INFORMATION SYSTEMS IN HEALTH CARE

Lesson 11 – Winter Term 2014

Agenda

1. **HL7**
2. **Android**
3. **Project**

Data Communication Standards

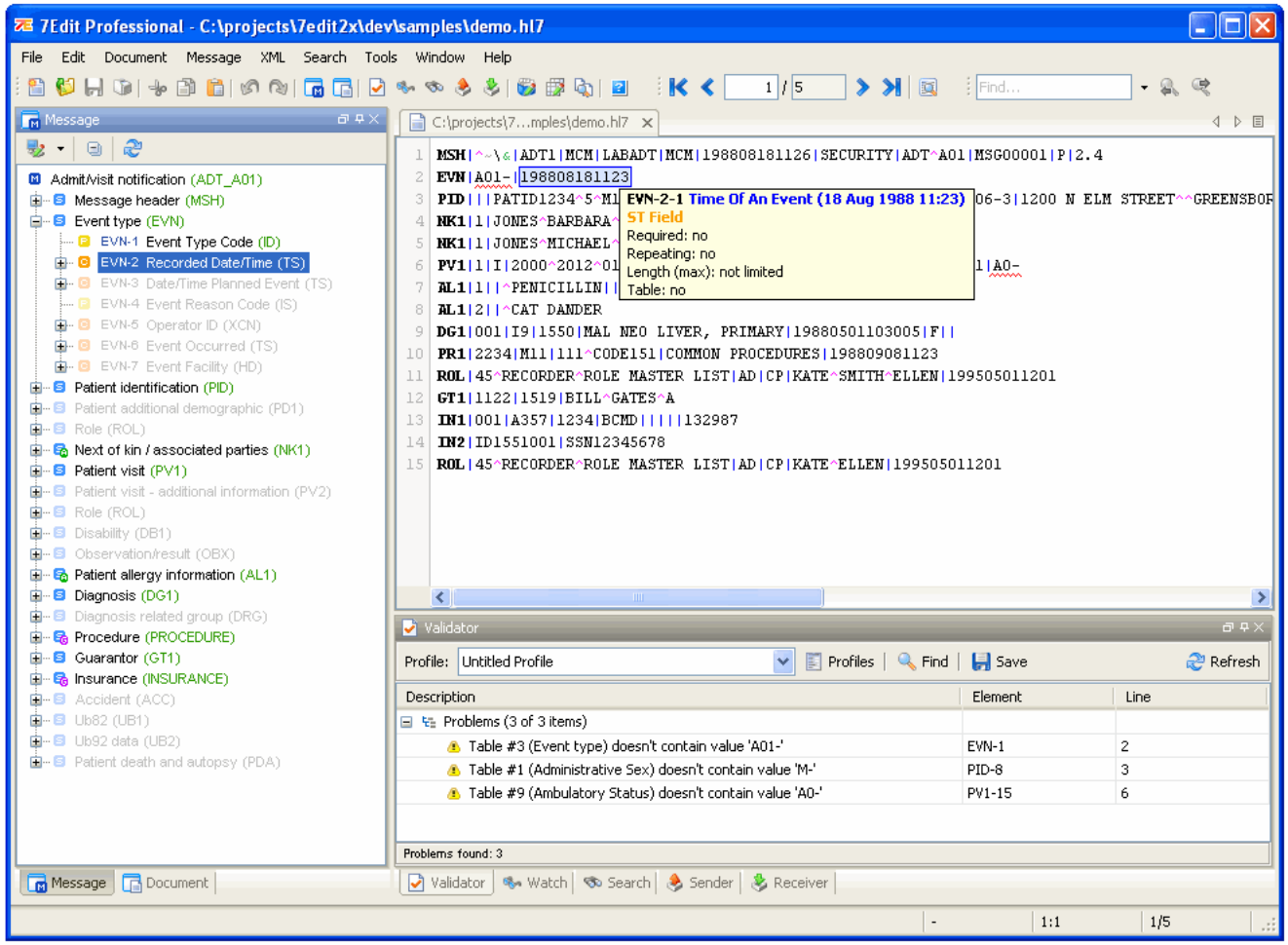
- Most patient information is stored in disparate systems across the healthcare community
- Health Level 7 (HL7) is an international standard for the transmission of medical data
- **MSH (Message Header) Key**
  - MSH|^~\&|FORMENTRY|AMRS|HL7LISTENER|AMRS|20050217152845||ORU^R01|?|P|2.5|1|||||||||1^AMRS-ELDORET^http://schemas.openmrs.org/2006/FormEntry/formId^URI
    - ^~\&
      - Encoding characters
      - ^ ? component separator
      - ~ ? repetition separator
      - \ ? escape character
      - & ? subcomponent separator
    - FORMENTRY
      - Sending Application
    - AMRS
      - Sending Facility
    - HL7LISTENER
      - Receiving Application
    - AMRS
      - Receiving Facility
    - 20050217152845|
      - Date/Time of Message (YYYYMMDDHHMMSS)
    - Security (not necessary)
    - ORU^R01
      - Message Type (ORU = Unsolicited Transmisstion of an observation message)

