Distributed Systems – Assignment 1

Marian George

Marian.george@inf.ethz.ch

The Exercise

- Objectives
  - Get familiar with Android programming
    - Emulator, debugging, deployment
  - Learn to use UI elements and to design an Activity
  - Learn how to connect Activities and Services using Intents
  - Learn how to use the Sensor API
  - Tackling problems with developing a real app

- Dates:
  - Exercise begins: Now
  - Exercise is due: 9:00am, October 14, 2013

The Exercise

- **Task 1:** Sensors and Actuators
  - Create an application to access all available sensors
  - Use selected actuators
- **Task 2:** AntiTheft Alarm
  - Create an application to *"secure"* the device against theft
- **Task 3:** Enhancements
  - Come up with creative solutions to deal with shortcomings of the simple alarm

Task 1: Sensors and Actuators

- **Objectives:**
  - Learn how to create an Android project
  - Familiarize yourself with UI Elements
  - Understand the concept of Activities and Intents
  - Learn to use the sensor API
- **Todo:**
  - Write an app that displays all available sensors in a `ListView`
  - Show sensor readings in a second activity
  - Trigger actuator events in a third activity

Hints

- Project names: VS_*nethz*_Sensors (leader’s nethz ID)
- Do not forget to add all components (Activities, Services) to the application in the manifest file
- Do not forget the permissions in the manifest file
- Listing all the sensors:

```
private SensorManager sensorMgr;
private List <Sensor> sensors;
```

```
sensorMgr = (SensorManager) getSystemService(SENSOR_SERVICE);
sensors = sensorMgr.getSensorList(Sensor.TYPE_ALL);
```

- Also see: http://developer.android.com/guide/topics/sensors/sensors_overview.html

Hints

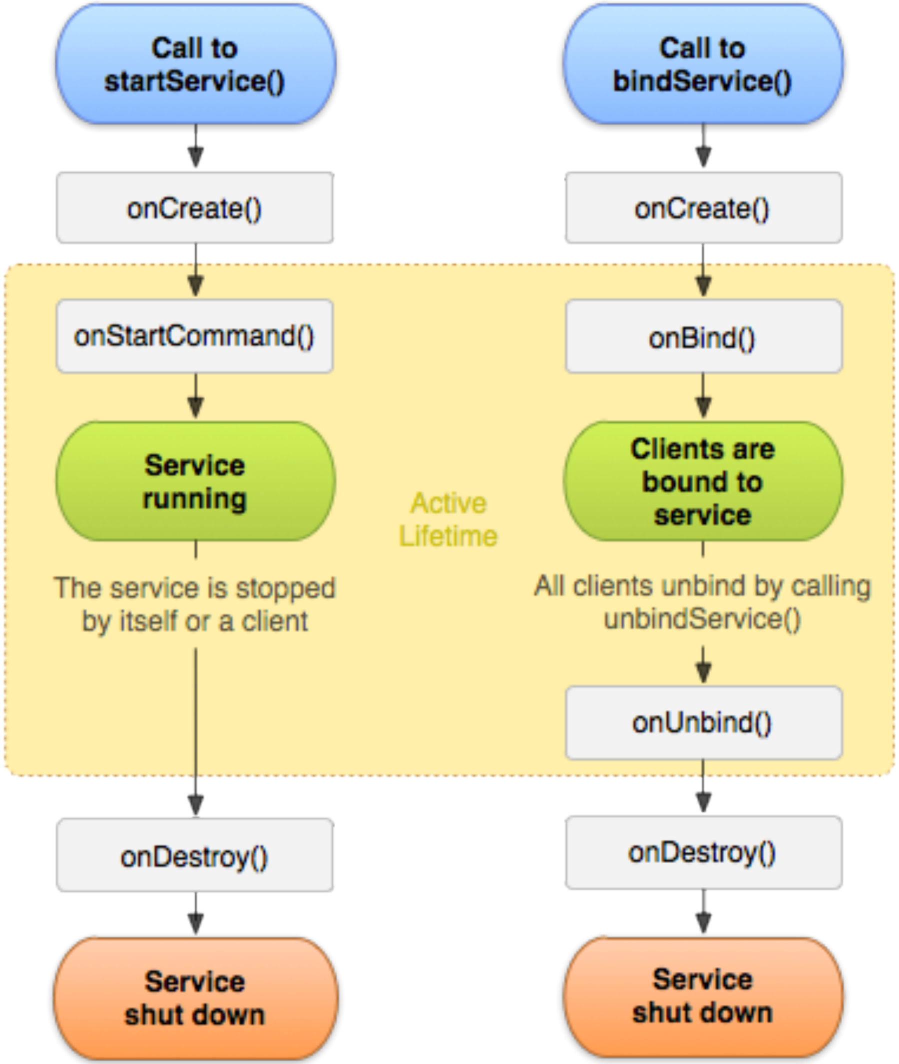
- When starting `SensorActivity`, the Intent should carry the information which sensor to display in detail
- `SensorActivity` should implement the `SensorEventListener` interface and continuously present the sensor's value(s)
- Check the `ArrayList<String>` and `ArrayAdapter<String>` classes, as they are useful to hold the sensor values as `Strings` for the `ListView`
- Also see: http://commonsware.com/Android/excerpt.pdf

Task 2: AntiTheft Alarm

- **Objectives:**
  - Transfer the knowledge of Task 1 into a real app
    - Understand problems stemming from a framework under development
  - Think about how to make use of the sensors
  - Learn how to use background Services
- **Todo:**
  - Write an app to *"secure"* the device against theft
    - Sound an alarm when the device is moved without authorization
  - The app MUST implement the code skeleton provided to you on the course website
    - AntiTheftService { startAlarm(); }
    - AbstractMovmentDetector { setCallbackService(AntiTheftService);
      onSensorChanged(SensorEvent); }

Hints

- Study service lifecycle
- A **Started** Service (unbound service) is good for us



unbound service ***bound service***

http://developer.android.com/guide/components/services.html

Hints

- We know that this service always runs in the same process as its clients, we don't need to deal with inter-process communication
- See the **LocalService.java** and the **ServiceStartArguments.java** examples
- Do not forget to declare the Service in the manifest file!

