PROG3150 Lecture 4

Mobile Application Development

Rick Kozak
Fall 2012

Agenda

- HTTP Requests
- XML parsing

Geocode REST Api

http://geocoder.ca/?latt=+43.39198&longt=-80.40544&corner=1&geoit=xml&range=&reverse=1


```
<result>
  <geodata>
    <latt>43.392075</latt><longt>-80.405865</longt>
    <city>Kitchener</city><prov>ON</prov><postal>N2P3M9</postal>
    <stnumber>350</stnumber><staddress>Doon Valley DR</staddress>
    <inlatt>43.39198</inlatt><inlongt>-80.40544</inlongt><distance>0.036</distance>
    <NearRoad>Doon Valley DR</NearRoad>
    <NearRoadDistance>0.017</NearRoadDistance>
    <betweenRoad1>Orchard Mill</betweenRoad1>
    <betweenRoad2>Old Mill</betweenRoad2>
    <intersection><street1>Doon Valley Dr</street1>
      <street2>Old Mill Rd</street2>
      <lattx>43.392436</lattx><longtx>-80.404169</longtx>
      <city>KITCHENER</city><prov>ON</prov><distance>0.114</distance>
    </intersection>
    <major_intersection><street1>Doon Valley Dr</street1>
      <street2>Old Mill Rd</street2>
      <lattx>43.3924358423</lattx><longtx>-80.4041694050</longtx>
      <city>KITCHENER</city><prov>ON</prov><distance>0.114</distance>
    </major_intersection>
  </geodata>
</result>
```

Android AsyncTask

```
private class Geocode extends AsyncTask <Location, Void,
                                          AddressData> {
  @Override
  protected AddressData doInBackground(Location... locs) {
  }

  @Override
  protected void onPreExecute(){
  }

  @Override
  protected void onProgressUpdate(Integer... progress){
  }

  @Override
  protected void onPostExecute(AddressData ad){
  }
}
```

Android onPreExecute

```
ProgressDialog pd;

@Override
protected void onPreExecute(){
        pd = new ProgressDialog(parent);
        pd.setProgressStyle(ProgressDialog.STYLE_SPINNER);
        pd.getWindow().setGravity(Gravity.BOTTOM);
        pd.setTitle("Getting address for current location");
        pd.setCancelable(false);
        pd.show();
}
```

Android doInBackground

```
@Override
protected AddressData doInBackground(Location... locs) {
  Location loc = locs[0];
  return getRevGeocode(loc.getLatitude(), loc.getLongitude());
}
```

Android onProgressUpdate

```
@Override
protected void onProgressUpdate(Integer... progress){
    pd.setProgress(progress[0]);
}
```

Android onPostExecute

```
@Override
protected void onPostExecute(AddressData ad){
  pd.dismiss();
  String s = ad.IntersectionRoad1 +
             " and " + ad.IntersectionRoad2 +
             ", " + ad.IntersectionCity + "," +
             ad.IntersectionProv;
  ((Button)findViewById(R.id.btnLocation)).setText(s);
}
```

Android getRevGeocode

```
public AddressData getRevGeocode(double lat, double lon){
  String url = String.format("http://geocoder.ca/?” +
                “latt=%f&longt=%f&corner=1&geoit=xml” +
                “&range=&reverse=1", lat, lon);

  ByteArrayInputStream bais = getHttpResponse(url);
  return parseAddressData(bais);
}
```

Android getHttpResponse

```
public ByteArrayInputStream getHttpResponse(String url){
  HttpClient hc = new DefaultHttpClient();
  HttpGet hg = new HttpGet(url);

  StringBuilder sb = new StringBuilder();
  HttpResponse hr = hc.execute(hg);
  HttpEntity he = hr.getEntity();

  if (he != null){
    InputStream is = he.getContent();
    InputStreamReader isr = new InputStreamReader(is);

    int len;
    char[] tmp = new char[2048];

    while ((len = isr.read(tmp))!= -1){
      sb.append(tmp, 0, len);
  }
  return new ByteArrayInputStream(sb.toString().getBytes());
}
```

Android parseAddressData

```
public AddressData parseAddressData(ByteArrayInputStream bais){
  DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
  DocumentBuilder builder = factory.newDocumentBuilder();
  Document dom = builder.parse(bais);
  Element root = dom.getDocumentElement();
  Element geodata = (Element)
                        ((root.getElementsByTagName("geodata")).item(0));

  ad.City = getXValue(geodata, "city");
  ad.Province = getXValue(geodata, "prov");
  ad.PostalCode = getXValue(geodata, "postal");
  ad.Number = getXValue(geodata, "stnumber");
  ad.Address = getXValue(geodata, "staddress");
  ad.NearRoad = getXValue(geodata, "NearRoad");
  ad.BetweenRoad1 = getXValue(geodata, "betweenRoad1");
  ad.BetweenRoad2 = getXValue(geodata, "betweenRoad2");

  NodeList nl = ((Element)geodata).getElementsByTagName("intersection");
  if (nl != null){
    Element isct = (Element)nl.item(0);
    ad.IntersectionCity = getXValue(isct, "city");
    ad.IntersectionProv = getXValue(isct, "prov");
    ad.IntersectionRoad1 = getXValue(isct, "street1");
    ad.IntersectionRoad2 = getXValue(isct, "street2");
  }
}
```

Android getXValue

```
private String getXValue(Node e, String name){
  String s = "";
  NodeList nl = ((Element) e).getElementsByTagName(name);
  if (nl != null && nl.getLength() > 0){
    Node n = nl.item(0).getFirstChild();
    if (n != null && n instanceof CharacterData){
      CharacterData cd = (CharacterData)n;
      s = cd.getData();
    }
  }
  return s;
}
```

