ?></option>
<?php
foreach($db->getGenreListAll() as $g ) { ?>
            <option data-id="<?php echo $g['id']; ?>" value="<?php echo $g['id']; ?>"><?php echo $g['name']; ?></option>
<?php
}
?>
        </select>
    </div>
<?php
$i = 1;

```

```

foreach($db->getGenreListAll() as $g ){ ?>
    <div id="genre<?php echo sprintf('%02d', $g['id']); ?>" class="appGenreList">
        <table class="table">
<?php
    $columnGenre = 4; // 一列に並ぶカテゴリ数
    $j=0;
    foreach($g['subGenre'] as $sg) {
        if(0 == $j) {
?>
            <tr>
<?php
                }
                ?>
                <td><label class="checkbox-inline"><input type="checkbox" value="<?php echo $sg['id']; ?>"><?php echo $sg['name']; ?></label> </td>
<?php
                $j = ($j+1) % $columnGenre;
                if(0 == $j) {
?>
                    </tr>
<?php
                }
            }
        }
    }
}

```

```

    if(0 != $j) {
        for($k=$j; $columnGenre > $k; $k++){
?>
                <td></td>
<?php
        }
?>
                </tr>
<?php
    }
?>
        </table>
    </div>
<?php
}
?>

    <div class="form-group">
        <label for="appselection"><?php echo $WORDING['WD-2010']; ?> <span class="require" id="warn-mailaddress"><?php echo $W
WORDING['WD-2020']; ?></span></label>
        <input type="email" id="mailaddress" class="form-control" placeholder="<?php echo $WORDING['WD-2012']; ?>">
        <input type="email" id="mailaddressConfirm" class="form-control" placeholder="<?php echo $WORDING['WD-2013']; ?>">
        <div id="confirmMessage" class="text-warning"></div>

```



```
</div>

<div class="rule">
  <label><?php echo $WORDING['WD-2014']; ?></label>
  <div class="ruleText">
<?php include_once('rule.php'); ?>
  </div>
  <label class="checkbox-inline"><input type="checkbox" id="agree"><?php echo $WORDING['WD-2015']; ?> <span class="require" id="warn-agree"><?php echo $WORDING['WD-2020']; ?></span></label>
</div>

<div class="captcha">
  <div class="form-group form-inline">
    
    <label><?php echo $WORDING['WD-2016']; ?>
    <input type="text" id="captchaAns" class="form-control">
    <span class="require" id="warn-captchaAns"><?php echo $WORDING['WD-2020']; ?></span>
  </label>
</div>
</div>

<div class="btnSubmit">
```

```
        <button id="submitAnalyze" class="btn btn-primary" disabled><?php echo $WORDING['WD-2017']; ?></button>
    </div>
</div>
</div>

<div id="lock"></div>
<?php
include_once('dialog.php');
include_once('dialogReject.php');

$formJS = array_merge($formJS, array('js/pp.js'));
include_once('footer.php');

$db->close();
```

3.3 W1002 Android アプリ解析依頼内容確認ダイアログ

W1002 Android アプリ解析依頼確認ダイアログのソースコードを以下に示す。

```
<?php
?>
<div class="modal fade" id="kakunin" tabindex="-1" role="dialog" aria-labelledby="l">
  <div class="modal-dialog" role="document">
    <div class="modal-content" >
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>
        <h4 class="modal-title" id="kakuninModalLabel"><?php echo $WORDING['WD-2101']; ?></h4>
      </div>
      <div class="modal-body">
        <dl class="dl-horizontal">
          <dt><?php echo $WORDING['WD-2003']; ?> &nbsp;</dt>
          <dd id="confirm_apkFile" class="confirmText"></dd>
        </dl>
        <dl class="dl-horizontal">
          <dt><?php echo $WORDING['WD-2005']; ?> &nbsp;</dt>
          <dd id="confirm_analyzeType" class="confirmText"></dd>
        </dl>
        <dl class="dl-horizontal">
          <dt><?php echo $WORDING['WD-2019']; ?> &nbsp;</dt>
          <dd id="confirm_appUser" class="confirmText"></dd>
        </dl>
      </div>
    </div>
  </div>
</div>
```

```

<dt><?php echo $WORDING['WD-2008']; ?> &nbsp;</dt>
<dd id="confirm_appCount" class="confirmText"></dd>
<dt><?php echo $WORDING['WD-2009']; ?> &nbsp;</dt>
<dd id="confirm_appGenre" class="confirmText"></dd>
<dt><?php echo $WORDING['WD-2011']; ?> &nbsp;</dt>
<dd id="confirm_mailaddress" class="confirmText"></dd>
</dl>

<form method="POST" action="result.php" id="send2nd" class="nodisplay">
  <input type="hidden" id="bsMailaddress" name="bsMailaddress" value="">
  <input type="hidden" id="bsReqNo" name="bsReqNo" value="">
</form>
<span id="kakuninMsg"><?php echo $WORDING['WD-2102']; ?></span>
<span id="uploadMsg"><?php echo $WORDING['WD-2105']; ?></span>
</div>
<div class="modal-footer">
  <div id="kakuninButtons">
    <button type="button" class="btn btn-default" id="kakuninCancel" data-dismiss="modal"><?php echo $WORDING['WD
-2103']; ?></button>
    <button type="button" class="btn btn-primary" id="kakuninSubmit"><?php echo $WORDING['WD-2104']; ?></button>
  </div>
  <div id="uploadProgress" class="progress progress-striped">
    <div class="progress-bar progress-bar-info" id="upPgsBar"></div>

```

```
        </div>
      </div>
    </div>
  </div>
</div>
```

3.4 W1003 Android アプリ解析依頼受付拒否ダイアログ

W1003 Android アプリ受付拒否ダイアログのソースコードを以下に示す。

```
<?php
?>
<div class="modal fade" id="reject" tabindex="-1" role="dialog" aria-labelledby="l">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>
        <h4 class="modal-title" id="rejectModalLabel"><?php echo $WORDING['WD-3101']; ?></h4>
      </div>
      <div class="modal-body">
        <div class="modalText">
          <?php echo $WORDING['WD-3102']; ?><br>
          <span id="rejectMessage"></span>
        </div>
        <dl class="dl-horizontal">
          <dt><?php echo $WORDING['WD-7001']; ?> &nbsp;</dt>
          <dd><?php echo $WORDING['WD-7002']; ?> <span id="ccount">0</span> <?php echo $WORDING['WD-7003']; ?></dd>
          <dt><?php echo $WORDING['WD-7007']; ?> &nbsp;</dt>
        </dl>
      </div>
    </div>
  </div>
</div>
```

```
        <dd id="ctime">hh:mm:ss</dd>
    </dl>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-primary" id="rejectClose" data-dismiss="modal"><?php echo $WORDING['WD-3103
']; ?></button>
</div>
</div>
</div>
</div>
```

3.5 Android アプリ解析依頼画面用 JavaScript

Android アプリ解析依頼画面用の JavaScript のソースコードを以下に示す。

