

Lab 2: Kết nối HTTP trong Android



MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

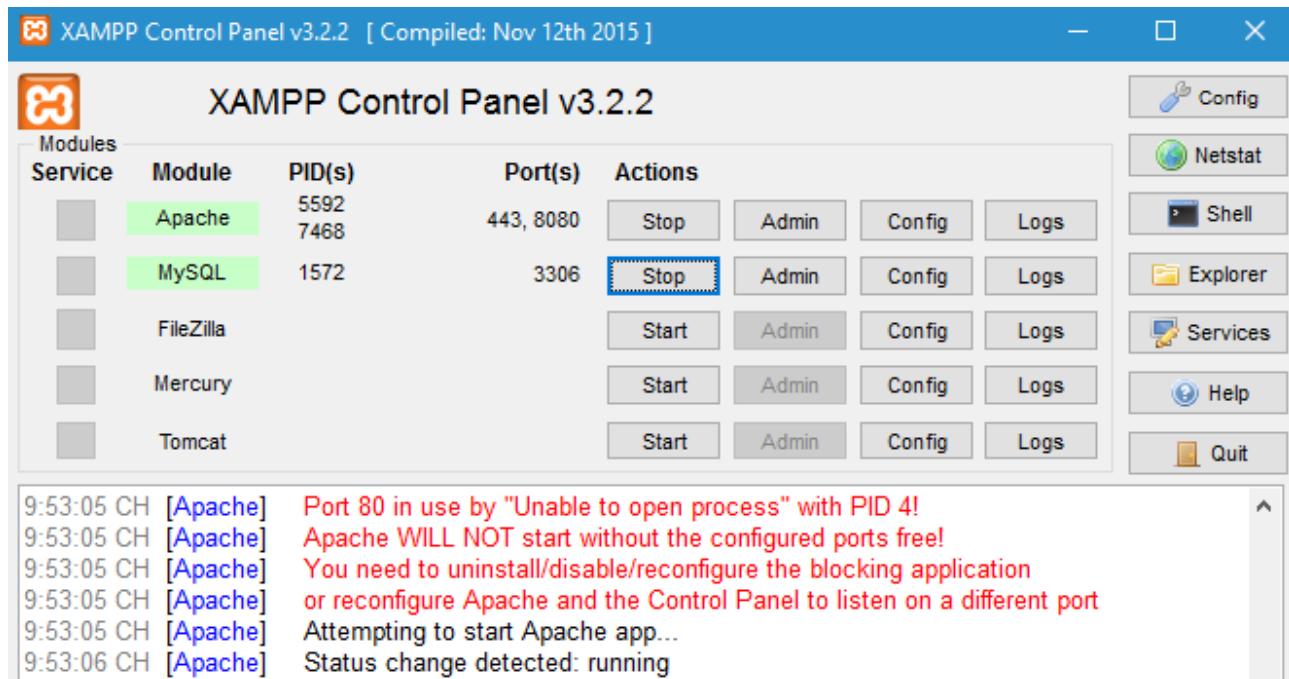
- ✓ Kết nối HTTP Giữa Client (Android) và Server
- ✓ Phân biệt và sử dụng được 2 phương thức POST và GET
- ✓ Xây dựng và triển khai kết nối HTTP Giữa Android và Server

BÀI 1 (2 ĐIỂM)

Viết chương trình cho phép nhập họ và tên sinh viên, điểm trung bình từ bàn phím sau đó hiển thị kết quả ra textview. Sử dụng phương thức GET để kết nối server.

HƯỚNG DẪN:

- ✓ Bật xampp và tiến hành start server



FPT POLYTECHNIC

- ✓ Vào thư mục htdocs của xampp tại thư mục mới đặt tên hoten_massv và tạo file student_get.php

PC > Local Disk (C:) > xampp > htdocs > nguyenvana_ps0000			
Name	Date modified	Type	Size
student_GET.php	26/12/2016 9:57 CH	PHP File	0 KB