Task 3: Enhancements

- Visualize sensor readings with Graphics API

- Problems with Task 2 include:
  - Headphones can suppress alarm
- Possible solutions:
  - Silent alarm (SMS, e-mail)
  - Send GPS coordinates
  - (BroadcastReceiver for headphones)
- **Todo:**
  - Come up with a creative solution

Designing Compelling and Usable Applications

- Purpose of the app
  - How does the app make life better?
  - Understand your user!
- Context of the app
  - Target user group: student/child/elderly
  - Activity: sitting/walking/driving/etc.
  - Environment: office/street/car/home/etc.
  - Device: phone/tablet/coffee machine

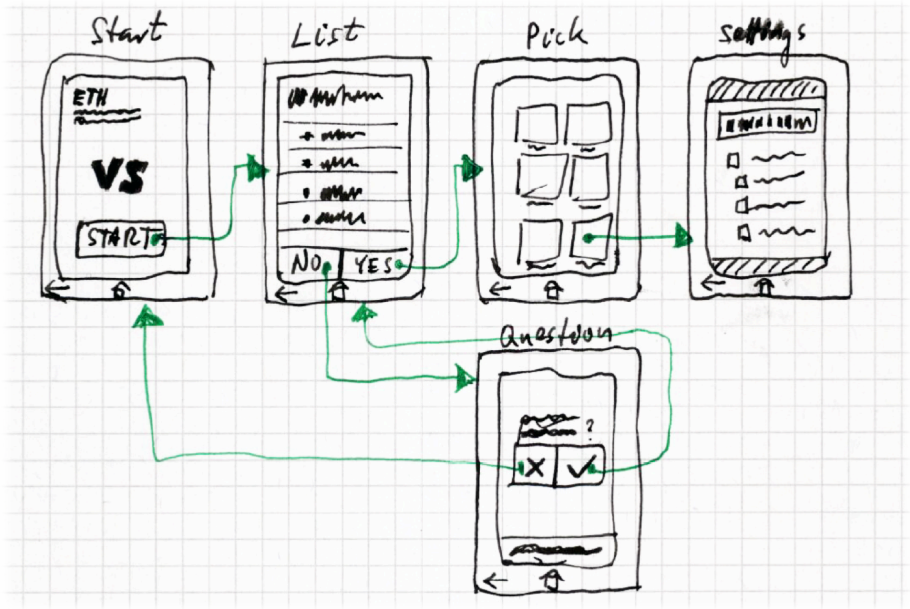
The 10 Usability Principles (Jakob Nielsen)

1. Keep the interface simple!
2. Speak the user’s language!
3. Minimize the user’s memory load!
4. Be consistent and predictable!
5. Provide feedback!
6. Design clear exits and closed dialogs!
7. Offer shortcuts for experts!
8. Help to recover from errors, offer Undo!
9. Prevent errors!
10. Include help and documentation!

www.useit.com

Workflow Prototyping and Evaluation on Paper

- Concrete guidelines: http://developer.android.com/design
  - Visual appearance, e.g., icon design
  - Purpose of user interface elements
  - Layout of user interface elements
  - Behavior, conventions of system features



Hints – Settings Activity (1)

http://developer.android.com/guide/topics/ui/settings.html

MySettingsActivity.java

```
package ch.ethz.inf.vs.android.nethz.project;
import android.os.Bundle;
import android.preference.PreferenceActivity;

public class MySettingsActivity extends PreferenceActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // Load the preferences from an XML resource
        addPreferencesFromResource(R.xml.preferences);
    }
}
```

Do not forget to add this Activity to your application in the manifest file!

Hints – Settings Activity (2)