```
$(function () {  
  
    /*  
    * 必須入力項目のチェック  
    */  
    var checkSubmit = function() {  
        $disableBtn = false;  
  
        if( '' == $('#apkFileCover').val() ){  
            $disableBtn = true;  
            $('#warn-apkFileCover').show();  
        }else{  
            $('#warn-apkFileCover').hide();  
        }  
  
        if(( 0 == $('#appUser').val() ) || ( '' == $('#appUser').val() )) {  
            $disableBtn = true;  
            $('#warn-appUser').show();  
        }else{  
            $('#warn-appUser').hide();  
        }  
    }  
}
```



```

}

if(( 0 == $('#appCount').val() ) || ( '' == $('#appCount').val() )) {
    $disableBtn = true;
    $('#warn-appCount').show();
}else{
    $('#warn-appCount').hide();
}

if(( 0 == $('#appGenre').val() ) || ( '' == $('#appGenre').val() )) {
    $disableBtn = true;
    $('#warn-appGenre').show();
}else{
    $('#warn-appGenre').hide();
}

if( '' != $('#confirmMessage').text() || ( '' == ($('#mailaddress').val() ) || ( '' == ($('#mailaddressConfirm').val() ) ) ) ){
    $disableBtn = true;
    $('#warn-mailaddress').show();
}else{
    $('#warn-mailaddress').hide();
}
}

```

```

    if( !$('#agree').prop('checked') ){
        $disableBtn = true;
        $('#warn-agree').show();
    }else{
        $('#warn-agree').hide();
    }

    if( '' == $('#captchaAns').val() ){
        $disableBtn = true;
        $('#warn-captchaAns').show();
    }else{
        $('#warn-captchaAns').hide();
    }

    $('#submitAnalyze').prop('disabled', $disableBtn);
}

/*
 * 選択した大カテゴリに応じて小カテゴリを開く（可視にする）
 */
var viewSubGenre = function() {
    var gid = $('#option:selected', '#appGenre').attr('data-id');
    $('#.appGenreList').hide();

```

```

        $('#genre' + ('00' + gid).slice(-2)).show();
    }

    /*
    * CAPTCHA 画像の張り替え
    */
    var refreshCaptcha = function() {
        var now = new Date().getTime();
        $('#img.captchaImg').attr('src', 'captcha/captcha.php?' + now);
        $('#captchaAns').val('');
        checkSubmit();
    }

    /*
    * ファイル選択フォームで選択した apk ファイル名をテキストボックスに入れる
    */
    $('#input#apkFile').on('change', function() {
        $('#apkFileCover').val($(this).val().replace(/¥¥/g, '/').replace(/. *¥//, ''));
        checkSubmit();
    });

    /*
    * テキストボックスとボタンで apk ファイル選択フォームの動作が変わる

```

```

*/
$('#apkFileCover, #apkFileCoverBtn').on('click', function() {
    $('#input#apkFile').click();
});

/*
 * メールアドレスの入力チェック
 */
$('#mailaddress, #mailaddressConfirm').on('keyup', function() {
    if(( $('#mailaddress').val().length > 0 ) && ( $('#mailaddressConfirm').val().length > 0 )) {
        if( $('#mailaddress').val() != $('#mailaddressConfirm').val() ) {
            $('#confirmMessage').text(' 確認のためメールアドレスの入力は2 つとも一致している必要があります');
        } else {
            if( !( $('#mailaddress').val().match(/^[^@]+@.+\.[^./]+$/) ) ) {
                $('#confirmMessage').text(' メールアドレスの形式が正しくありません');
            } else {
                $('#confirmMessage').empty();
            }
        }
    }
    checkSubmit();
});

```

```

/*
 * 入力値の更新による必須項目チェック
 */
$('#appUser, #appCount, #agree').on('change', function() {
    checkSubmit();
});
$('#appGenre').on('change', function() {
    viewSubGenre();
    checkSubmit();
});
$('#captchaAns').on('keyup', function() {
    checkSubmit();
});

/*
 * 解析状況の確認をしてダイアログを出す
 */
$('#submitAnalyze').on('click', function() {
    $('#confirmText').text("");
    $('#bsMailaddress').val("");

    var appGenre2 = [];
    $('#input:checked', '#genre'+('00'+$('#option:selected', '#appGenre').attr('data-id')).slice(-2)).each(function() {

```

```

        appGenre2.push($(this).val());
    });

    var postdata = {
        'filename': $('#apkFileCover').val(),
        'appUser': $('#appUser').val(),
        'appCount': $('#appCount').val(),
        'appGenre': $('#appGenre').val(),
        'appGenre2': appGenre2,
        'mailaddress': $('#mailaddress').val(),
        'captcha': $('#captchaAns').val()
    }
    // 解析状況を確認する
    $.ajax({
        url: "send1st.php",
        type: "POST",
        data: postdata,
        timeout: 30000,
        dataType: 'json'
    }).done(function(data, status, xhr) {
        // 通信成功時の処理
        if(data.status) {
            // 受付中

```

```
var genreStr = data.appGenre;
if((null != data.appGenre2) && (0 < data.appGenre2.length)) {
    genreStr += ' (';
    var sep = '';
    for(i=0; data.appGenre2.length > i; i++) {
        genreStr += ""+sep + data.appGenre2[i];
        sep = ', ';
    }
    genreStr += ') ';
}
$('#confirm_apkFile').text(data.filename);
$('#confirm_appUser').text(data.appUser);
$('#confirm_appCount').text(data.appCount);
$('#confirm_appGenre').text(genreStr);
$('#confirm_mailaddress').text(data.mailaddress);

$('#bsMailaddress').val(data.mailaddress);
$('#bsKey').val(data.bsKey);

// 確認ダイアログを出す
$('#kakuninButtons').show();
$('#kakuninMsg').show();
$('#uploadProgress').hide();
```

```

        $('#uploadMsg').hide();
        $('#kakunin').modal();
    }else{
        // 受付中止
        $('#ccount').text(data.count);
        $('#ctime').text((null != data.avgtime)?data.avgtime:"--:--:--");
        $('#rejectMessage').text(data.message);

        // 受付拒否ダイアログを出す
        $('#reject').modal();
    }
}).fail(function(xhr, status, error) {
    // 通信失敗時の処理
    $('#ccount').text('-');
    $('#ctime').text("--:--:--");
    $('#rejectMessage').text("通信に失敗しました");

    $('#reject').modal();
});

});

/*

```



```
* ダイアログを閉じたときの CAPTCHA リフレッシュ
*/
$('#reject').on('hide.bs.modal', function() {
  refreshCaptcha();
});
$('#kakunin').on('hide.bs.modal', function() {
  refreshCaptcha();
});

/*
* ファイルアップロード
*/
$('#kakuninSubmit').on('click', function() {
  $('#lock').css({
    position: 'absolute',
    left: 0,
    top: 0,
    width: $(document).width(),
    height: $(document).height()
  }).show();