Data Communication Standards

- **PID (Patient Identification) Key**
  - PID|||1MT^0^M10||Patient^Jonny^Dee^DR|Patient^Momma^Thee^MS|20040101000000|M||B|555 Johnson Road^Apt. 555^Indianapolis^IN^46202^USA||||||||||Indianapolis, IN|||||||||||||||||TRIBE CODE
    - Set-ID, Patient ID, Alternate Patient ID
    - Patient^Jonny^Dee^DR
      - Family Name (Patient)
      - Given Name (Jonny)
      - Second / Middle Name (Dee)
      - Suffix ()
      - Prefix (DR)
    - 20040101000000
      - Date/Time of Birth (YYYYMMDDHHMMSS)
    - M
      - Administrative Sex
- **PV1 (Patient Visit) Key**
  - PV1||1|O|^^^^^MTRH^1^AMRS|2|||1^Mamlin^Joseph^^^^^^^8^M10^^AMRS|||||||||||||||||||||||||||||||||||||||||20050217140000||||||V
- **ORC (Common Order Segment)**
  - ORC|RE|||||||20050221130000|1^Enterer^Ima^^^^^AMRS
- **OBX / NM (Observation Result, Numeric Datatype) Key**
  - OBX|3|NM|5089^WEIGHT (KG)^DCT||25|kg|20-300|L|||F|||20050217204000

Data Communication Standards

- most patient information is stored in disparate systems across the healthcare community
- Health Level 7 (HL7) is an international standard for the transmission of medical data
- **PID (Patient Identification) Key**
  - PID|||1MT^0^M10||Patient^Jonny^Dee^DR|Patient^Momma^Thee^MS|200401 01000000|M||B|555 Johnson Road^Apt. 555^Indianapolis^IN^46202^USA|||||||||||Indianapolis, IN|||||||||||||||TRIBE CODE
    - Set-ID, Patient ID, Alternate Patient ID
    - Patient^Jonny^Dee^DR
      - Family Name (Patient)
      - Given Name (Jonny)
      - Second / Middle Name (Dee)
      - Suffix ()
      - Prefix (DR)
    - 20040101000000
      - Date/Time of Birth (YYYYMMDDHHMMSS)
    - M
      - Administrative Sex