\res\xml\preferences.xml

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android">
    <PreferenceCategory android:title="Server IP settings">
        <CheckBoxPreference
           android:key="network_mode_enable"
           android:title = "Stream to server"
           android:summary = "Enable to stream the sensor values to the server"
           android:defaultValue = "true" />
        <EditTextPreference
            android:key = "server_ip"
           android:dependency = "network_mode_enable"
           android:title = "IP address"
           android:summary = "IP address of the fancy server"
           android:layout = "?android:attr/preferenceLayoutChild"
           android:dialogTitle = "IP address of the Tracker"
           android:defaultValue = "192.168.1.1" />
        <EditTextPreference
           android:key = „server_port"
           android:dependency = "network_mode_enable"
            . . . />
    </PreferenceCategory>
</PreferenceScreen>
```

Hints – Settings Activity (3)

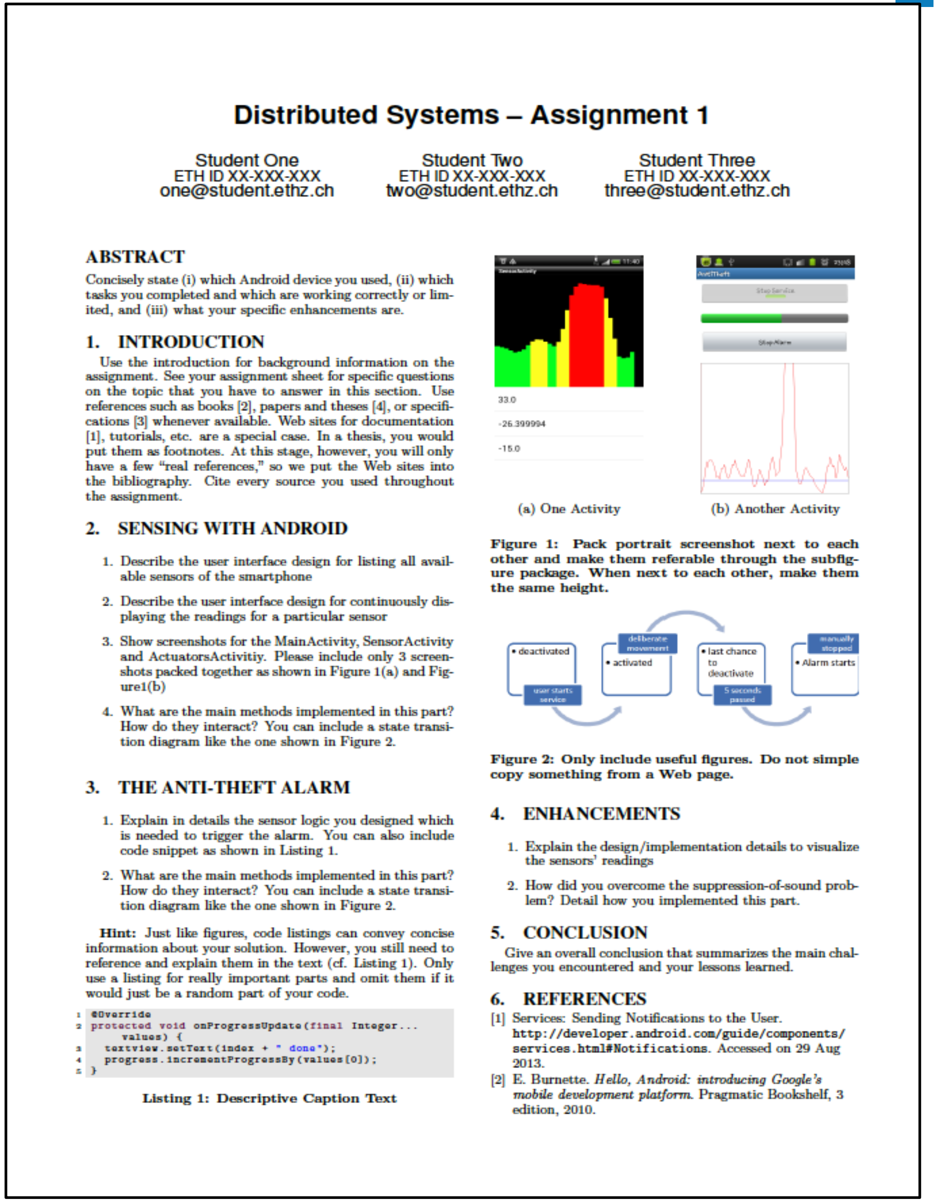
MyMainActivity.java

```
import android.preference.PreferenceManager;
```

```
SharedPreferences sharedPrefs =
      PreferenceManager.getDefaultSharedPreferences( getApplicationContext() );

boolean network_mode_enable =
      sharedPrefs.getBoolean( "network_mode_enable", false );

InetAddress saddr =
      InetAddress.getByName( sharedPrefs.getString( "server_ip", "192.168.1.1" ) );

int sport =
      Integer.parseInt( sharedPrefs.getString( "server_port", "8080" ) );
```

Report

- Required for each project
- LaTeX or Word
- English
- 1-2 pages
- Follow the template provided to you on the course website:
  - Abstract, Introduction, Sensing, AntiTheft, Enhancement, Conclusion, References
  - Figures, code snippets
  - Scientific writing



Distributed Systems – Assignment 1

Student One
ETH ID XX-XXX-XXX
one@student.ethz.ch

Student Two
ETH ID XX-XXX-XXX
two@student.ethz.ch

Student Three
ETH ID XX-XXX-XXX
three@student.ethz.ch

ABSTRACT

Concisely state (i) which Android device you used, (ii) which tasks you completed and which are working correctly or limited, and (iii) what your specific enhancements are.

1. INTRODUCTION

Use the introduction for background information on the assignment. See your assignment sheet for specific questions on the topic that you have to answer in this section. Use references such as books [2], papers and theses [4], or specifications [3] whenever available. Web sites for documentation [1], tutorials, etc. are a special case. In a thesis, you would put them as footnotes. At this stage, however, you will only have a few "real references," so we put the Web sites into the bibliography. Cite every source you used throughout the assignment.

2. SENSING WITH ANDROID

1. Describe the user interface design for listing all available sensors of the smartphone
2. Describe the user interface design for continuously displaying the readings for a particular sensor
3. Show screenshots for the MainActivity, SensorActivity and ActuatorsActivitiy. Please include only 3 screenshots packed together as shown in Figure 1(a) and Figure1(b)
4. What are the main methods implemented in this part? How do they interact? You can include a state transition diagram like the one shown in Figure 2.

3. THE ANTI-THEFT ALARM

1. Explain in details the sensor logic you designed which is needed to trigger the alarm. You can also include code snippet as shown in Listing 1.
2. What are the main methods implemented in this part? How do they interact? You can include a state transition diagram like the one shown in Figure 2.

Hint: Just like figures, code listings can convey concise information about your solution. However, you still need to reference and explain them in the text (cf. Listing 1). Only use a listing for really important parts and omit them if it would just be a random part of your code.