WP7 - Geocode

```
public class Geocode
{
  public delegate void OnGeoComplete(AddressData ad);
  public static OnGeoComplete onGeoComplete = null;

  public static void getRevGeocode(double lat, double lon,
                                   OnGeoComplete ogc)
  {
    onGeoComplete = ogc;
    string url = string.Format("http://geocoder.ca/?" +
                 "latt={0}&longt={1}&corner=1&geoit=xml" +
                 "&range=&reverse=1", lat, lon);
    HttpWebRequest wrq = (HttpWebRequest)WebRequest.Create(url);

    RequestState rs = new RequestState();
    rs.wrq = wrq;

    wrq.BeginGetResponse(new AsyncCallback(onHandleResponse), rs);
  }
}
```

WP7 - onHandleResponse

```
private static void onHandleResponse(IAsyncResult iar)
{
  try
  {
    RequestState rs = (RequestState)iar.AsyncState;
    rs.wrs = (HttpWebResponse)rs.wrq.EndGetResponse(iar);
    rs.s = rs.wrs.GetResponseStream();
    rs.s.BeginRead(rs.buf, 0, RequestState.BUFFER_SIZE,
                              new AsyncCallback(ReadCallBack), rs);
  }
  catch (WebException e)
  {
  }
}
```

WP7 - ReadCallback

```
private static void ReadCallBack(IAsyncResult iar)
{
    RequestState rs = (RequestState)iar.AsyncState;
    Stream s = rs.s;
    int read = s.EndRead(iar);

    if (read > 0)
    {
      rs.requestData.Append(
                        Encoding.UTF8.GetString(rs.buf, 0, read));
      s.BeginRead(rs.buf, 0, RequestState.BUFFER_SIZE,
                           new AsyncCallback(ReadCallBack), rs);
    }
    else
    {
      s.Close();
      AddressData ad = parseXML(rs.requestData.ToString());
      if (onGeoComplete != null)
        onGeoComplete(ad);
    }
}
```

WP7 - parseXML

```
private static AddressData parseXML(string xml){
  AddressData ad = new AddressData();
  XDocument d = XDocument.Parse(xml);
  XElement e = d.Element("result").Element("geodata");

  ad.City = getXValue(e, "city");
  ad.Province = getXValue(e, "prov");
  ad.PostalCode = getXValue(e, "postal");
  ad.Number = getXValue(e, "stnumber");
  ad.Address = getXValue(e, "staddress");
  ad.NearRoad = getXValue(e, "NearRoad");
  ad.BetweenRoad1 = getXValue(e, "betweenRoad1");
  ad.BetweenRoad2 = getXValue(e, "betweenRoad2");

  e = e.Element("intersection");
  if (e != null){
    ad.IntersectionRoad1 = getXValue(e, "street1");
    ad.IntersectionRoad2 = getXValue(e, "street2");
    ad.IntersectionCity = getXValue(e, "city");
    ad.IntersectionProv = getXValue(e, "prov");
  }
  return ad;
}
```

Blackberry 7

```
public class Geocode extends Thread {
    private GeocodeCallback gcb;
    private QualifiedCoordinates qc;

    Geocode(GeocodeCallback gcb, QualifiedCoordinates qc){
        this.gcb = gcb;
        this.qc = qc;
    }

     public void run(){
        String xml;
         try{
             xml = getXml();
         } catch (Exception e){
             xml = "";
         }

         gcb.geocodeReady(parseXml(xml));
     }
}
```

Blackberry 7 - getXml

```
private String getXml() throws IOException {
  String[] o = new String[2];
  o[0]=Double.toString(qc.getLatitude());
  o[1]=Double.toString(qc.getLongitude());
  String url = Formatter.formatMessage("http://geocoder.ca/?
      latt={0}&longt={1}&corner=1&geoit=xml&range=&reverse=1", o);
  HttpConnection c = (HttpConnection)Connector.open(url);
  int rc = c.getResponseCode();
  if (rc != HttpConnection.HTTP_OK)
      throw new IOException("HTTP response code: " + rc);
  InputStream is = c.openInputStream();
  StringBuffer x = new StringBuffer();
  int bytesread = 0;
  byte[] temp = new byte[2048];
  while (bytesread !=  -1) {
    bytesread = is.read(temp, 0, temp.length);
    if (bytesread > 0)
      x.append(new String(temp, 0, bytesread));
  }
  if (is != null) is.close();
  if (c != null) c.close();
  return x.toString();
}
```

Blackberry 7 - parseXml

Same as Android

BB10

```
QNetworkAccessManager *mNetworkAccessManager;

Void App::App(){
    mNetworkAccessManager = new QNetworkAccessManager(this);
    bool result = connect(mNetworkAccessManager,
            SIGNAL(finished(QNetworkReply*)),
            this, SLOT(requestFinished(QNetworkReply*)));
}

void App::initiateRequest(){
    QUrl url = Qurl(
            sprintf("http://geocoder.ca/?latt=%s&longt=
              %s&corner=1&geoit=xml&range=&reverse=1", latt, lont));
    QNetworkRequest request = QNetworkRequest();
    request.setUrl(url);
    mNetworkAccessManager->get(request);
}

void App::requestFinished(QNetworkReply* reply){
    if (reply->error() == QnetworkReply::NoError)
       parseXml(reply->readAll()); //using libxml2
}
```