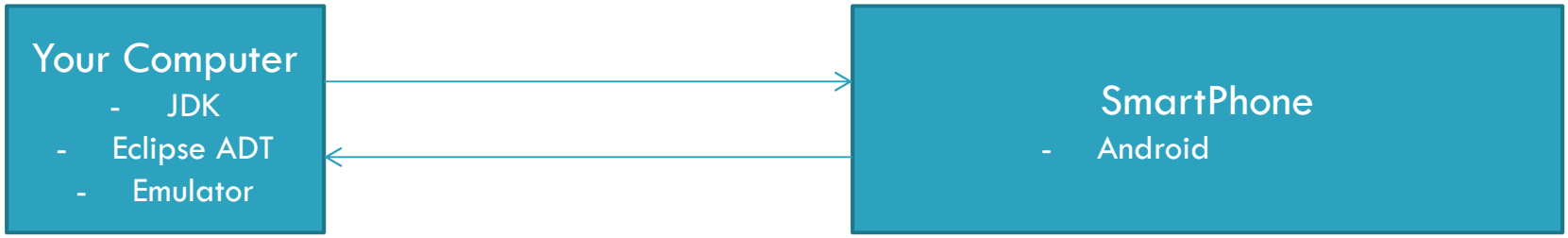


http://7edit.com

Introduction to Android

- Android is a mobile operating system based on the Linux kernel and currently developed by Google.
- Android is designed for touchscreen mobile devices such as smartphones, tablets, televisions, wrist watches
- Android is the most widely used mobile OS and, as of 2013, the highest selling OS overall.

Development environment



1. Java Development Kit (JDK)
2. Eclipse ADT
3. Emulator

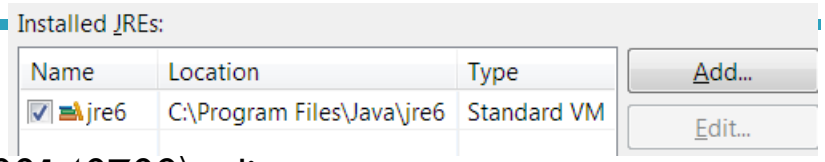
Java Development Kit

- Current version
  - Check the current version of Java with the command line
    - java –version
- Installation
  - Donwload JDK6 (http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase6-419409.html#jdk-6u45-oth-JPR)
    - Registration on oracle.com is obligatory
    - You need the 32bit version
    - Install JDK6, e.g. in C:\Program Files\Java\jdk1.6.0_45
    - When prompted, install JRE6, e.g. in C:\Program Files\Java\jre6

Eclipse ADT

- Installation of ADT
  - Donwload Eclipse ADT (http://developer.android.com/sdk/index.html)
    - Version 32bit is needed
    - It is a zipped file
  - Unzip Eclipse ADT, e.g. to C:\adt-bundle-windows-x86-20140702
- Installation of the emulator
  - Open the folder C:\adt-bundle-windows-x86-20140702\eclipse
  - Start eclipse.exe
  - Click on **SDK Manager** 
  - Click on ‚Deselect All‘
  - Select ARM EABI under Android 4.4.2 (API19)
  - Click on Install 1 package, then Accept and Install
  - At the end, restart Eclipse

