Lec #19: Conclusion

Tamer Nadeem
Dept. of Computer Science
Objective

• MidTerm Statistics

• Publishing your application

• Next Course
Publish to Android Market
The Checklist

• Have I tested my application extensively?
  ➤ Emulator + Physical device
  ➤ Multiple hardware devices running different Android versions.

• Does my application perform well?
  ➤ Performance is really important especially in games

• Have I decided on SDK compatibility?
  ➤ Android 2.2 & 2.3 dominate.
Getting your application ready

• Step 1: Request necessary Android permissions
  ➤ Make sure that you’re requesting all necessary permissions, otherwise your application won’t work.

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

• Step 2: Specify a name and icon
  ➤ Name your application and give it an icon using the `android:label` and `android:icon` attribute in the application tag

```xml
<application android:label="@string/app_name" android:icon="@drawable/myIcon"/>
```
Getting your application ready

• Step 3: Configure version manifest data
  ➤ Pick a version your application using android:versionCode and android:versionName.
  ➤ versionCode is an integer that must increment for every update.
  ➤ versionName is a user-friendly value (e.g., 0.1 or 1.0b or 2.4.1)

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.example" android:versionCode="1"
  android:versionName="1.0.0">
Getting your application ready

• Step 4: Set compatibility options
  ➤ If you’re utilizing Android features that aren’t available in older versions → specify a set of version requirements within `uses-sdk`.
  ➤ `android:minSdkVersion` The minimum Android platform API level on which your application will be able to run.
  ➤ `android:targetSdkVersion` The API level that your application was designed to run on.
  ➤ `android:maxSdkVersion` An upper limit for compatibility. Don’t set this unless you have a very good reason to.

• Step 5: Cleanup files and remove logging
Getting your application ready

• Step 6: Sign and ZIP-align your application
  ➤ Android applications must be digitally signed with a certificate that the developer holds to ensure the authenticity.
  ➤ Pick a strong password for your private key and ensure to keep it safe
  ➤ Eclipse by default signs compiled apps with debug key.
  ➤ Use Export Wizard:
    1. Select the project and select File > Export.
    2. Open the Android drop-down and select Export Android Application
    3. Follow the wizard’s steps to compile, sign and ZIP-align your application.

Create a new one for the first time or use an existing one.

Your validity period must extend 2033 or else the Android Market will reject your app!
Becoming a Market publisher

• Registration

  1. Register as a publisher and setup your profile.
     ➤ [http://market.android.com/publish](http://market.android.com/publish) and sign in with your Google account
     ➤ Fill out all the required information along with your real phone number
     ➤ Note: you can always change “developer name” later via your profile page

  2. Read and agree to the Android Market Developer Distribution Agreement.

     ➤ Click the Google Checkout button and pay the one-time registration fee
Becoming a Market publisher

• Uploading an application
  ➤ Login to your publisher account and click “Upload an Application”.
  ➤ Fill in all of the form data and include screenshots if you can and provide a precise, short and well-formatted description.
  ➤ You can also choose whether or not you want to release it as a free or paid app.
  ➤ Click “Publish.”

Congratulations, you’ve just successfully published an application to the Android Market!
Wireless Networking / Mobile Computing CS 752/852
The Future

RFID and Sensor Networks

Citywatchers, Walmart, Intel, Philips, Bosch ...

Personal Area Networks
Motorola, Intel, Samsung ...

Mesh Networks and Wireless Backbones
Microsoft, Intel, Cisco ...

Internet
Why are wireless networks different?

Here are a few reasons

• open communication medium
• radio signals spread through the environment in contrast with wired communications
• user mobility – changing topology
• lack of centralized control
• mobile users are easily compromised
• cooperative/distributed algorithms and protocols

Protocols developed for wired networks often do not apply to wireless

• Unreliable and Unpredictable Wireless Coverage - vary over time and space
• Open Wireless Medium – Interference, Hidden Node, Exposed Nodes, Security
• Mobility - poor-quality wireless links, intermittent connection, changes context
• Portability - Limited battery power, Limited processing, display and storage
But, what does it take for that mobile/wireless future to become feasible?
Enabling wireless ubiquity. Showing what is feasible, and what is not ...

Applications that exploit ubiquity and mobility. Challenges underlying such applications
Research Challenges

- **PHY**
  - MAC / Link
  - Network
- **Transport**
- **Security**
- **Application**

**Wireless Networking**
- Ubiquitous Services
- Incentives
- Loss Discrimination
- Energy Savings
- Spatial Reuse

**Mobile Computing**
- Privacy
- Eavesdropping
- Mobility
- Interference Mgmt.
- Channel fluctuations

**Ubiquitous Services**

**Interference Mgmt.**
CS 752/852 Course

- Introduces fundamentals and applications of wireless and mobile networking
  - The departure from wired networks …
  - Emerging innovations in EE, communications area

- Exposes implications on protocol design
  - At MAC, Network, Transport, Security
  - Investigates gap between idea and actual system
  - Considers theoretical aspects

- Envisions new mobile computing applications
  - Identifies challenges underlying them
  - Resolves these challenges into a full system solution
At the End of this Course …

• You understand
  • Physical layer (radios, rate, antennas, channels)
  • MAC protocols (who gets the chance to talk)
  • Routing (path selection algorithms and issues)
  • Reliability (wireless congestion control, rate control)
  • Applications (social networks, personal networks, P2P)
  • Human sensing, Urban sensing
    • Localization (extracting the location of a device)
    • Mobility (how it helps and disrupts communication)
    • Interfaces (phones are more than communication devices)
    • Privacy (how to protect a user from being tracked)
  • Energy-awareness (how it percolates various network functions)
  • Emerging Topics (interference cancellation, multicast, rural nets)
  • Capacity (what is feasible, what are performance bounds)
What this Course Does Not Cover

• Not a wireless communications course

• Does not cover
  • Modulation schemes
  • Transmitter/Receiver design
  • Signal processing and antenna design
  • Source coding / channel coding
  • Etc.

• This is course on
  • Design, analysis, and implementation of protocols and algorithms in (mobile) wireless network systems
Course Workload

• Class presentations and open discussions
  ➤ Each student will be required to present 2-3 papers in the entire semester

• 1 mid term, No Final Exam
  ➤ Tentative date of mid-term: Mid march after Spring break

• Survey paper
  ➤ Each student prepares a survey paper on an interesting topic to you (suggested list will be provided by me)

• Semester-long class project
  ➤ In groups of 2 (max 3).