  $('#kakuninButtons').hide();
  $('#kakuninMsg').hide();
});
```

```

$("#upPgsBar").width("0%");
$('#uploadProgress').show();
$('#uploadMsg').show();

var fd = new FormData();
fd.append(' bsKey', $('#bsKey').val());
fd.append(' aMailaddress', $('#mailaddress').val());
fd.append(' aUser', $('#appUser').val());
fd.append(' aCount', $('#appCount').val());
fd.append(' aGenre', $('#appGenre').val());
var appGenre2 = [];
$('#input:checked', '#genre'+('00'+$('#option:selected', '#appGenre').attr('data-id')).slice(-2)).each(function() {
    appGenre2.push($(this).val());
});
fd.append(' aGenre2', appGenre2);
fd.append(' apkFile', $('#apkFile').get(0).files.item(0));
$.ajax({
    async: true,
    xhr : function() {
        // プログレスバーで進捗表示
        XHR = $.ajaxSettings.xhr();
        if(XHR.upload) {
            XHR.upload.addEventListener('progress', function(e) {

```

```

        progre = parseInt(e.loaded/e.total*10000)/100 ;

        $("#upPgsBar").width(progre+"%");
    }, false);
}
return XHR;
},
url: "apkupload.php",
type: "POST",
data: fd,
contentType: false,
processData: false,
dataType: 'json'
}).done(function(data, status, xhr) {
    $('#kakuninButtons').show();
    $('#kakuninMsg').show();
    $('#uploadProgress').hide();
    $('#uploadMsg').hide();

    if('success'==data.request) {
        $('#body').removeClass('modal-open');
        $('#modal-backdrop').remove();
        $('#kakunin').modal('hide');
    }
});

```

```

    $('#bsReqNo').val(data.request_no);
    $('#send2nd').submit();
    $('#lock').hide();
} else {
    $('body').removeClass('modal-open');
    $('.modal-backdrop').remove();
    $('#kakunin').modal('hide');

    $('#ccount').text(data.count);
    $('#ctime').text((null != data.avgtime)?data.avgtime:"--:--:--");

    $('#rejectMessage').text(data.message);
    $('#lock').hide();
    $('#reject').modal();
}
}).fail(function(xhr, status, error) {
    // 通信失敗時の処理
    $('body').removeClass('modal-open');
    $('.modal-backdrop').remove();
    $('#kakunin').modal('hide');

    $('#ccount').text('-');

```

```
        $('#ctime').text("--:--:--");

        $('#rejectMessage').text("通信に失敗しました");
        $('#lock').hide();
        $('#reject').modal();
    });
});

refreshCaptcha();
viewSubGenre();
checkSubmit();

});
```

3.6 CAPTCHA 画像生成

W1004 解析依頼完了画面のソースコードを以下に示す。

```
<?php
include_once("php-captcha.inc.php");

$font_list = array(
    '../fonts/ipag.ttf'
```

```
, '../fonts/ipam.ttf'  
);  
  
$captcha = new PhpCaptcha($font_list, 200, 40);  
  
$captcha->CaseInsensitive(false);  
$captcha->SetCharSet('0-9');  
$captcha->SetNumChars(5);  
$captcha->UseColour(false);  
  
$captcha->Create();
```

3.7 W1004 解析依頼完了画面

W1004 解析依頼完了画面のソースコードを以下に示す。

```
<?php  
include_once('wording.php');  
$TITLE = $WORDING['WD-3001'];  
include_once('header.php');  
$mailaddress = filter_input(INPUT_POST, 'bsMailaddress');  
$request_no = filter_input(INPUT_POST, 'bsReqNo');
```

```

if(! (filter_var($request_no, FILTER_VALIDATE_INT) && filter_var($mailaddress, FILTER_VALIDATE_EMAIL) ) ){
    $ErrWord = array('title' => $WORDING['WD-1001']
                    , 'head' => $WORDING['WD-9003']
                    , 'text' => $WORDING['WD-9004'] );
    include_once('error.php');
    exit;
}

// DB への接続
include_once('ppDb.php');
$db = new ppDB();
if( NULL !== $db->getDbError() ){
    header('Location: dberror.php');
    exit;
}

// 解析状況を取得
$status = $db->getAnalysisStatus();

$db->close();
?>
<div class="container">
    <div class="title bg-primary">

```

```

    <h1><?php echo $WORDING['WD-3001']; ?></h1>
    <div class="titleback"><a href="."/ "><?php echo $WORDING['WD-1009']; ?></a></div>
</div>
<div class="titlebackClear"></div>

<div class="sectionBody">
    <div class="subsectionText">
        <?php echo $WORDING['WD-3002']; ?>
    </div>

    <dl class="dl-horizontal">
        <dt><?php echo $WORDING['WD-3003']; ?> &nbsp;</dt>
        <dd id="rnumber"><?php echo htmlspecialchars($request_no, ENT_QUOTES|ENT_HTML5, 'UTF-8'); ?></dd>
        <dt><?php echo $WORDING['WD-7001']; ?> &nbsp;</dt>
        <dd><?php echo $WORDING['WD-7002']; ?> <span id="rcount"><?php echo $status['count']; ?></span> <?php echo $WORDING['W
D-7003']; ?></dd>
        <dt><?php echo $WORDING['WD-7007']; ?> &nbsp;</dt>
        <dd id="vertime"><span><?php echo !empty($status['avgtime'])? $status['avgtime']: '---:--:--'; ?></span></dd>
    </dl>

    <div class="subsectionMessage">
        <?php echo $WORDING['WD-3007']; ?><br>
        <?php echo $WORDING['WD-3005']; ?><?php echo htmlspecialchars($mailaddress, ENT_QUOTES|ENT_HTML5, 'UTF-8'); ?><?php echo

```



```
$WORDING['WD-3006']; ?><br>
    </div>

    <div class="btnSubmit">
        <button type="button" id="resultReturn" class="btn btn-primary"><?php echo $WORDING['WD-3004']; ?></button>
    </div>
</div>
</div>
<?php
$formJS = array_merge($formJS, array('js/ppResult.js'));
include_once('footer.php');
```

3.8 解析依頼完了画面用 JavaScript

解析依頼完了画面用 JavaScript のソースコードを以下に示す。

```
$(function () {
    $('#resultReturn').on('click', function() {
        location.href = "apkform.php";
    });
});
```

3.9 W2001 統計情報画面

W2001 統計情報画面のソースコードを以下に示す。

```
<?php
include_once('wording.php');
$title = $WORDING['WD-4001'];
include_once('header.php');

// DB 接続
include_once('ppDb.php');
$db = new ppDB();
if( NULL !== $db->getDbError() ){
    header('Location: dberror.php');
    exit;
}

// 解析状況を取得
$status = $db->getAnalysisStatus();

$db->close();

// 統計情報ファイル一覧取得
include_once('ppFiles.php');
```

```

$lf = new ppFiles();
$logs = $lf->getFileList();
?>
<div class="container">
  <div class="title bg-primary">
    <h1><?php echo $WORDING['WD-4001']; ?></h1>
    <div class="titleback"><a href="."/ "><?php echo $WORDING['WD-1009']; ?></a></div>
  </div>
  <div class="titlebackClear"></div>