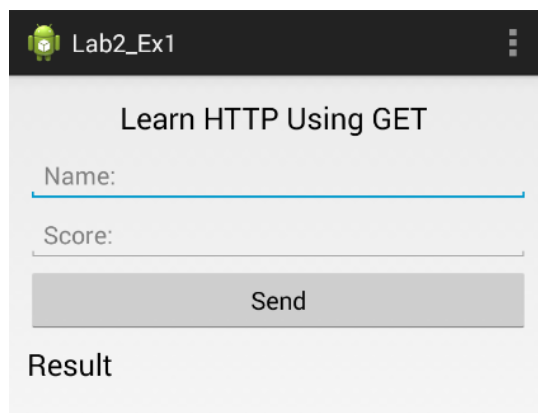
- ✓ Trong file student viết code đơn giản như sau

```

]<?php
    $name=$_GET['name'];
    $score=$_GET['score'];
    echo "Name: " . $name . "; Score: " . $score;
-?>

```

- ✓ Bây giờ mở Eclipse tạo mới ứng dụng android thiết kế giao diện như bên dưới



- ✓ Tạo biến static kiểu chuỗi chứa đường dẫn server lưu ý đường dẫn. Nếu không code sẽ không chạy.

```

public class MainActivity extends Activity implements OnClickListener {
    public static final String SERVER_NAME = "http://192.168.102.2:8080/android_server/student_GET.php";
}

```

- ✓ Tiến hành khai báo, anh xạ các view vừa tạo và bắt sự kiện Click cho button

```
private EditText edtName, edtScore;
private Button btnSend;
private TextView tvResult;

String strName, strScore;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    edtName = (EditText) findViewById(R.id.edtName);
    edtScore = (EditText) findViewById(R.id.edtScore);
    btnSend = (Button) findViewById(R.id.btnSend);
    tvResult = (TextView) findViewById(R.id.tvResult);
    btnSend.setOnClickListener(this);
}
```

- ✓ Tạo file java mới đặt tên BackgroundTask_GET. Khai báo các biến như sau để tương tác với view

```
public class BackgroundTask_GET extends AsyncTask<Void, Void, Void> {
    String duongdan = MainActivity.SERVER_NAME;
    TextView tvResult;
    String strName, strScore;
    String str;
    ProgressDialog pDialog;
    Context context;
```

- ✓ Tạo Constructor và truyền tham số giống với các biến vừa tạo

```
public BackgroundTask_GET(Context context, TextView tvResult, String strName, String strScore) {
    this.context = context;
    this.tvResult = tvResult;
    this.strName = strName;
    this.strScore = strScore;
}
```

- ✓ Viết code hiển thị ProgressDialog chờ trong hàm onPreExecute

FPT POLYTECHNIC

```
@Override
protected void onPreExecute() {
    // TODO Auto-generated method stub
    super.onPreExecute();
    progressDialog = new ProgressDialog(context);
    progressDialog.setMessage("Sending...");
    progressDialog.setIndeterminate(false);
    progressDialog.setCancelable(false);
    progressDialog.show();
}
```

- ✓ Xử lý code gửi và nhận dữ liệu từ server trong hàm doInBackground

```
@Override
protected Void doInBackground(Void... params) {
    // TODO Auto-generated method stub
    duongdan += "?name=" + this.strName + "&score=" + this.strScore;
    try {
        URL url = new URL(duongdan);
        HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();
        BufferedReader bfr = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()));
        String line = "";
        StringBuffer sb = new StringBuffer();
        while ((line = bfr.readLine()) != null) {
            sb.append(line);
        }
        str = sb.toString();
        urlConnection.disconnect();
    } catch (Exception e) {
        e.printStackTrace();
    }

    return null;
}
```

- ✓ Viết code gán dữ liệu cho textview

```
@Override
protected void onPostExecute(Void result) {
    // TODO Auto-generated method stub
    super.onPostExecute(result);
    if (progressDialog.isShowing()) {
        progressDialog.dismiss();
    }
    tvResult.setText(str);
}
}
```

- ✓ Xin quyền INTERNET

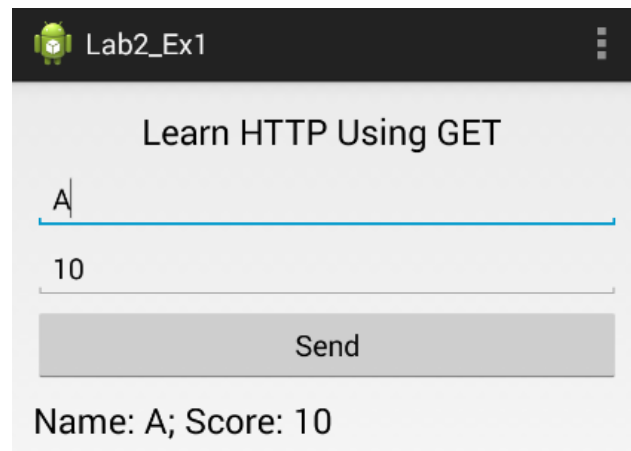
```
<uses-permission android:name="android.permission.INTERNET"/>
```