```
@Override
protected void onProgressUpdate(final Integer...
    values) {
  textview.setText(index + " done");
  progress.incrementProgressBy(values[0]);
}
```

Listing 1: Descriptive Caption Text

(a) One Activity

(b) Another Activity

Figure 1: Pack portrait screenshot next to each other and make them referable through the subfigure package. When next to each other, make them the same height.

Figure 2: Only include useful figures. Do not simple copy something from a Web page.

4. ENHANCEMENTS

1. Explain the design/implementation details to visualize the sensors' readings
2. How did you overcome the suppression-of-sound problem? Detail how you implemented this part.

5. CONCLUSION

Give an overall conclusion that summarizes the main challenges you encountered and your lessons learned.

6. REFERENCES

[1] Services: Sending Notifications to the User. `http://developer.android.com/guide/components/services.html#Notifications`. Accessed on 29 Aug 2013.

[2] E. Burnette. *Hello, Android: introducing Google's mobile development platform*. Pragmatic Bookshelf, 3 edition, 2010.

Report –Structure

- Abstract
  - Summarize your key take-away points of A1 for Android programming in a **brief** first sentence, e.g., (at least) Activity vs Service and hardware access
  - State which tasks you solved to which degree
  - Highlight your custom extensions
- One section per app
  - App design and program structure (cf. figure on slide 13)
  - Development decisions
  - Problems and solutions
  - Code snippets, UML diagrams, app Snapshots
  - Who did what
- Conclusion
  - Discuss the lessons learned in more detail
  - Comment on open issues if any

Deliverables

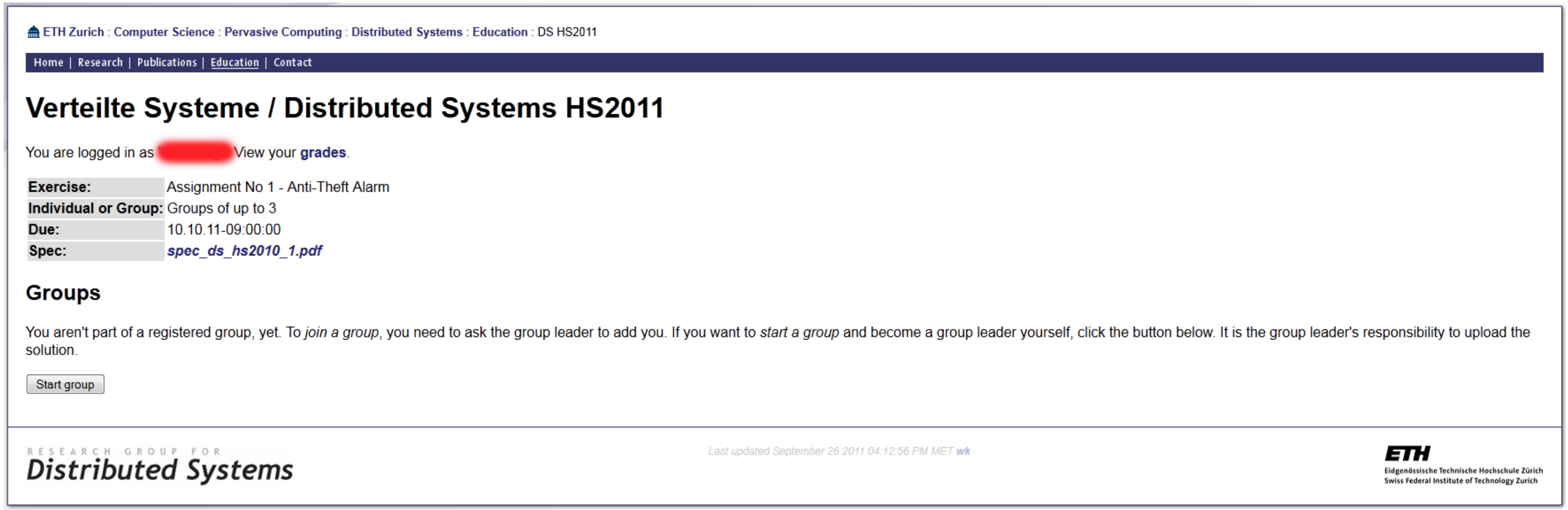
- https://www.vs.inf.ethz.ch/edu/vs/submissions/
- Use your *nethz* logins
- File names: **report.pdf** and **code.zip**
- New uploads will overwrite the old ones
- Check uploaded files

LEADER:

- Create group
- Add members
- Upload files

MEMBERS:

- Sign the submission