  <div>
    <div class="subsectionTitle">
      <h2><?php echo $WORDING['WD-4002']; ?></h2>
    </div>
    <div class="subsectionBody">
      <dl class="dl-horizontal topblank8">
        <dt><?php echo $WORDING['WD-7001']; ?></dt>
        <dd>
<?php if($status['accept']) { ?>
          <?php echo $WORDING['WD-7004']; ?> <span><?php echo $status['count']; ?></span> <?php echo $WORDING['WD-7005']; ?>
        >
<?php }else{ ?>
          <?php echo $WORDING['WD-7006']; ?>

```

```

<?php } ?>
    </dd>
    <dt><?php echo $WORDING['WD-7007']; ?></dt>
    <dd><span><?php echo !empty($status['avgtime'])? $status['avgtime']:' --:--:--'; ?></span></dd>
</dl>
</div>

<div class="subsectionTitle">
    <h2><?php echo $WORDING['WD-4003']; ?></h2>
</div>
<div class="subsectionBody">
<?php if( 0 < count($logs)) { ?>
    <table class="table table-condensed">
        <tr><th><?php echo $WORDING['WD-4004']; ?></th><th><?php echo $WORDING['WD-4005']; ?></th></tr>
<?php
foreach($logs as $l) {
    $yyyymm = substr($l['yyyymm'], 0, 4) . '年' . substr($l['yyyymm'], 4, 2) . '月';
?>
        <tr><td><?php echo $yyyymm; ?></td><td><a href="<?php echo $l['urlpath']; ?>"><?php echo $l['filename']; ?></a></td>
<</tr>
<?php
}
?>

```

```
        </table>
<?php }else{ ?>
        <?php echo $WORDING['WD-4006']; ?>
<?php } ?>
        </div>
    </div>
</div>

<?php
include_once(' footer.php ');
```

3.10 文言定義ファイル

文言定義ファイルのソースコードを以下に示す。

```
<?php
/*
 * 画面で使用する文言の宣言
 *
 */

$WORDING = array(

    // W0001 タイトル画面向け
    'WD-1001' => 'Android アプリ解析システム',
    'WD-1002' => 'Android アプリ解析システム',
    'WD-1003' => 'メッセージ',
    'WD-1004' => 'お知らせ',
    'WD-1005' => '解析受付・情報',
    'WD-1006' => 'Android アプリ解析依頼',
    'WD-1007' => '統計情報',
    'WD-1008' => '実証実験関連資料',
    'WD-1009' => 'タイトルへ戻る',

    // W1001 解析依頼 Web フォーム画面向け
```

'WD-2001' => 'Android アプリ解析依頼',
'WD-2002' => 'APK 指定',
'WD-2003' => 'APK ファイル指定',
'WD-2004' => 'APK ファイル指定',
'WD-2008' => '本システムの利用回数',
'WD-2009' => 'アプリケーション分野',
'WD-2010' => 'ユーザー情報指定',
'WD-2011' => 'メールアドレス',
'WD-2012' => '返信先メールアドレス指定',
'WD-2013' => '返信先メールアドレス確認',
'WD-2014' => '本システムの利用規約',
'WD-2015' => '本システムの利用規約に同意する',
'WD-2016' => '左の数字を入力してください',
'WD-2017' => '解析依頼',
'WD-2018' => 'ファイル選択',
'WD-2019' => '本システム利用者',
'WD-2020' => '必須',
'WD-2021' => '選択してください (必須)',
'WD-2022' => '選択してください (必須)',
'WD-2023' => '選択してください (必須)',

// W1002 受付内容確認ダイアログ向け

'WD-2101' => '解析依頼内容確認',

'WD-2102' => ' 以上の内容で解析を依頼します',
'WD-2103' => ' キャンセル',
'WD-2104' => ' 依頼',
'WD-2105' => ' APK ファイルのアップロードをしています',

// W1004 受付完了画面向け

'WD-3001' => ' Android アプリ解析依頼完了',
'WD-3002' => ' アプリケーションの検査解析の受付を完了しました',
'WD-3003' => ' 受付番号',
'WD-3004' => ' 戻る',
'WD-3005' => ' 解析結果は JSAS@jssec.org から ',
'WD-3006' => ' 宛てにメールが届きます',
'WD-3007' => ' 受付番号は解析結果が届くまで控えておいてください',
'WD-3008' => ' 解析には最大で1時間ほどかかります',

// W1003 受付拒否ダイアログ向け

'WD-3101' => ' 解析依頼を受け付けることができませんでした',
'WD-3102' => ' アプリケーションの検査解析依頼を受け付けることができません',
'WD-3103' => ' 閉じる',
'WD-3104' => ' しばらく時間をおいて再度お試しください',
'WD-3105' => ' 入力値に問題があります。確認してください',


```
// W2001 統計情報画面向け
'WD-4001' => '統計情報',
'WD-4002' => 'リアルタイム情報',
'WD-4003' => 'Android アプリ静的解析 統計情報ダウンロード',
'WD-4004' => '対象年月',
'WD-4005' => 'ファイル名',
'WD-4006' => 'ダウンロード可能な統計情報はありません',

// 共通・キューステータス表示向け
'WD-7001' => 'Android アプリ解析の受付状況',
'WD-7002' => '現在処理待ち',
'WD-7003' => '件です',
'WD-7004' => '受付可能 (現在処理待ち',
'WD-7005' => '件です)',
'WD-7006' => '受付不可能 (現在受付を中止しています)',
'WD-7007' => '解析時間の平均',

// 共通・エラーメッセージ用
'WD-9001' => 'DB 接続エラー',
'WD-9002' => 'データベースへの接続でエラーが発生しました',
'WD-9003' => 'エラーが発生しました',
'WD-9004' => '入力値が不正です'

);
```

```
// システム利用回数用文言
```

```
$COUNT = array(
```

```
    '1' => '初回',
```

```
    '2' => '二回目',
```

```
    '3' => '三回目',
```

```
    '4' => '四回目',
```

```
    '5' => '五回目以上',
```

```
);
```

```
// システム利用回数用文言
```

```
$COUNT = array(
```

```
    '1' => '初回',
```

```
    '2' => '二回目',
```

```
    '3' => '三回目',
```

```
    '4' => '四回目',
```

```
    '5' => '五回目以上',
```

```
);
```

4 Ajax リクエスト処理ソースコード

4.1 解析状況確認

解析依頼後に解析確認するためのソースコードを以下に示す。

```
<?php
/*
 * send1st.php
 *
 * 返却値 json 形式
 * 解析受付成功時
 *   {
 *     appUser : 利用者分類
 *     appCount : システム利用回数
 *     appGenre : 大カテゴリ
 *     appGenre2[] : 小カテゴリ (配列)
 *     mailaddress : メールアドレス
 *     bsKey : セッションキー
 *   }
 * 解析受付失敗時
 *   {
```

```
*      message : エラーメッセージ
*      count  : 解析待ち数
*      avgtime : 解析平均時間
*    }
*/

error_reporting(E_ALL & ~E_STRICT);

include_once("captcha/php-captcha.inc.php");
include_once('wording.php');