- ✓ Qua lại file MainActivity.java xử lý sự kiện button send

```
@Override
public void onClick(View v) {

    switch (v.getId()) {
        case R.id.btnSend:
            strName = edtName.getText().toString();
            strScore = edtScore.getText().toString();
            BackgroundTask_GET backgroundTask = new BackgroundTask_GET(this, tvResult, strName, strScore);
            backgroundTask.execute();
            break;
    }
}
```

- ✓ Chạy ứng dụng và xem kết quả như bên dưới



FPT POLYTECHNIC

BÀI 2 (2 ĐIỂM)

Viết chương trình nhập từ bàn phím 2 cạnh của hình chữ nhật. Dữ liệu được gửi lên server sau đó tính và trả chu vi, diện tích của hình chữ nhật về lại textview trên android. sử dụng phương thức POST kết nối.

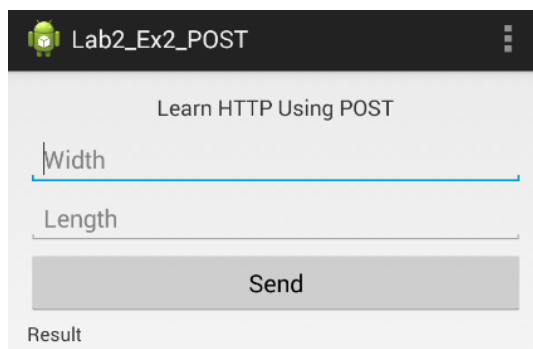
HƯỚNG DẪN:

- ✓ Chu vi = $(dai + rong) * 2$
- ✓ Diện tích = $dai * rong$
- ✓ Tương tự bài 1 tạo mới 1 file đặt tên là rectangle_POST.php trong thư mục htdocs và viết code đơn giản như sau.

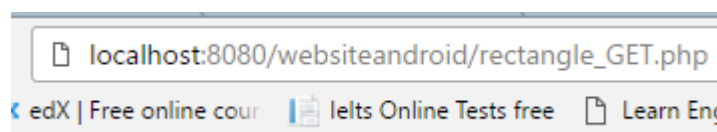
```
<?php
    $rong=$_POST['chieurong'];
    $dai=$_POST['chieudai'];
    $chuvi = ($dai + $rong) * 2;
    $dientich = $dai * $rong;

    echo "Chu vi: " . $chuvi . "; dientich: " . $dientich;
```

- ✓ Tạo mới project và thiết kế giao diện như bên dưới



- ✓ Lên trình lấy file server và sửa lại localhost cho phù hợp



- ✓ Khai báo biến static kiểu chuỗi chứa đường dẫn server

```
public class MainActivity extends Activity implements OnClickListener {
    public static final String SERVER_NAME = "http://192.168.102.2:8080/android_server/rectangle_POST.php";
```

- ✓ Khai báo và ánh xạ các view vừa tạo, bắt sự kiện onclick cho button

```
private EditText edtWidth, edtLength;
private Button btnSend;
private TextView tvResult;
String strWidth, strLength;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    edtWidth = (EditText) findViewById(R.id.edtWidth);
    edtLength = (EditText) findViewById(R.id.edtLength);
    btnSend = (Button) findViewById(R.id.btnSend);
    tvResult = (TextView) findViewById(R.id.tvResult);
    btnSend.setOnClickListener(this);
}
```

- ✓ Tạo mới file đặt tên BackgroundTask_POST và khai báo các biến xử lý với view như bên dưới

```
public class BackgroundTask_POST extends AsyncTask<Void, Void, Void> {
    String duongdan = MainActivity.SERVER_NAME;
    Context context;
    String strWidth, strLength;
    TextView tvResult;
    ProgressDialog pDialog;
    String strResult;
```

- ✓ Tiến hành tạo Constructor và truyền tham số giống với các biến vừa tạo

```
public BackgroundTask_POST(Context context, String strWidth, String strLength, TextView tvResult) {

    this.context = context;
    this.strWidth = strWidth;
    this.strLength = strLength;
    this.tvResult = tvResult;
}
```