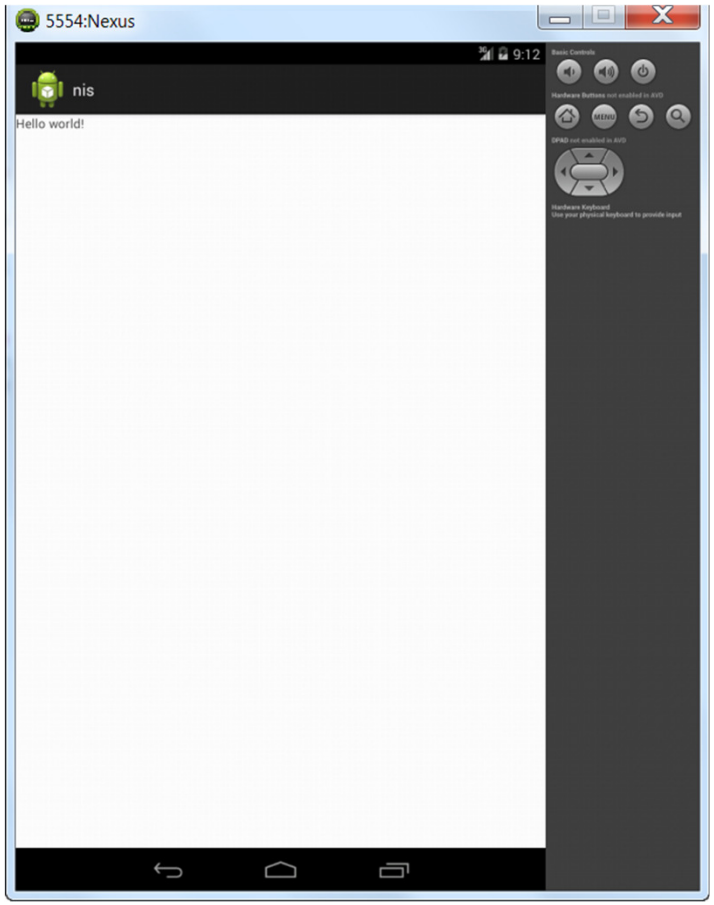
Eclipse - Configuration



- Configuration Eclipse
  - Open the folder C:\adt-bundle-windows-x86-20140702\eclipse
  - Start eclipse.exe
  - Enter the path to a workspace as an empty or new folder, e.g. C:\dev\nis
  - In Menu Windows-Preferences-Java-Installed JRE, add the JRE6 folder
- Configuration of the emulator
  - Click on the **Android Virtual Device Manager** 
  - Click on Create
  - Endter AVD name Nexus
  - Choose Device Nexus 7, Target Android 4.4.2, CPU Android ARM
  - Enter Hardware keyboard, Skin with dynamic hw control, RAM 1024, Internal storage 512, SD Card 512
  - Click on Start
    - It can take a while for Android to start on the emulator