ETH Zurich : Computer Science : Pervasive Computing : Distributed Systems : Education : DS HS2011

Verteilte Systeme / Distributed Systems HS2011

You are logged in as [redacted] View your **grades**.

Exercise:	Assignment No 1 - Anti-Theft Alarm
Individual or Group:	Groups of up to 3
Due:	10.10.11-09:00:00
Spec:	*spec_ds_hs2010_1.pdf*

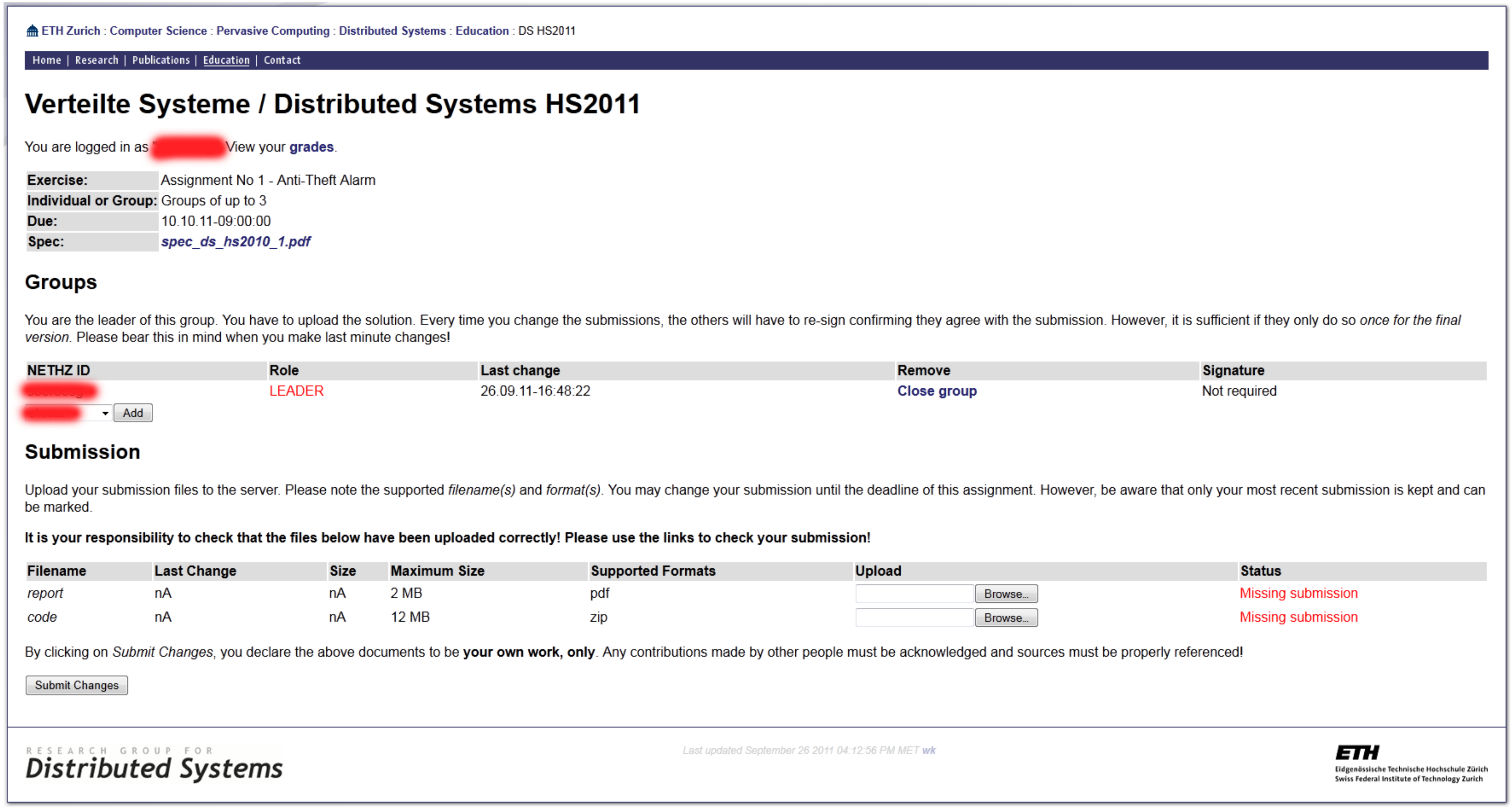
Groups

You aren't part of a registered group, yet. To *join a group*, you need to ask the group leader to add you. If you want to *start a group* and become a group leader yourself, click the button below. It is the group leader's responsibility to upload the solution.

Last updated September 26 2011 04:12:56 PM MET wk

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Leader creates a group



ETH Zurich : Computer Science : Pervasive Computing : Distributed Systems : Education : DS HS2011

Verteilte Systeme / Distributed Systems HS2011

You are logged in as [redacted] View your **grades**.

Exercise:	Assignment No 1 - Anti-Theft Alarm
Individual or Group:	Groups of up to 3
Due:	10.10.11-09:00:00
Spec:	***spec_ds_hs2010_1.pdf***

Groups

You are the leader of this group. You have to upload the solution. Every time you change the submissions, the others will have to re-sign confirming they agree with the submission. However, it is sufficient if they only do so *once for the final version*. Please bear this in mind when you make last minute changes!

NETHZ ID	Role	Last change	Remove	Signature
[redacted]	LEADER	26.09.11-16:48:22	**Close group**	Not required
[redacted] ▾ Add				

Submission

Upload your submission files to the server. Please note the supported *filename(s)* and *format(s)*. You may change your submission until the deadline of this assignment. However, be aware that only your most recent submission is kept and can be marked.

It is your responsibility to check that the files below have been uploaded correctly! Please use the links to check your submission!

Filename	Last Change	Size	Maximum Size	Supported Formats	Upload	Status
report	nA	nA	2 MB	pdf	Browse...	Missing submission
