

WeFiLab: A Web-based WiFi Laboratory Platform for Wireless Networking Education

Lin Cui, Fung Po Tso, Di Yao, Weijia Jia, *Senior Member, IEEE*
City University of Hong Kong, Tat Chee Avenue, Kowloon HKSAR

Abstract—Remote access to physical laboratories for education has been received significant attention from both researchers and educators as it provides access at reduced cost in sharing manner of real devices and gives students practical training. With the rapid growing of wireless technologies, it has been become an essential of learning to have the hand-on experience on wireless networking for the proliferated number of students in engineering. Some of current implementations for wireless networking are either using simulations, which lose the reality, or too complicated for undergraduate students to control experiments. In this paper, we present a practical online laboratory platform, Web-based WiFi Laboratory (WeFiLab). WeFiLab mainly focuses on providing students hand-on experience of doing experiments on real devices through webpage anytime anywhere. It uses the structure of two-level operations, which facilities increasing the scale of wireless devices and allows WeFiLab to be extended to more complicated operations. The schedule schemes of WeFiLab let more students share and make efficient use of wireless devices. A prototype of WeFiLab has been implemented and used successfully as complements for an undergraduate course for two years in a university. Totally 315 computer science students attended the evaluation.

Index Terms—Wireless Networking, Remote Laboratories, Web-based Laboratories, E-learning

1 INTRODUCTION

1.1 Motivation

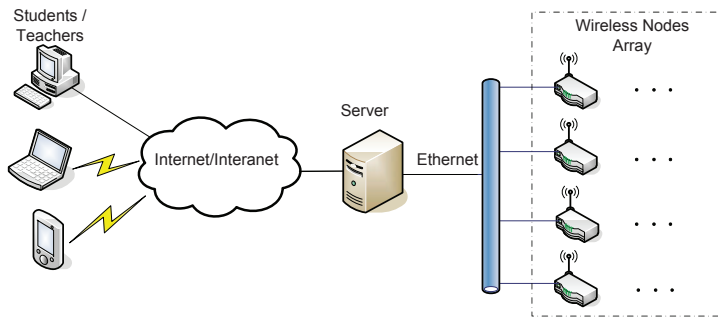
Networking technologies have been playing irreplaceable roles in today's global village. With the Internet, information at one corner of the world can instantly travel to another corner of the world. In particular, Mobile Internet which links together mobile users has contributed an essential part to this achievement. Recent years have witnessed an explosive growth of Mobile Internet access fueled by the increasing popularity of WiFi and the worldwide deployment of high speed wide area wireless networks such as the third generation (3G) wireless cellular networks (e.g. HSPA). On the other hand, proliferation of hand-held multi-mode Smartphones and other wireless devices has significantly stimulated the use of wireless Internet, e.g. the city wide high-bandwidth low-priced WiFi networks. As the popularity of wireless networking increases, the industry and academic have accumulated a huge demand for people with the necessary skills to design and create practical networking facilities.

For science and engineering students, laboratory plays a crucial role in their undergraduate degree courses because the laboratory activities provide the hands-on problem solving skills and necessary experience for adopting their knowledge in real-life contexts [1] [2]. With enough laboratory experience, students can put concepts into practice to consolidate their knowledge learned during lectures. Nevertheless, running laboratories creates high financial burden to colleges or universities because they are expensive and resource-intensive. So traditional laboratories have following shortcomings.

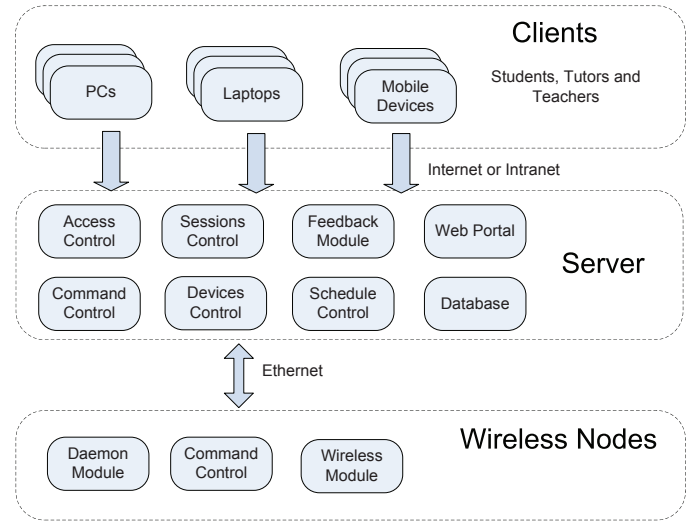
First, a number of qualified staffs are required to manage the laboratory and assist the conduct of experiments. Second, constant budgets are needed for continuous equipment maintenance and renewal. Third, laboratories are only accessible during opening hours for a very limited amount of registered students. Nowadays, mobile learning that not being tied to a particular location has become the next generation of E-learning [3], but the traditional laboratories have failed to meet this demand. Due to these reasons, a plenty of previous efforts have suggested various types of online laboratories which allow students to access via Internet to share the computer controlled equipments [4] [5] [6] [7] [8] [9].