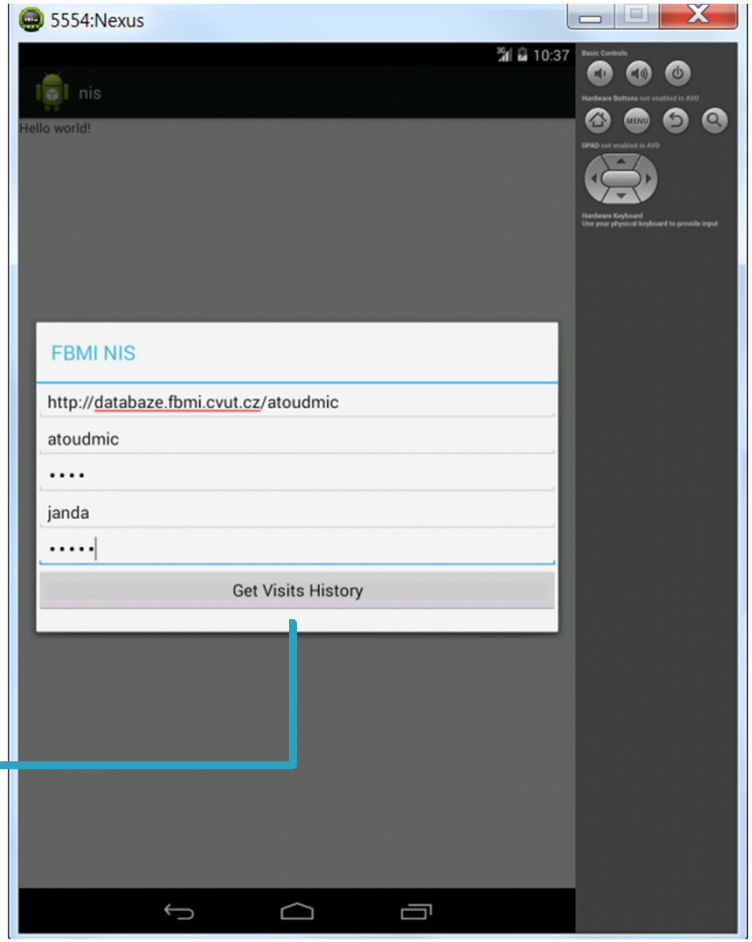
Eclipse – First project

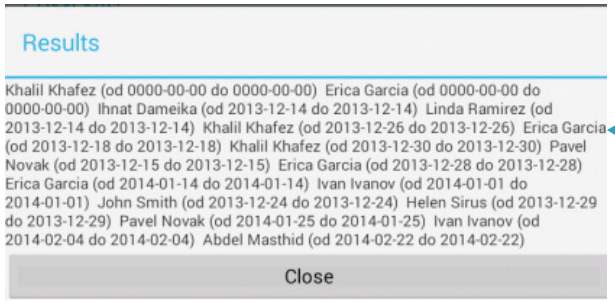
- Your first project
  - Click File – New – Android Application Project
  - Enter Application name HelloWorld, Min API 9, Target API 19, Compile with API 19
  - Enter Package name com.example.helloworld
  - In the last page, select Empty Activity
  - Leave the default settings
- Test your project
  - Open src - com.example.helloworld - MainActivity.java
  - Click on Run
  - Choose Android Application
  - Choose Nexus emulator
  - (Eclipse allows to run a project on an Android phone connected via USB as well, see http://developer.android.com/tools/extras/oem-usb.html)



Eclipse – base project health care IS

- Prepare the Eclipse workspace for your IS
  - Unzip NIS_Android.zip, which was sent to you via email
  - File – Import – Android – Existing Android Code into Workspace
  - Browse – choose the folder wher you unzipped NIS_Android.zip
  - Choose the project nis (do not select appcompat_v7)
  - Check copy project into workspace
  - Click on Finish (a new project nis will appear in the workspace)
- Prepare the web server
  - With a text editor, update your database credentials in getDataForAndroid.php
  - Copy getDataForAndroid.php to your web server
- Test your IS project
  - Open nis – src – cz.cvut.fbmi.nis - MainActivity.java
  - Click on Run





Eclipse – orientation in the workspace

- Structure of the workspace
  - There are three projects: HelloWorld (your first Android projekt), nis (health care IS project), appcompat_v7 (library project for backward compatibility with older versions of Android devices)
- Code structure - src – cz.cvut.fbmi.nis
  - MainActivity.java
    - Main page of the application
    - Initialize GUI and data
- Structure of the user interface – res - layout
  - activity_main.xml
    - Layout of the main page
  - login.xml
    - Layout of the page for entering login information
  - Results.xml
    - Layout of the results page

Project NIS – MainActivity.java

- Dialog for login
  - final Dialog dialog = new Dialog(MainActivity.this);
  - dialog.setContentView(R.layout.login);
  - dialog.setTitle("FBMI NIS");
  - dialog.show();

Login.xml

```
<EditText
    android:id="@+id/editTextUserNameToLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Doctor user name"
    android:ems="10" >
</EditText>

<EditText
    android:id="@+id/editTextPasswordToLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPassword"
    android:hint="Doctor password"/>

<Button
    android:id="@+id/buttonSignIn"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Get Visits History" />
```

Project NIS – MainActivity.java

- Button *Get Visits History*
  - Button btnSignIn=(Button)dialog.findViewById(R.id.buttonSignIn);
  - Data input
    - final EditText editTextUserName=(EditText)dialog.findViewById(R.id.editTextUserNameToLogin);
    - final EditText editTextPassword=(EditText)dialog.findViewById(R.id.editTextPasswordToLogin);
    - String doctorUserName=editTextUserName.getText().toString();
    - String doctorPassword=editTextPassword.getText().toString();
  - Dynamic
    - btnSignIn.setOnClickListener
      - String visitsHistory = getVisitsHistory(doctorUserName, doctorPassword, facultyUserName, facultyPassword,facultyUrl);