code	nA	nA	12 MB	zip	Browse...	Missing submission

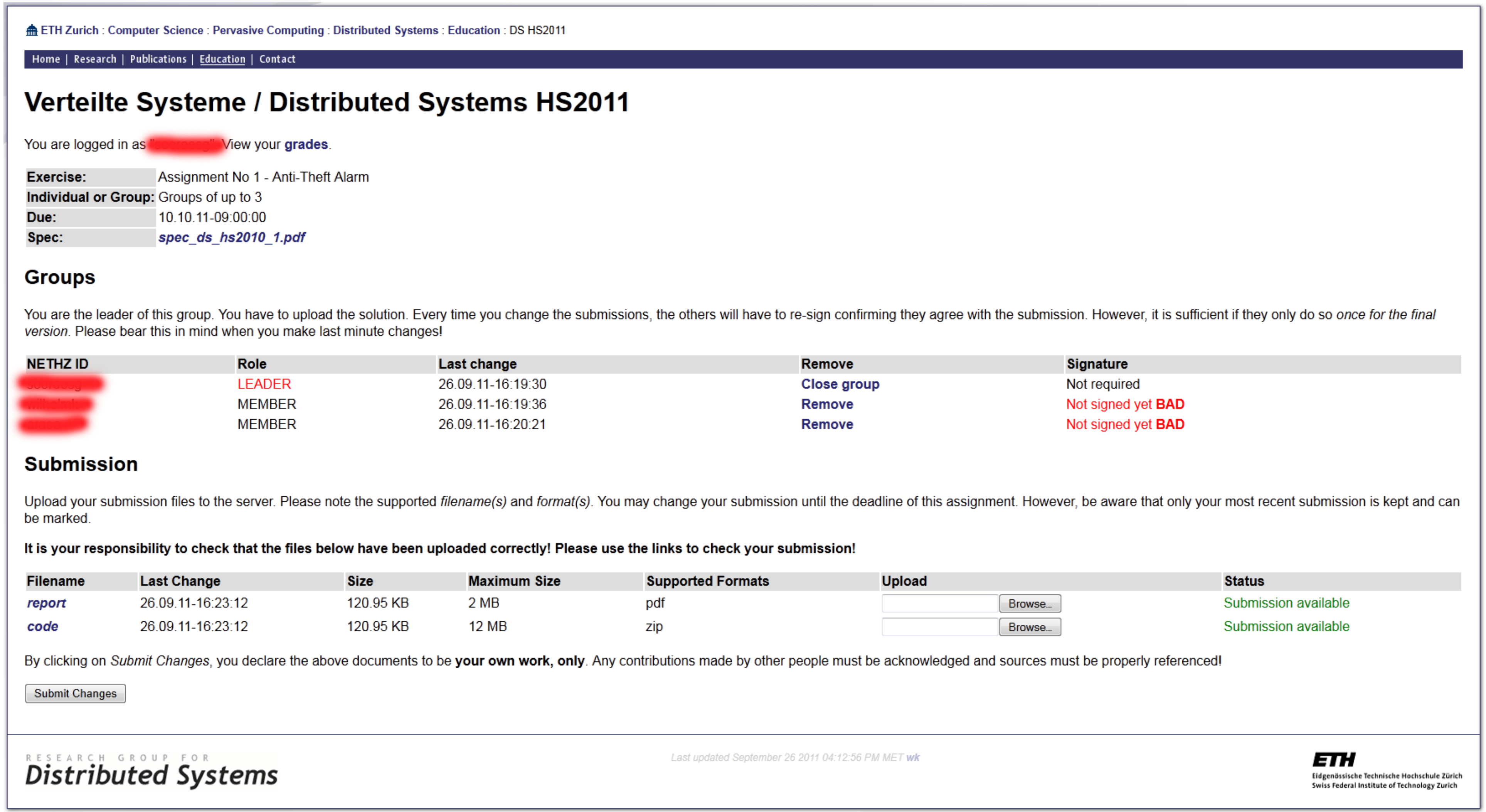
By clicking on *Submit Changes*, you declare the above documents to be **your own work, only**. Any contributions made by other people must be acknowledged and sources must be properly referenced!

RESEARCH GROUP FOR Distributed Systems

Last updated September 26 2011 04:12:56 PM MET wk

ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

Leader adds teammates



ETH Zurich : Computer Science : Pervasive Computing : Distributed Systems : Education : DS HS2011

Verteilte Systeme / Distributed Systems HS2011

You are logged in as [redacted] View your **grades**.

Exercise:	Assignment No 1 - Anti-Theft Alarm
Individual or Group:	Groups of up to 3
Due:	10.10.11-09:00:00
Spec:	***spec_ds_hs2010_1.pdf***

Groups

You are the leader of this group. You have to upload the solution. Every time you change the submissions, the others will have to re-sign confirming they agree with the submission. However, it is sufficient if they only do so *once for the final version*. Please bear this in mind when you make last minute changes!

NETHZ ID	Role	Last change	Remove	Signature
[redacted]	LEADER	26.09.11-16:19:30	**Close group**	Not required
[redacted]	MEMBER	26.09.11-16:19:36	**Remove**	Not signed yet **BAD**
[redacted]	MEMBER	26.09.11-16:20:21	**Remove**	Not signed yet **BAD**

Submission

Upload your submission files to the server. Please note the supported *filename(s)* and *format(s)*. You may change your submission until the deadline of this assignment. However, be aware that only your most recent submission is kept and can be marked.

It is your responsibility to check that the files below have been uploaded correctly! Please use the links to check your submission!

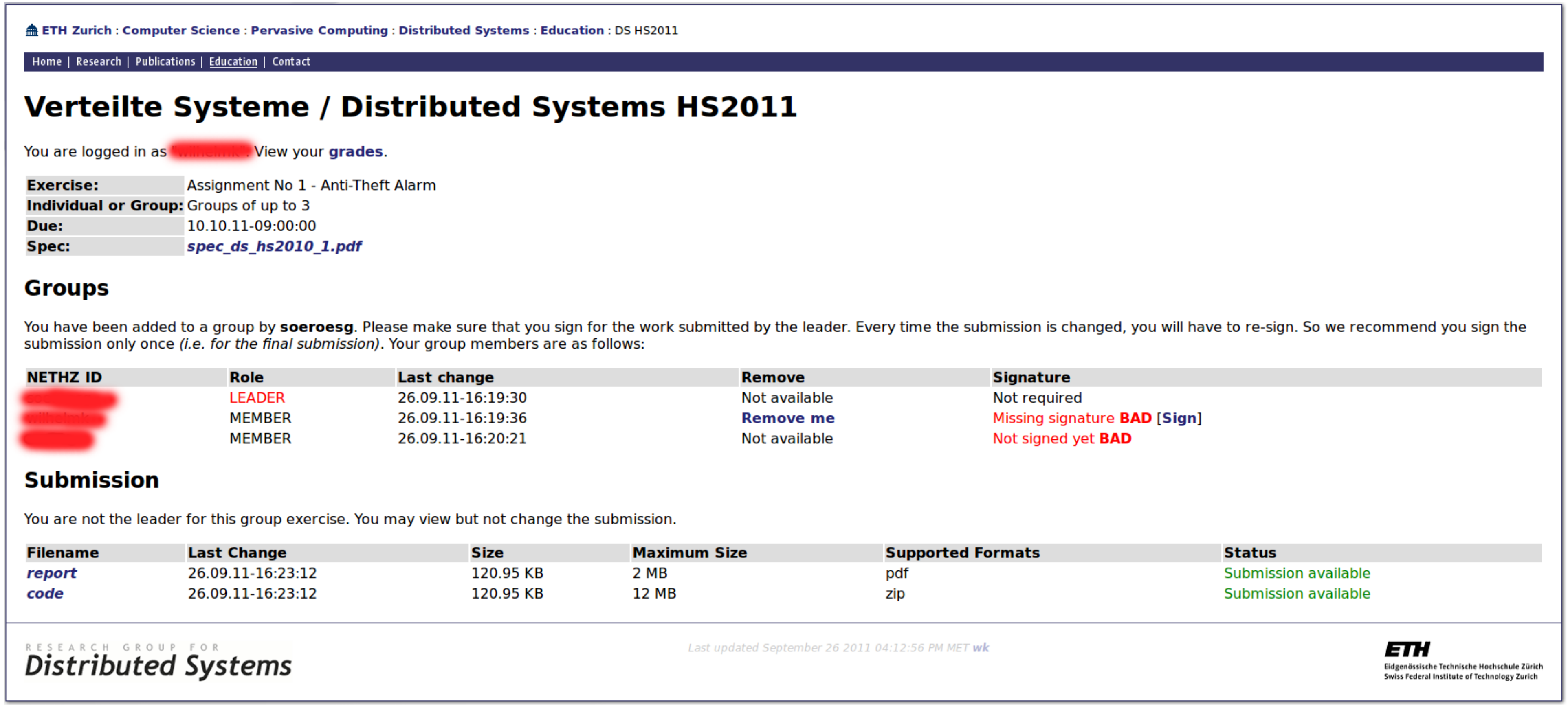
Filename	Last Change	Size	Maximum Size	Supported Formats	Upload	Status
report	26.09.11-16:23:12	120.95 KB	2 MB	pdf	Browse...	Submission available
code	26.09.11-16:23:12	120.95 KB	12 MB	zip	Browse...	Submission available