- ✓ Viết code xử lý trong hàm onPreExecute

FPT POLYTECHNIC

```
@Override
protected void onPreExecute() {
    super.onPreExecute();
    progressDialog = new ProgressDialog(context);
    progressDialog.setMessage("Calculating...");
    progressDialog.setIndeterminate(false);
    progressDialog.setCancelable(false);
    progressDialog.show();
}
}
```

- ✓ Viết code xử lý việc truyền và nhận dữ liệu từ server trong hàm doInBackground như bên dưới

```
@Override
protected Void doInBackground(Void... params) {
    try {
        URL url = new URL(duongdan);
        String param = "chieurong=" + URLEncoder.encode(strWidth, "utf-8") + "&chieudai="
            + URLEncoder.encode(strLength, "utf-8");
        HttpURLConnection urlConnection = (HttpURLConnection)url.openConnection();
        urlConnection.setDoOutput(true);
        urlConnection.setRequestMethod("POST");
        urlConnection.setFixedLengthStreamingMode(param.getBytes().length);
        urlConnection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

        PrintWriter print = new PrintWriter(urlConnection.getOutputStream());
        print.print(param);
        print.close();

        String line = "";
        BufferedReader bfr = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()));
        StringBuffer sb = new StringBuffer();
        while ((line = bfr.readLine()) != null) {
            sb.append(line);
        }
        strResult = sb.toString();
        urlConnection.disconnect();
    } catch (Exception ex) {
    }
    return null;
}
```

- ✓ Viết code xử lý hiển thị kết quả cho textview trong hàm onPostExecute

```
@Override
protected void onPostExecute(Void result) {
    super.onPostExecute(result);
    if (progressDialog.isShowing()) {
        progressDialog.dismiss();
    }
    tvResult.setText(strResult);
}
}
```


FPT POLYTECHNIC

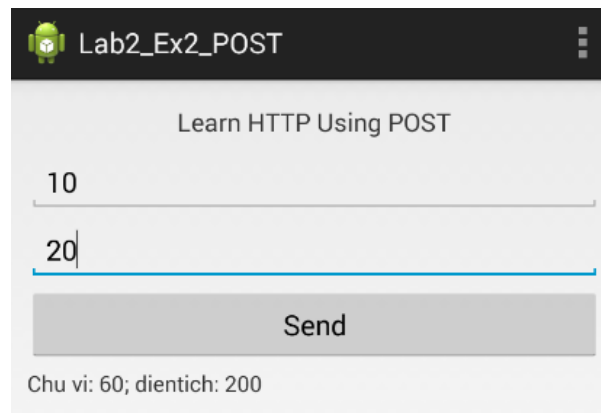
- ✓ Xin quyền INTERNET cho ứng dụng

```
<uses-permission android:name="android.permission.INTERNET"/>
```

- ✓ Mở lại file MainActivity.java Xử lý sự kiện click cho button

```
@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    switch (v.getId()) {
        case R.id.btnSend:
            strWidth = edtWidth.getText().toString();
            strLength = edtLength.getText().toString();
            BackgroundTask_POST backgroundTask_POST = new BackgroundTask_POST(this, strWidth, strLength, tvResult);
            backgroundTask_POST.execute();
            break;
    }
}
```

- ✓ Chạy ứng dụng và xem kết quả như bên dưới



BÀI 3 (2 ĐIỂM)

Viết chương trình nhập từ bàn phím cạnh của một khối lập phương. Dữ liệu được gửi lên server sau đó tính và trả thể tích của khối lập phương về lại textview trên android. Sử dụng phương POST kết nối.

HƯỚNG DẪN:

- ✓ Thể tích lập phương = cạnh * cạnh * cạnh
- ✓ Hoặc $\text{pow}(\text{cạnh}, 3)$
- ✓ Tương tự bài 1,2 ta tạo file php mới đặt tên là canh_POST.php viết code xử lý đơn giản như sau.

```
<?php
    $canh=$_POST['canh'];
    $thetich = pow($canh,3);

    echo "The tich " . $thetich;

?>
```

- ✓ Tạo project mới và xây dựng giao diện như bên dưới



- ✓ Khai báo biến static kiểu chuỗi chứa đường dẫn server tương tự bài 1,2 chú ý đường dẫn file phải chính xác.
- ✓ Khai báo anh xạ các view vừa tạo và bắt sự kiện click cho button.
- ✓ Tạo mới file đặt tên BackgroundTask_POST, khai báo các biến, gán tham số cho hàm dựng constructor

```
public class BackgroundTask_POST extends AsyncTask<String, Void, Void> {

    String duongdan = MainActivity.SERVER_NAME;
    Context context;
    TextView tvKetqua;
    String strKetqua;
    ProgressDialog pDialog;

    public BackgroundTask_POST(Context context, TextView tvKetqua) {
        // TODO Auto-generated constructor stub
        this.context = context;
        this.tvKetqua = tvKetqua;
    }
}
```