Login.xml

```
<EditText
    android:id="@+id/editTextUserNameToLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Doctor user name"
    android:ems="10" >
</EditText>

<EditText
    android:id="@+id/editTextPasswordToLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPassword"
    android:hint="Doctor password"/>

<Button
    android:id="@+id/buttonSignIn"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Get Visits History" />
```

Project NIS – MainActivity.java - getVisitsHistory

- HTTP communication
  - HttpPost httppost = new HttpPost(facultyUrl + "/getDataForAndroid.php?log=" + doctorUserName + "&pwd=" + doctorPassword);
  - httppost.addHeader(BasicScheme.authenticate(new UsernamePasswordCredentials(FacultyUserName, FacultyPassword), "UTF-8", false));
  - HttpResponse response = httpclient.execute(httppost);
  - String data = inputStreamToString(response.getEntity().getContent()).toString();

http://database.fbmi.cvut.cz/vaslogin/getDataForAndroid.php

```
<?php

$login = $_REQUEST['log'];
$password = $_REQUEST['pwd'];
$query = "SELECT patient.name patient_name, visit.start visit_start, visit.end visit_end FROM user, visit, patient " ."WHERE user.user_id = visit.doctor_id " . "AND
visit.patient_id = patient.id " . "AND user.user_login ='" . $login . "' " . "AND user.user_password = '" . $password . "'";
$result = mysql_query($query);
if (!$result) { echo "Could not execute query: $query";}
$data = '';
while($row=mysql_fetch_assoc($result)){
   $data=$data . $row['patient_name'] . " (od " .   $row['visit_start'] . " do " . $row['visit_start'] . ") \n ";
}
mysql_close($con);
echo $data;
?>
```

Project NIS – MainActivity.java

- Dialog for displaying results
  - final Dialog dialogResults = new Dialog(MainActivity.this);
  - dialogResults.setContentView(R.layout.results);
  - dialogResults.setTitle("Results");
  - final TextView textViewResult=(TextView)dialogResults.findViewById(R.id.textViewResult);
  - textViewResult.setText(visitsHistory);

Results.xml

```
<TextView
    android:id="@+id/textViewResult"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:singleLine="false"
    android:maxLines="100"
/>

<Button
    android:id="@+id/buttonClose"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Close" />
```

Syllabus of lectures and tutorials

		Lectures (45 min)	Tutorials (45 min)
Lesson 1	Sep 30	Medical Informatics and IS definition	OpenEMR
Lesson 2	Oct 7	HW infrastructure of IS	OpenEMR
Lesson 3	Oct 14	Operation systems	GaiaEHR
Lesson 4	Oct 21	Databases of IS	SQL
~~Lesson 5~~	~~Oct 28~~		
Lesson 6	Nov 4	Clinical oriented IS	SQL
Lesson 7	Nov 11	Decision support systems Medical data coding	OpenMRS
Lesson 8	Nov 18	Phase and IS development principles	UML
Lesson 9	Nov 25	Standard implementation methodology	Programing in HTML and PHP
Lesson 10	Dec 2	Standard implementation methodology	Programing in PHP and MySQL
Lesson 11	Dec 9	Data and communication standards	Android, 7Edit
Lesson 12	Dec 16	Final exam	
Lesson 13	Jan 20	Presentation of practical project	

Requirements for grade

- Attendance to ALL lessons
  - In case of non-attendance, provide a valid reason
- 30 points from homework
  - 1 homework per week
  - 3 points per homework
  - The homework is an essay about the topics covered in the lecture
- 20 points from practical projects
  - 4 points for analyzing your own information system
  - 10 points for implementing your own information system
  - 4 points for testing your own information system
  - 2 points for presenting your own information system
- 50 points from final exam
  - 30 points about the lectures
  - 20 points about the tutorials