By clicking on *Submit Changes*, you declare the above documents to be **your own work, only**. Any contributions made by other people must be acknowledged and sources must be properly referenced!

Last updated September 26 2011 04:12:56 PM MET wk

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Leader uploads the files



ETH Zurich : Computer Science : Pervasive Computing : Distributed Systems : Education : DS HS2011
Home | Research | Publications | Education | Contact

Verteilte Systeme / Distributed Systems HS2011

You are logged in as [redacted] View your **grades**.

Exercise:	Assignment No 1 - Anti-Theft Alarm
Individual or Group:	Groups of up to 3
Due:	10.10.11-09:00:00
Spec:	***spec_ds_hs2010_1.pdf***

Groups

You have been added to a group by **soeroesg**. Please make sure that you sign for the work submitted by the leader. Every time the submission is changed, you will have to re-sign. So we recommend you sign the submission only once *(i.e. for the final submission)*. Your group members are as follows:

NETHZ ID	Role	Last change	Remove	Signature
[redacted]	LEADER	26.09.11-16:19:30	Not available	Not required
[redacted]	MEMBER	26.09.11-16:19:36	**Remove me**	Missing signature **BAD** [**Sign**]
[redacted]	MEMBER	26.09.11-16:20:21	Not available	Not signed yet **BAD**

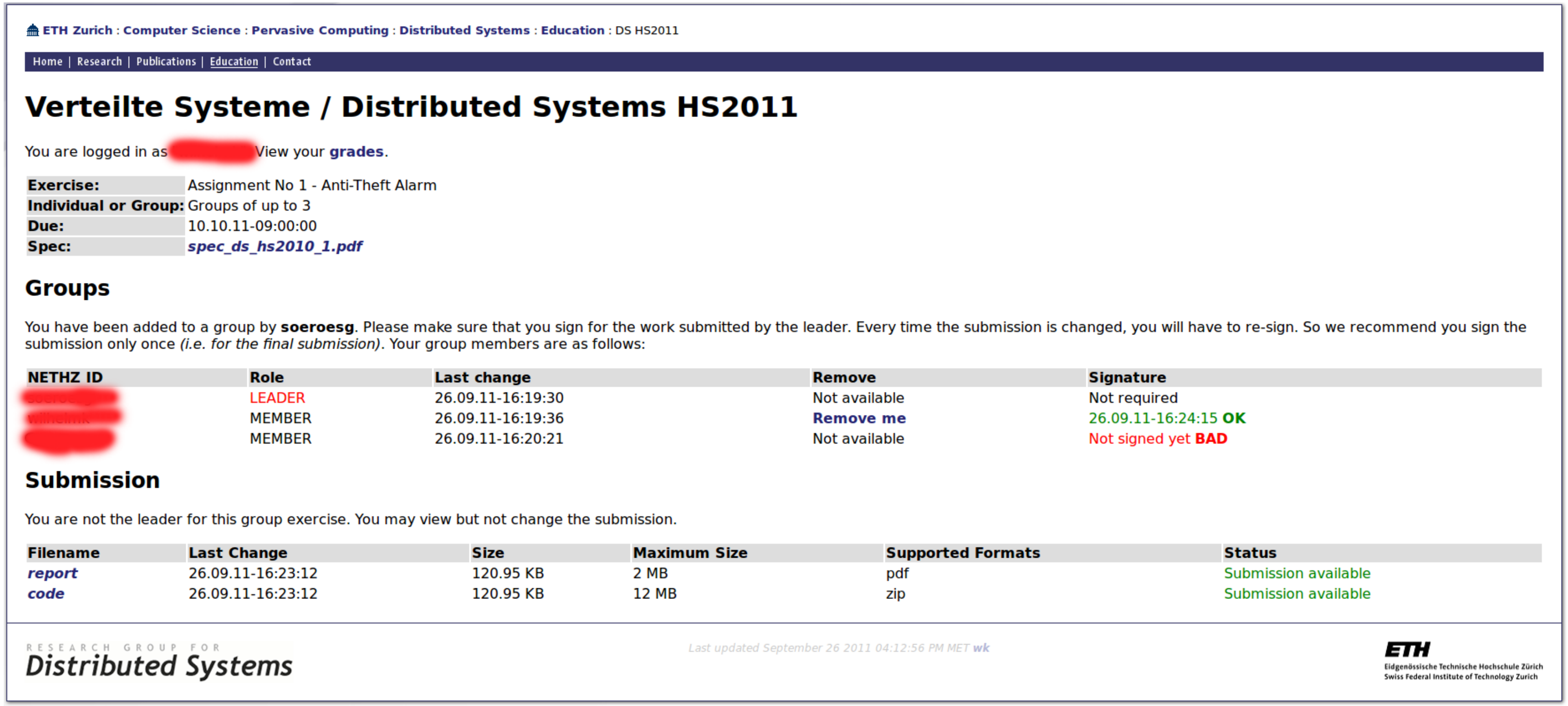
Submission

You are not the leader for this group exercise. You may view but not change the submission.

Filename	Last Change	Size	Maximum Size	Supported Formats	Status
report	26.09.11-16:23:12	120.95 KB	2 MB	pdf	Submission available
code	26.09.11-16:23:12	120.95 KB	12 MB	zip	Submission available

RESEARCH GROUP FOR
Distributed Systems
Last updated September 26 2011 04:12:56 PM MET wk
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Member logs in



ETH Zurich : Computer Science : Pervasive Computing : Distributed Systems : Education : DS HS2011

Verteilte Systeme / Distributed Systems HS2011

You are logged in as [redacted] View your **grades**.