- ✓ Viết code xử lý trong hàm onPreExecute

```
@Override
protected void onPreExecute() {
    super.onPreExecute();
    pDialog = new ProgressDialog(context);
    pDialog.setMessage("Calculating...");
    pDialog.setIndeterminate(false);
    pDialog.setCancelable(false);
    pDialog.show();
}
```

FPT POLYTECHNIC

✓ Viết code xử lý truyền và nhận dữ liệu ở hàm doInBackground

```
@Override
protected Void doInBackground(String... params) {
    try {
        URL url = new URL(duongdan);
        String param = "canh=" + URLEncoder.encode(params[0].toString(), "utf-8");
        HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();
        urlConnection.setDoOutput(true);
        urlConnection.setRequestMethod("POST");
        urlConnection.setFixedLengthStreamingMode(param.getBytes().length);
        urlConnection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

        PrintWriter print = new PrintWriter(urlConnection.getOutputStream());
        print.print(param);
        print.close();

        String line = "";
        BufferedReader bfr = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()));
        StringBuffer sb = new StringBuffer();
        while ((line = bfr.readLine()) != null) {
            sb.append(line);
        }
        strKetqua = sb.toString();
        urlConnection.disconnect();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
    return null;
}
```

✓ Tiến hành gán và hiển thị kết quả cho textview

```
@Override
protected void onPostExecute(Void result) {
    super.onPostExecute(result);
    if (pDialog.isShowing()) {
        pDialog.dismiss();
    }
    tvKetqua.setText(strKetqua);
}
```

✓ Xin quyền INTERNET cho ứng dụng

✓ Xử lý sự kiện click cho button, khai báo và thực thi file vừa tạo

```
@Override
public void onClick(View v) {
    strCanh = edtCanh.getText().toString();
    bPost = new BackgroundTask_POST(this, tvKetqua);
    bPost.execute(strCanh);
}
```

✓ Chạy ứng dụng và xem kết quả



BÀI 4 (2 ĐIỂM)

Viết chương trình nhập các hệ số của phương trình bậc 2. Dữ liệu được gửi lên server sau đó tính delta và trả kết quả về lại textview trên android. Sử dụng phương thức POST kết nối.

HƯỚNG DẪN:

- ✓ $\Delta = \text{Math.pow}(b, 2) - 4 * a * c$
- ✓ Sử dụng $\text{Math.sqrt}(\Delta)$ để tính căn delta
- ✓ Tương tự bài 1,2,3 tạo file php mới đặt tên giaiphuongtrinh_POST.php và viết code xử lý như bên dưới

```
<?php
$a=$_POST['a'];
$b=$_POST['b'];
$c=$_POST['c'];

$delta = pow($b,2) - 4 * $a * $c;

$ketqua = "";
if($a == 0){
    if($b == 0){
        $ketqua = "Vo so nghiem";
    }else{
        $x1 = -$c/$b;
        $ketqua = "PT co 1 duy nhât: ".$x1;
    }
}else{
    if($delta < 0){
        $ketqua = "PT vo nghiem";
    }else{
        if($delta == 0){
            $x1 = -$b/(2*$a) ;
            $ketqua = "PT co nghiem kep: ".$x1;
        }else{
            $x1 = ((-$b+sqrt($delta))/(2*$a));
            $x2 = ((-$b-sqrt($delta))/(2*$a));

            $ketqua = "PT co 2 nghiem phan biet: x1= ".$x1."; x2= ".$x2;
        }
    }
}
echo "Ket qua " . $ketqua;
```

FPT POLYTECHNIC

- ✓ Tạo project android mới và xây dựng giao diện như sau



- ✓ Tạo biến static có kiểu chuỗi chứa đường dẫn server tương tự bài 1,2,3

```
public class MainActivity extends Activity implements OnClickListener {
    public static final String SERVER_NAME = "http://192.168.102.2:8080/android_server/giaiphuongtrinh_POST.php";
```