$returnStatus = false;

// CAPTCHA の入力をチェック
$postCaptcha = filter_input(INPUT_POST, 'captcha');
$captcha = false;
if (PhpCaptcha::Validate($postCaptcha, false)) {
    $captcha = true;
}

$params = array();
$message = '';
```

```
include_once('ppDb.php');
include_once('ppSession.php');

// DB 接続
$db = new ppDB();
$sess = new ppSession();

// DB 接続エラー
if( NULL !== $db->getDbError() ){
    header('Content-Type: application/json');
    header("X-Content-Type-Options: nosniff");
    echo json_encode( array('status' => false,
        'message' => $WORDING['WD-9002'],
        'count' => '-',
        'avgtime' => '---:--:--'
    ), JSON_HEX_TAG | JSON_HEX_APOS | JSON_HEX_QUOT | JSON_HEX_AMP);
    exit;
}

// 受付状況・解析平均時間・待ち件数を取得
$status = $db->getAnalysisStatus();
```

```
if($captcha) {
    // CAPTCHA が正しく通ったとき
    $returnStatus = true;

    // ブラウザからの直接 json リクエストを受け付けない
    if($_SERVER['HTTP_X_REQUESTED_WITH'] !== 'XMLHttpRequest'){
        $message = $WORDING['WD-3105'];
        $returnStatus = false;
    }elseif( $status['accept'] ){
        // 解析受付中であるとき

        $post_user = filter_input(INPUT_POST, 'appUser');
        $userStr = $db->getUserNameFromId($post_user);
        if(false === $userStr) {
            $message = $WORDING['WD-3105'];
            $returnStatus = false;
        }else{
            $param['appUser'] = $userStr;
        }

        $post_ct = filter_input(INPUT_POST, 'appCount');
        if( filter_var($post_ct, FILTER_VALIDATE_INT) && isset($COUNT[$post_ct]) ){
            $param['appCount'] = $COUNT[$post_ct];
        }
    }
}
```

```

}else{
    $message = $WORDING['WD-3105'];
    $returnStatus = false;
}

$genre = filter_input(INPUT_POST, 'appGenre');
$genre2 =filter_input(INPUT_POST, 'appGenre2', FILTER_DEFAULT , FILTER_REQUIRE_ARRAY);
$genreStrAry = $db->getGenreNameFromId($genre, $genre2);

if(NULL != $genreStrAry['name']){
    $param['appGenre'] = $genreStrAry['name'];
    $param['appGenre2'] = $genreStrAry['subgenre'];
}else{
    $message = $WORDING['WD-3105'];
    $returnStatus = false;
}

$param['filename'] = filter_input(INPUT_POST, 'filename');

$mailaddress= filter_input(INPUT_POST, 'mailaddress');
if( filter_var($mailaddress, FILTER_VALIDATE_EMAIL) ){
    $param['mailaddress'] = $mailaddress;
}else{

```

```
        $message = $WORDING['WD-3105'];
        $returnStatus = false;
    }

    $param['bsKey'] = $sess->setbsKey();
} else {
    // 解析の受付を中止している場合
    $message = $WORDING['WD-3104'];
    $param['count'] = $status['count'];
    $param['avgtime'] = $status['avgtime'];

    $returnStatus = false;
}

} else {
    $message = $WORDING['WD-3105'];
    $param['count'] = $status['count'];
    $param['avgtime'] = $status['avgtime'];
}

$db->close();
```



```
$param['status'] = $returnStatus;
if(!$returnStatus) {
    $param['message'] = $message;
}

header('Content-Type: application/json');
header("X-Content-Type-Options: nosniff");
echo json_encode($param, JSON_HEX_TAG | JSON_HEX_APOS | JSON_HEX_QUOT | JSON_HEX_AMP);
```

4.1 apk ファイルアップロード

apk ファイルアップロードのソースコードを以下に示す。

```
<?php
/*
 * apk ファイルをアップロードを行い、情報を DB に格納する
 *
 * 返却値 json 形式
 * 解析受付成功時
 * {
 *     request: 成功時 → true
 *     request: 失敗時 → false
```

```

*      message : 表示メッセージ
*      count   : 解析待ち数
*      avgtime  : 解析平均時間
*   }
*
*/

error_reporting(E_ALL & ~E_STRICT);

setlocale(LC_ALL, 'ja_JP.UTF-8');

// DB への接続
include_once('ppDb.php');
$db = new ppDB();
if( NULL !== $db->getDbError() ){
    header('Content-Type: application/json');
    header("X-Content-Type-Options: nosniff");
    echo json_encode( array('request' => 'fail',
                           'message' => $WORDING['WD-9002'],
                           'count'  => '-',
                           'avgtime' => '--:--:--'
                        ), JSON_HEX_TAG | JSON_HEX_APOS | JSON_HEX_QUOT | JSON_HEX_AMP);
}
exit;

```

```

}

// 解析状況を取得
$status = $db->getAnalysisStatus();

include_once('ppSession.php');
$sess = new ppSession();

$ret = false;
$bsKey = filter_input(INPUT_POST, 'bsKey');
$reqId = '';

$is_upError = (!isset($_FILES['upfile']['error']) || !is_int($_FILES['upfile']['error']));

if($is_upError && $status['accept'] && $bsKey === $sess->getbsKey()
    && ($_SERVER['HTTP_X_REQUESTED_WITH'] === 'XMLHttpRequest')) {
    $param = array();

    $param['apkname'] = mb_convert_encoding(basename($_FILES['apkFile']['name']), 'UTF-8');

    foreach ( array('aMailaddress', 'aUser', 'aCount', 'aGenre', 'aGenre2') as $ap) {
        $param[$ap] = filter_input(INPUT_POST, $ap);
    }
}

```

```

$Genre2 = array();
if(!empty($param['aGenre2'])) {
    $Genre2 = explode(',', $param['aGenre2']);
}

if( ( '' !== $param['apkname'] ) && (is_uploaded_file($_FILES['apkFile']['tmp_name']))){
    $reqId = $db->insertAnalysisRequest();
    $uploadfile = NULL;

    if(is_numeric($reqId)) {
        if( mkdir(UPLOAD_DIR . $reqId) ){
            chmod(UPLOAD_DIR . $reqId, 0755);
            $uploadfile = UPLOAD_DIR . $reqId . '/' . $param['apkname'];
        }
    }

    if( (false !== $reqId) && (!empty($uploadfile))
        && (move_uploaded_file($_FILES['apkFile']['tmp_name'], $uploadfile))
        && filter_var($param['aMailaddress'], FILTER_VALIDATE_EMAIL) ){
        $db->transaction('start');
        $ins = true;
        if( false !== $reqId ){
            $r1 = $db->insertResponseInfo($reqId, $param['apkname'], $param['aMailaddress']);
        }
    }
}

```