Generally, there are mainly two approaches for on-line laboratories: virtual laboratories and remote laboratories [10]. Virtual laboratories are based on software to simulate experiments environment, while in remote laboratories experiments are conducted and controlled remotely through the Internet, using real components or devices. One of the very important features of virtual laboratories is to let the students learn from failures without causing any real damages. However, virtual laboratories limit students learning activities due to the missing of reality [11]. While remote laboratories are designed to provide real-time experiments to students via the Internet, gives students the opportunity to work on the remote real devices, which will eventually provide students real laboratory experience.

However, our literature review results show that most of the current remote laboratories are too complicated for network beginners as students may need to write programs/scripts to conduct experiments. Among reviewed projects, we also analyzed their web laboratories



(a) System Architecture



(b) Logical Components

Fig. 1: WeFiLab Architecture

and found that Java, Flash or ActiveX are frequently used tools for User Interfaces (UIs) to link the laboratory equipments to the users. These tools can be utilized to build powerful and sophisticated systems, but they are also too resource-hungry and computation-intensive to be employed on some resource limited devices such as mobile devices.

1.2 Contribution

To resolve the above mentioned problems, we present our Web-based WiFi (IEEE 802.11) Laboratory platform - WeFiLab. We believe that WeFiLab covers both beginning and advanced learners as it enables students to learn, produce the network design and conduct experiments about WiFi networks through the web browser via Internet anytime anywhere. All these learning and experiments activities are conducted on real wireless devices.

The WeFiLab system has been successfully used as the complement for the course of CS4284-*Mobile Computing* [12] and the course CS4289-*Pervasive Computing* for two years, which totally included 315 undergraduate computer science students, in City University of Hong Kong. The case studies' results show that WeFiLab not only improve students' understanding about the WiFi but also provide them with adequate hands-on experience about WiFi network design.

In short, our main contributions are three-folded:

- 1) We have designed and prototyped a Web-based WiFi laboratory platform to control and manage various wireless devices for conducting experiments via Internet;
- 2) We designed and implemented a plugin free Web-based GUI framework which is compatible with most of the current popular web browsers, allowing students to conduct experiments on the web page and see the response immediately, without writing any programs/scripts;

- 3) We have evaluated the WeFiLab through real case studies for two courses with 315 students involved.

The rest of this paper is organized as follows: Section 2 provides some background and related works about networking laboratories. Section 3 describes the details of the WeFiLab design including the structure of two-level operations, roles and capacity of WeFiLab. Section 4 introduces some experiments that we designed based on WeFiLab. Section 5 is the implementation of the prototype of WeFiLab. And two study cases for evaluation of WeFiLab including the OBTL (Outcome-Based Teaching and Learning) scheme are introduced in Section 6. In Section 7, we discuss the current challenges problems and give some future works about WeFiLab. Finally, Section 8 is the conclusion of the paper.

2 RELATED WORKS

In education, simulation software is seen as one way of imparting practical knowledge by allowing students to conduct experiments on the computer simulating all the steps, which a student would take in real laboratory while performing the experiments [13]. Compared to the cost and time involved in setting up an actual laboratory, network simulators are considered as a fast and inexpensive way to provide students with hand-on training on networking. Graphical User Interface (GUI) is often provided to allow users to configure and visualize the simulation environment and to display simulation result. There are a variety of network simulators, ranging from the very simple ones to the very complex ones [14]. Usually, the simple ones enable a user to represent a network topology, specify the nodes (e.g. switch or router) on the network, the links between those nodes and the traffic between the nodes. On the contrary, the complex ones allow the user to configure everything about the protocols used to handle network traffic.