Exercise:	Assignment No 1 - Anti-Theft Alarm
Individual or Group:	Groups of up to 3
Due:	10.10.11-09:00:00
Spec:	*spec_ds_hs2010_1.pdf*

Groups

You have been added to a group by **soeroesg**. Please make sure that you sign for the work submitted by the leader. Every time the submission is changed, you will have to re-sign. So we recommend you sign the submission only once *(i.e. for the final submission)*. Your group members are as follows:

NETHZ ID	Role	Last change	Remove	Signature
[redacted]	LEADER	26.09.11-16:19:30	Not available	Not required
[redacted]	MEMBER	26.09.11-16:19:36	**Remove me**	26.09.11-16:24:15 **OK**
[redacted]	MEMBER	26.09.11-16:20:21	Not available	Not signed yet **BAD**

Submission

You are not the leader for this group exercise. You may view but not change the submission.

Filename	Last Change	Size	Maximum Size	Supported Formats	Status
report	26.09.11-16:23:12	120.95 KB	2 MB	pdf	Submission available
code	26.09.11-16:23:12	120.95 KB	12 MB	zip	Submission available

RESEARCH GROUP FOR Distributed Systems

Last updated September 26 2011 04:12:56 PM MET ***wk***

ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

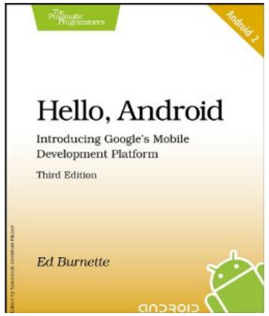
Member signs the submission

Remarks

- Point reduction if the project does not compile!
- Point reduction if the project does not follow the naming conventions!
- Point reduction if the project does not use the code skeleton provided to you!

- Task 2 and Report are necessary to meet the minimum requirements.
- Elements of Task 1 can be reused in Task 2.

Books


The Pragmatic Programmers
Android 2
Hello, Android
Introducing Google's Mobile Development Platform
Third Edition
Ed Burnette

- Ed Burnette – Hello, Android (Pragmatic Bookshelf 2010)
  http://kronox.org/documentacion/Hello.Android.new.pdf

- Mark L. Murphy – The Busy Coder's Guide to Android Development (CommonsWare)
  http://commonsware.com/Android/

- Arno Becker – Android Grundlagen und Programmierung
  http://www.dpunkt.de/ebooks_files/free/3436.pdf

Have fun!

Questions?