```

    $r2 = $db->insertApplicationInfoCategory($reqId, $param['aUser'], $param['aCount'], $param['aGenre']);
    if(0 < count($param['aGenre2'])) {
        $r3 = $db->insertApplicationInfoFamily($reqId, $param['aGenre'], $pGenre2);
    }else{
        $r3 = true;
    }
    if(!($r1 && $r2 && $r3)){
        $ins = false;
    }
} else {
    $ins = false;
}
if( false === $ins){
    $db->transaction('rollback');
    $ret = false;
} else {
    $db->transaction('commit');
    $ret = true;
}
} else {
    $ret = false;
}
} else {

```

```
        $ret = false;
    }
}
$error = $db->getDbError();
$db->close();

$rs = array('request' => $ret?'success':'fail');
if($ret) {
    $rs['request_no'] = $reqId;
} else {
    $rs['count'] = $status['count'];
    $rs['avgtime'] = $status['avgtime'];
}

header('Content-Type: application/json');
header("X-Content-Type-Options: nosniff");
echo json_encode($rs , JSON_HEX_TAG | JSON_HEX_APOS | JSON_HEX_QUOT | JSON_HEX_AMP);
```

5 利用クラスソースコード

5.1 DB 接続クラス

DB 接続クラスのソースコードを以下に示す。

```
<?php
include_once('dbconf.php');

/*
 * DB 接続クラス
 *
 */
class ppDb{
    private $pdo = NULL;
    private $err = NULL;

    /*
     * コンストラクタ
     * DB (PostgreSQL) への接続
     */
    public function __construct() {
        $dsn = 'pgsql:dbname=' . DB_NAME . ':';
        $dsn .= 'host=' . DB_HOST . ';port=' . DB_PORT . ':';
    }
}
```

```

    try{
        $this->pdo = new PDO($dsn, DB_USER, DB_PASS);
    }catch (PDOException $e){
        $this->err = $e->getMessage();
    }
}

/*
 * DB エラーの返却
 */
public function getDbError() {
    return $this->err;
}

/*
 * getAnalysisStatus()
 * 解析キュー状況の取得
 * 返却値
 *     array( 'accept' => 受付状況, 'count' => 解析待ち数, 'avgtime' => 平均解析時間 )
 * 受付状況: 受付可能 true
 *           受付付加 false
 */

```



```

public function getAnalysisStatus() {
    $count = 0;
    $avgtime = '00:00:00';
    $accept = false;

    try{
        // 解析待ち・解析中のキューの数の取得
        $sql = 'SELECT COUNT(request_no) AS count_analysisreq FROM analysis_request';
        $sql.= ' WHERE analysis_status = 0 ';
        $sql.= "OR ( analysis_status = 1 AND CURRENT_TIMESTAMP < analysis_starttime + CAST('3 hour' AS INTERVAL) )";
        $stmt = $this->pdo->prepare($sql);
        $stmt->execute();
        $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
        if( 0 < count($result) ){
            $count = (int)$result[0]['count_analysisreq'];
        }

        // 過去10件の解析時間平均を取得
        $sql = 'SELECT CAST(AVG(analysis_endtime - analysis_starttime) AS interval(0)) AS avgtime FROM analysis_request';
        $sql.= ' WHERE request_no IN (';
        $sql.= ' SELECT request_no FROM analysis_request';
        $sql.= ' WHERE analysis_status = 2';
        $sql.= ' AND analysis_starttime IS NOT NULL AND analysis_endtime IS NOT NULL';
    }
}

```

```

    $sql.= ' ORDER BY request_no DESC OFFSET 0 LIMIT 10' ;
    $sql.= ' )' ;
    $stmt = $this->pdo->prepare($sql);
    $stmt->execute();
    $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
    if( 0 < count($result) ){
        $avgtime = $result[0]['avgtime'];
    }
} catch(PDOException $e) {
    // DB エラー発生時処理
    $this->err = $e->getMessage();
}

/*
 * キューの蓄積件数から受付可能か否かを判定 (ANALYSIS_REQUEST_MAX 未満かで判定)
 * 件数が規定未満 $accept : true
 * 件数が規定異常 $accept : false
 */
if(ANALYSIS_REQUEST_MAX > $count){
    $accept = true;
}

$ret = array('accept' => $accept, 'count' => $count, 'avgtime' => $avgtime );

```

```

    return $ret;
}

/*
 * getGenreListAll()
 * アプリケーションのカテゴリ分類・小分類を全取得する
 * 返却値
 * array( array( 'name' => '大カテゴリ名', subgenre => NULL ) )
 * array( array( 'name' => '大カテゴリ名', subgenre => array( '小カテゴリ名 1', '小カテゴリ名 1' …… ) ) )
 */
public function getGenreListAll() {
    $genre = array();

    try {
        // 大カテゴリを DB より読み込み
        $sql = 'SELECT category_id, category_name FROM category';
        $sql.= ' WHERE NOT delete_flg = 1';
        $sql.= ' ORDER BY display_order';
        $stmt = $this->pdo->prepare($sql);
        $stmt->execute();
        $resultCategory = $stmt->fetchAll(PDO::FETCH_ASSOC);
    }
}

```

```

// 上で取得した大カテゴリ配下の小カテゴリを DB から読み込み
$holder = substr(str_repeat(',?', count($resultCategory)), 1);
$sql = 'SELECT category_id, family_id, family_name FROM family';
$sql.= ' WHERE NOT delete_flg = 1';
$sql.= sprintf(' AND category_id IN (%s)', $holder);
$sql.= ' ORDER BY category_id, display_order';
$stmt = $this->pdo->prepare($sql);
$i=1;
foreach( $resultCategory as $r ){
    $stmt->bindValue($i, (int)$r['category_id'], PDO::PARAM_INT);
    $i++;
}
$stmt->execute();
$resultFamily = $stmt->fetchAll(PDO::FETCH_ASSOC);

// 整形
// 小分類をカテゴリ ID 毎に配列にしておく
$subGenre = array();
foreach($resultFamily as $r){
    $subGenre[$r['category_id']][] = array('id'          =>$r['family_id'],
                                           'name'       =>$r['family_name']
                                           );
}

```

```

// 大分類を整形
foreach($resultCategory as $r) {
    array_push($genre, array('id'=>$r['category_id'],
        'name'=>$r['category_name'],
        'subGenre'=> isset($subGenre[$r['category_id']])?$subGenre[$r['category_id']]:array()
    ));
}
} catch (PDOException $e) {
    $this->err = $e->getMessage();
}

return $genre;
}

/*
* getGenreNameFromId($id, $subid[])
* カテゴリの ID 番号からカテゴリ名を返す
* 引数
* $id : 大カテゴリ ID (必須)
* $subid[] : 小カテゴリ ID 配列 (任意)
* 返却値
* array('name'=>'大カテゴリ名', subgenre => NULL )
* array('name'=>'大カテゴリ名', subgenre => array('小カテゴリ名 1', '小カテゴリ名 1' ……))

```

```

*
*/
public function getGenreNameFromId($id, $subid = NULL) {

    $genreName = NULL;
    $subgenreName = NULL;