Some educators argue that although simulation studies can make students meet the requirements of the

course and enable him to complete it satisfactorily from the point of view of the university that offers the course, the element of reality will be missing, which will cause students to become spectators rather than learners [11]. Involvement of body and mind in a real experiment yields rich dividends of knowledge gained by the experimenter. Remote laboratories take the advantages of real laboratories while also extend the capability of a conventional laboratory by increasing the number of times and places students can perform experiments [15], [16] and extending its availability to larger groups of students [17]. In the last years, significant research effort has been done by teaching institutions on the development of remote laboratories for a wide variety of engineering courses. Authors in [18] described the history of some of these changes and explored in some depth a few of the major factors influencing laboratories today. Authors in [19] have developed an online RFID laboratory. Students can access to technological resources through the Web technology to apply appropriate configurations to the system, conduct experiments using RFID technology, and perform statistical analysis on the acquired data. [20] described the authors' experience in the development of a reusable framework for remote laboratories, which has been adopted as a case study for IT engineering classes. Authors in [5] [6] [7] [8] share their experience on building a networking laboratory, in particular, J. DeHart et al. [8] have built a remotely accessible network testbed of high performance routers for users from the naive to expert. Emulab [4] is an online virtual and remote laboratory which has been used for several universities. It provides both virtual nodes and real test-bed for both wired and wireless networks. One of the important features of Emulab is that it uses the same directives as the ns2 simulator for setting up real nodes and connections. The main difference between WeFiLab and Emulab is that WeFiLab tries to let students conduct experiments through mouse-events and view the responses immediately on the web page, while Emulab requires users to write complicated scripts to control experiments.

While a lot of efforts have been paid to develop Web-based frameworks for various uses of remote laboratories, many out of them adopt Java or ActiveX or other technologies that require additional plug-ins for Web browsers. Authors in [21] pointed out the shortcomings of existing implementations and indicated that emerging internet technologies can assist in overcoming these shortcomings. Nevertheless, these technologies aim to serve end users with powerful terminals. WeFiLab specifically adopts Ajax, which is compatible with most of the current popular standard web browsers. The main underlying technology provides true anywhere anytime wireless networking laboratories.

3 WEFiLAB DESIGN

Following the footsteps of previous successful works, we provide one step further system WeFiLab, which

aims at teaching and learning on wireless networking. At current state, the system only concerns about 802.11b/g (WiFi). The reason for setting WiFi off first is that it is so far the most commonly available wireless networking technology surrounding us. We assume that the system users are both students and teachers/tutors of a course in a university/college. Students can gain instant access to the WeFiLab for hands-on training anytime anywhere so as to consolidate the knowledge they have already gained from lectures. Teachers/tutors can design experiments, assess students performance and review students' records, e.g. operation history, mistakes they have made and their experiments results. All these information will help teachers to understand students' performance and coordinate their teaching activities.

The design of WeFiLab focuses on encouraging students to be fully involved into the experiments on real wireless devices. It hides the details of the complex configurations on wireless devices from users through the two-level operation structure, so students can concentrate on their experiments, which are conducted completely on web page. WeFiLab also allows the administrator to add new wireless devices whenever needed to support more students.

3.1 System Architecture

The illustration of WeFiLab architecture is shown in Fig. 1a. The whole WeFiLab system consists of a server, the wireless nodes array and edge client terminals. The corresponding logical components of the whole system are depicted in Fig. 1b.

Clients: All students, teachers and tutors can be users of the WeFiLab. Web browsers running on their terminals connecting them to the WeFiLab are the clients of the system. These terminals can be PCs, laptop computers and even mobile devices, e.g. mobile smartphones.

Server: As a bridge between clients and wireless devices, the server plays two important roles in the WeFiLab platform. One is to act as a web server to provide Ajax based user-friendly GUI for enabling sufficient interactivity for students and teachers/tutors. Another role of the server is to act as a gateway to relay control and response messages between clients and wireless devices. Building on top of Ajax, the web server allows every user with capable web-browsers to log on and use the provided functions without additional plugins even on mobile phones. And using the structure of two-level operations, server enables users to control the assigned wireless devices through web browser remotely. All the operations are done on the web page, students needn't write any scripts/programs. The database equipped along with the web server facilitates the control of users' sessions and wireless devices. It can also store all the useful records/data, e.g. operation histories and captured frames produced during the experiments, for students' and teachers' further study and analysis.

The Wireless Nodes Array: The wireless nodes array contains a group of off-the-shelf wireless devices (routers), which all the experiments of students are actually carried on. These wireless devices are shared among all the users of WeFiLab. All these devices are connected to the server through Ethernet for the exchange of control and management messages. The Ethernet can provide efficient communication between wireless devices and server without influencing the wireless communication among wireless devices. Each wireless device receives instructions from the server and sends back the response/results upon finishing the tasks. These wireless devices can work under different modes according to students' experiments, such as AP/STA, Ad Hoc¹ mode and so on. Particularly, there are also some devices working under the monitor(sniffer) mode, which can capture all the wireless frames in the air and filter the results based on students' needs.