- ✓ Khai báo, ánh xạ các view vừa tạo, bắt sự kiện click cho button

```
private EditText edtA, edtB, edtC;
private Button btnSend;
private TextView tvResult;
String strA, strB, strC;
BackgroundTask_POST bPost;
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    edtA = (EditText) findViewById(R.id.edtA);
    edtB = (EditText) findViewById(R.id.edtB);
    edtC = (EditText) findViewById(R.id.edtC);
    btnSend = (Button) findViewById(R.id.btnSend);
    tvResult = (TextView) findViewById(R.id.tvResult);
    btnSend.setOnClickListener(this);
}
```

- ✓ Tạo file BackgroundTask_POST.java, khai báo các biến gán tham số cho hàm dựng như sau

FPT POLYTECHNIC

```
public class BackgroundTask_POST extends AsyncTask<String, Void, Void> {

    String duongdan = MainActivity.SERVER_NAME;
    Context context;
    TextView tvResult;
    String strResult;
    ProgressDialog pDialog;

    public BackgroundTask_POST(Context context, TextView tvResult) {
        // TODO Auto-generated constructor stub
        this.context = context;
        this.tvResult = tvResult;
    }
}
```

- ✓ Viết code xử lý ở hàm onPreExecute

```
@Override
protected void onPreExecute() {
    super.onPreExecute();
    pDialog = new ProgressDialog(context);
    pDialog.setMessage("Calculating...");
    pDialog.setIndeterminate(false);
    pDialog.setCancelable(false);
    pDialog.show();
}
```

- ✓ Viết code xử lý truyền và nhận từ server ở hàm doInBackground như sau

```
@Override
protected Void doInBackground(String... params) {
    try {
        URL url = new URL(duongdan);
        String param = "a=" + URLEncoder.encode(params[0].toString(), "utf-8") + "&b="
            + URLEncoder.encode(params[1].toString(), "utf-8") + "&c="
            + URLEncoder.encode(params[2].toString(), "utf-8");
        HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();
        urlConnection.setDoOutput(true);
        urlConnection.setRequestMethod("POST");
        urlConnection.setFixedLengthStreamingMode(param.getBytes().length);
        urlConnection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");

        PrintWriter print = new PrintWriter(urlConnection.getOutputStream());
        print.print(param);
        print.close();

        String line = "";
        BufferedReader bfr = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()));
        StringBuffer sb = new StringBuffer();
        while ((line = bfr.readLine()) != null) {
            sb.append(line);
        }
        strResult = sb.toString();
        urlConnection.disconnect();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
    return null;
}
```

- ✓ Trả kết quả về cho textview

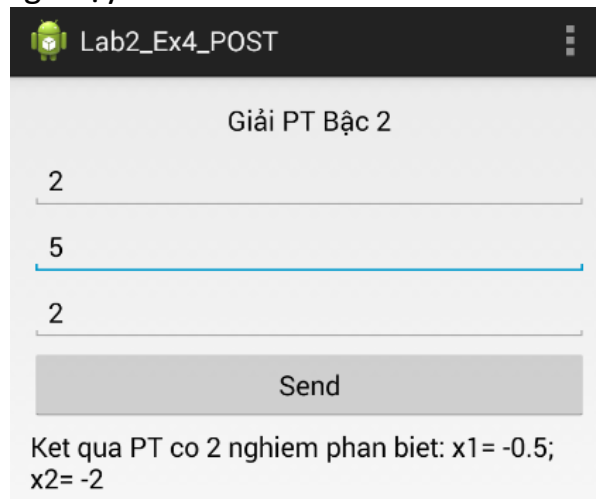
FPT POLYTECHNIC

```
@Override
protected void onPostExecute(Void result) {
    super.onPostExecute(result);
    if (pDialog.isShowing()) {
        pDialog.dismiss();
    }
    tvResult.setText(strResult);
}
```

- ✓ Xin quyền INTERNET cho ứng dụng
- ✓ Viết code xử lý sự kiện click cho button, thực thi việc kết nối

```
@Override
public void onClick(View v) {
    switch (v.getId()) {
        case R.id.btnSend:
            strA = edtA.getText().toString();
            strB = edtB.getText().toString();
            strC = edtC.getText().toString();
            bPost = new BackgroundTask_POST(this, tvResult);
            bPost.execute(strA, strB, strC);
            break;
    }
}
```

- ✓ Kết quả ứng dụng chạy như bên dưới



BÀI 5 (2 ĐIỂM):

Giảng viên cho thêm