    // 大カテゴリの名称を ID から DB より取得
    $sql = 'SELECT category_name FROM category';
    $sql.= ' WHERE NOT delete_flg = 1';
    $sql.= ' AND category_id = :id';
    $stmt = $this->pdo->prepare($sql);
    $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);
    $stmt->execute();
    $resultCategory = $stmt->fetchAll(PDO::FETCH_ASSOC);
    if( 0 < count($resultCategory) ){
        $genreName = $resultCategory[0]['category_name'];
    }

    // 引数で小カテゴリ ID が与えられた場合は、小カテゴリの名称を ID から取得
    $holder = '';
    if(!empty($subid) && is_array($subid)){
        $holder = substr(str_repeat(',?', count($subid)), 1);
    }
}

```

```

    $sql = 'SELECT family_name FROM family';
    $sql.= ' WHERE NOT delete_flg = 1';
    $sql.= sprintf(' AND family_id IN (%s)', $holder);
    $sql.= ' AND category_id = ?';
    $sql.= ' ORDER BY display_order';
    $stmt = $this->pdo->prepare($sql);
    $stmt->execute(array_merge($subid, array((int)$id)));
    $resultFamily = $stmt->fetchAll(PDO::FETCH_ASSOC);
    if( 0 < count($resultFamily) ){
        $subgenreName = array();
        foreach($resultFamily as $r){
            array_push($subgenreName, $r['family_name']);
        }
    }
}

return array('name'=>$genreName, 'subgenre' => $subgenreName);
}

/*
 * getUserListAll()
 * 利用者マスターから利用者分類リストを取得

```

```

* 返却値
* 配列
*   キー id : 利用者分類 ID
*   キー name: 利用者分類名
*/
public function getUserListAll() {
    $user = array();
    try {
        $sql = 'SELECT service_user_id, service_user_name FROM service_user';
        $sql.= ' WHERE NOT delete_flg = 1';
        $sql.= ' ORDER BY display_order';

        $stmt = $this->pdo->prepare($sql);
        $stmt->execute();
        $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
        foreach($result as $r) {
            array_push($user, array('id' => $r['service_user_id'], 'name' => $r['service_user_name']));
        }
    } catch (PDOException $e) {
        $this->err = $e->getMessage();
    }

    return $user;
}

```



```
}

/*
 * getUserFromId($id)
 * 利用者 ID に該当する利用者分類名を利用者マスターから取得
 * 引数
 *   $id : 利用者 ID
 * 返却値
 *   成功時: 利用者分類名称
 *   失敗時: false
 */
public function getUserFromId($id) {

    try {
        $sql = 'SELECT service_user_name FROM service_user';
        $sql.= ' WHERE NOT delete_flg = 1';
        $sql.= ' AND service_user_id = :uid';
        $sql.= ' ORDER BY display_order';

        $stmt = $this->pdo->prepare($sql);
        $stmt->bindValue(':uid', (int)$id, PDO::PARAM_INT);
        $stmt->execute();
    }
}
```

```

        $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
        if( 0 < count($result) ){
            return $result[0]['service_user_name'];
        }
    }catch (PDOException $e) {
        $this->err = $e->getMessage();
    }

    return false;
}

/*
 * transaction($mode)
 * DB のトランザクション指定
 * 引数 (必須)
 *   $mode: start トランザクション開始
 *           commit コミット
 *           rollback ロールバック
 * 返却値
 *   成功時: true
 *   失敗時: false
 */
public function transaction($mode) {

```

```
$dbf = false;
try {
    switch($mode) {
        case 'start' :
            $dbf = $this->pdo->beginTransaction();
            break;
        case 'commit' :
            $dbf = $this->pdo->commit();
            break;
        case 'rollback' :
            $dbf = $this->pdo->rollBack();
            break;
        default:
            $dbf = false;
    }
    return $dbf;
} catch (PDOException $e) {
    $this->err = $e->getMessage();
}
return false;
}

/*
```

```

* insertAnalysisRequest()
* 解析依頼テーブルに、レコードを insert する
* 返却値
*   成功時: 受付番号 (request_no)
*   失敗時: false
*/
public function insertAnalysisRequest() {
    try{
        $sql = "INSERT INTO analysis_request";
        $sql.= " (analysis_status, request_datetime, insert_datetime, update_datetime)";
        $sql.= " VALUES (0, now(), now(), now() )";
        $stmt = $this->pdo->prepare($sql);
        $stmt->execute();

        $sql = "SELECT CURRVAL('analysis_request_request_no_seq') AS curval";
        $stmt = $this->pdo->prepare($sql);
        $stmt->execute();

        $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
        if( 0 < count($result) ){
            return $result[0]['curval'];
        }
    } catch (PDOException $e) {

```

```

        $this->err = $e->getMessage();
    }
    return false;
}

/*
 * insertResponseInfo($id, $aplname, $mailaddress)
 * 返信情報テーブルに、レコードを insert する
 * 引数
 *     $id : 受付番号
 *     $aplname : apk ファイル名
 *     $mailaddress : メールアドレス (結果返却先)
 * 返却値
 *     成功時: true
 *     失敗時: false
 */
public function insertResponseInfo($id, $aplname, $mailaddress) {
    if(empty($id) || !is_numeric($id)) {
        return false;
    }
    try {
        $sql = "INSERT INTO resp";
        $sql.= " (request_no, application_name, email, insert_datetime, update_datetime)";
    }
}

```

```

        $sql.= " VALUES (:id , :apkname , :mailaddress , now(), now())";
        $stmt = $this->pdo->prepare($sql);
        $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);
        $stmt->bindValue(':apkname', $apkname, PDO::PARAM_STR);
        $stmt->bindValue(':mailaddress', $mailaddress, PDO::PARAM_STR);
        $stmt->execute();

        return true;
    }catch (PDOException $e) {
        $this->err = $e->getMessage();
    }
    return false;
}

/*
 * insertApplicationInfoCategory($id, $user, $count, $category)
 * アプリケーション情報テーブルに、レコードを insert する
 * 引数
 *     $id : 受付番号
 *     $user : 利用者分類
 *     $count : システム利用回数 (初回は 1、5 回以上は 5)
 *     $category : アプリケーション大分類
 * 返却値

```

```

*    成功時: true
*    失敗時: false
*/
public function insertApplicationInfoCategory($id, $user, $count, $category) {
    if(empty($id) || !is_numeric($id)){
        return false;
    }

    try{
        $sql = "INSERT INTO application_info_category";
        $sql.= " (request_no, service_user, analysis_count, category, insert_datetime, update_datetime)";
        $sql.= " VALUES (:id , :user , :count , :category , now(), now())";
        $stmt = $this->pdo->prepare($sql);
        $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);
        $stmt->bindValue(':user', (int)$user, PDO::PARAM_INT);
        $stmt->bindValue(':count', (int)$count, PDO::PARAM_INT);
        $stmt->bindValue(':category', (int)$category, PDO::PARAM_INT);
        $stmt->execute();

        return true;
    }catch (PDOException $e) {
        $this->err = $e->getMessage();
    }
}

```

```

        return false;
    }

    /*
    * insertApplicationInfoFamily($id, $category, $subcategory)
    * アプリケーション情報（小）テーブルに、レコードを insert する
    * 引数
    *     $id : 受付番号
    *     $category : アプリケーション大分類
    *     $family : アプリケーション小分類（配列）
    * 返却値
    *     成功時: true
    *     失敗時: false
    */
    public function insertApplicationInfoFamily($id, $category, $subcategory) {
        if(empty($id) || !is_numeric($id)){
            return false;
        }
        try{
            $sql = "INSERT INTO application_info_family";
            $sql.= " (request_no, category, family, insert_datetime, update_datetime)";
            $sql.= " VALUES (:id , :category , :family , now(), now())";
            $stmt = $this->pdo->prepare($sql);

```



```

        $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);
        $stmt->bindValue(':category', (int)$category, PDO::PARAM_INT);
        foreach($subcategory as $fm) {
            $stmt->bindValue(':family', (int)$fm, PDO::PARAM_INT);
            $stmt->execute();
        }

        return true;
    } catch(PDOException $e) {
        $this->err = $e->getMessage();
    }
    return false;
}
}
/*
 * close()
 * DB を閉じる
 */
public function close() {
    $this->pdo = NULL;
}
}
}

```

5.2 セッションクラス

セッションクラスのソースコードを以下に示す。

```
<?php
/*
 * セッションクラス
 *
 */
class ppSession{

    /*
     * setbsKey()
     * セッションに数値キーを作成する
     * 返却値
     * キー
     */
    public function setbsKey() {
        $bsKey = bin2hex(openssl_random_pseudo_bytes(16));

        // セッションの状態を判定し、開始状態でなければ開始する
        if ( PHP_SESSION_ACTIVE !== session_status() ) {
            session_start();
        }
    }
}
```

```
    $_SESSION[' bsKey' ] = $bsKey;

    return $bsKey;
}

/*
 * getbsKey()
 * セッションに設定されている数値キーを取得する
 * 返却値
 * キー
 * セッションにキーが設定されていない場合 : false
 */
public function getbsKey() {

    // セッションが開始されていなければ開始
    if ( PHP_SESSION_ACTIVE !== session_status() ) {
        session_start();
    }

    // セッションに設定されたキーの値を取得（なければ false）
    $bsKey = isset($_SESSION[' bsKey' ])?$_SESSION[' bsKey' ]:false;
    unset($_SESSION[' bsKey' ]);
}
```

```
        return $bsKey;
    }
}
```

5.3 ファイルクラス

ファイルクラスのソースコードを以下に示す。

```
<?php
include_once('dbconf.php');

/*
 * ファイル操作クラス
 *
 */
class ppFiles{

    private $files = array();
    private $err = NULL;

    /*
     * コンストラクタ
```

```

* 統計情報格納ディレクトリの一覧を保持
*/
public function __construct() {

    if ($dh = @opendir(LOG_FILEPATH)) {
        while (false !== ($f = readdir($dh))) {
            $m = array();
            if ($f != "." && $f != ".." && preg_match(LOG_REGPATTERN, $f, $m)) {
                $this->files[$m[1]] = $m[0];
            }
        }
        closedir($dh);
    }else{
        $this->err = 'Cannot open dir'.LOG_FILEPATH;
    }
    krsort($this->files, SORT_NUMERIC);
}

/*
* getFileList() 統計情報ファイル一覧取得
* 返却値
* 配列
* キー yyyyymm : 年月

```

```
*   キー urlpath : リンク URL
*   キー filename : ファイル名
*/
public function getFileList() {
    $ret = array();
    foreach($this->files as $fkey=>$fval) {
        array_push($ret, array('yyyymm' => $fkey,
                                'urlpath' => LOG_URLPATH.'/'.$fval,
                                'filename' => $fval
                                ));
    }
    return $ret;
}
}
```

6 CSS ファイル

ファイルクラスのソースコードを以下に示す。

```
@charset "UTF-8";

body {
    font-size: medium;
}

.nodisplay{
    display: none;
}

input[type=file].nodisplay{
    display: none !important;
}

.title{
    padding: 0 15px;
}

h1{
    display: inline-block;
}
```

```
.titleback{
    float: right;
    height: 60px;
    padding-top:20px;
}

.titleback a{
    text-decoration: underline;
    color: #fff;
    font-size: small;
}

.titlebackClear{
    clear: both;
}

.sectionBody{
    margin-top: 12px;
    margin-bottom: 12px;
    padding: 10px 16px;
// border: 1px solid #aaa;

// border-radius: 6px;
```



```
}  
  
.subsectionTitle{  
    background-color: #d9edf7;  
    // margin-top: 12px;  
    padding: 4px;  
    // border: 1px solid #aaa;  
    border-bottom: 0;  
  
    /* border-top-left-radius: 6px;  
       border-top-right-radius: 6px;  
    */  
}  
  
.subsectionTitle h2{  
    font-size: 24px;  
    margin: 8px 12px 4px 12px;  
    padding: 0;  
}  
  
.subsectionBody {  
    margin: 0;
```

```
margin-bottom: 12px;
padding: 10px 16px;
// border: 1px solid #aaa;
border-top: 0;

/* border-bottom-left-radius: 6px;
border-bottom-right-radius: 6px;
*/
}

.subsectionText{
margin-bottom: 12px;
}
.subsectionMessage{
margin: 10px 0;
}

#topMessage{
// overflow-y: scroll;
overflow-y: auto;
height: 28em;
}
```

```
#topInfo{
//      overflow-y: scroll;
      overflow-y: auto;
      height: 8em;
}

.topblank8{
  margin-top: 8px;
}

.dl-horizontal dt{
  width: 250px;
}

.dl-horizontal dd{
  padding-left: 10px;
  margin-left: 250px;
}

#titleLinks{
  margin-left: 16px;
}

#titleLinks li a{
```

```
    display: block;
}
#titleLinks li a:before{
    content:
}
.require{
    color: #c00;
    font-size: small;
    font-weight: bold;
}

.appGenreList{
    display: none;
}

.btnSubmit{
    margin: 10px 0;
    text-align: center;
}

.ruleText{
    overflow-y: scroll;
    width: 100%;
}
```

```
    height: 10em;
    border: 1px solid #ccc;
    padding: 15px;
}

.modalText{
    margin-bottom: 10px;
}

.captcha{
    margin-top: 14px;
}

.captchaImg{
    border: 1px solid #ccc;
}

#kakuninMsg{
    display: inline;
}

#uploadMsg{
    display: none;
}
```

```
#kakuninButtons{
    display: block;
}

#uploadProgress{
    display: none;
}

#upPgsBar {
    width: 0%;
}

.statusList > li{
    margin-bottom: 10px;
}

.statusList table th{
}

#lock{
    z-index: 9999;
    display: none;
}

#footer{
```

```
margin-top: 10px;  
padding-top: 10px;  
border-top: 1px solid #ccc;  
color: #999;  
font-size: small;  
}
```