Some typical logical components of WeFiLab in Figure 1b are described as follows:

Session Control: The WeFiLab system will create a *session* for each login user. Then all the users' operations will be tracked for further analysis. Especially, each session has a session timer, which will be expired after MAX_SESSION_TIMEOUT. The system will retrieve all resources, e.g. wireless devices, allocated to the user once the user's session expired.

Feedback Module: The Feedback Module is for pushing messages from server to clients. These messages include any updates of wireless devices' configuration, status and responses after users' operations, as well as teachers'/tutors' suggestions during the experiments.

Schedule Control: The Schedule Control allows students to do the experiment through two different schemes: *Reservation* and *FCFS*. The details of the two schedule schemes are discussed in Section 3.4.

Device Control: The main tasks for this component are the wireless devices discovery and devices maintenance. It can either discover wireless devices through continuing listening on a pre-defined ports, or broadcast query packets for proactive discovery. After devices have been found, Device Control will further check the status of each device periodically, e.g. the working mode, current channel, SSID, IP address, etc. Based on these information, the system can assign available devices for the experiments according to users' requests.

Command Control: Command Control module is mainly responsible for the interpretation of commands between the two-level operations. The details are provided in Section 3.2.

3.2 Two-level Operations

The WeFiLab uses the structure of two-level operations to provide communication between clients and wireless

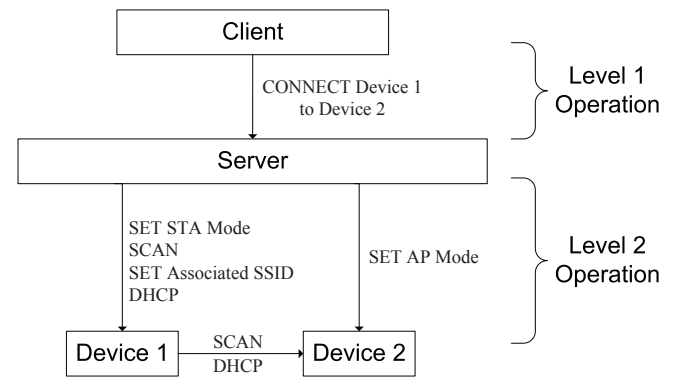


Fig. 2: An Example of Two-level Operations

devices. Fig. 2 gives an example of how the two-level operations work when a client wants to connect one wireless device to another under AP/STA mode.

The structure of two-level operations is explained as follows:

- 1) The first level operation is the communication between clients and server. These operations are *user-oriented* operations(commands) that are designed and abstracted for students according to the operations in wireless network. For example, as shown in Fig. 2, when the user wants to connect two devices, the corresponding operation in this level is CONNECT device 1 to device 2, which works in the AP mode.
- 2) The second level operation includes the interactions between server and wireless devices. This kind of operations are *device-oriented* specific operations(commands), which respond for instructing the wireless device to finish a specific task and give corresponding response/result. For example, SET the device 1 to STA mode (client), SCAN and SET the SSID of device 2 as the AP for association, obtain IP address through DHCP from device 2.

Under this structure, the server is responsible for interpreting these two kinds of operations(commands) between clients and wireless devices. Through the two-level operations, students just need to focus on the first level operations which are independent of specific wireless devices, without bothering the detailed operation commands on the specific platform wireless devices. As the client's operations of WeFiLab are GUI-based, the first-level operations are mainly triggered by mouse-events rather than scripts.

When a student needs to use some wireless devices to do experiments, he/she must lock those devices first to avoid unnecessary operation collisions that one device is used by more than one student simultaneously. After receiving the LOCK request from a client, the server will check the status of that device and do some necessary configuration to the wireless device before responding to the client.

After locking enough wireless devices successfully, students can conduct their experiments. These experiments can be designed by the teachers or based on

1. The AP/STA (access-point/station) and Ad Hoc are two typical modes of operation in the IEEE 802.11 standard [22].

the students' own idea. Using the two-level operations provided by WeFiLab, students can deploy a wireless network, capture wireless frames, send out specified packets and any other operations. Students can also change the configuration of each locked device, such as the SSID, current channel, the working mode and so on. Section 4 will introduce some basic experiments that we designed based on WeFiLab.

The structure of two-level operations also simplifies the maintenance of wireless devices. When students are doing experiments in WeFiLab, the status/configuration of the locked devices may be changed. In light of this, the server is responsible to recover the configuration of wireless devices automatically by using `WD_RECOVER` command to let devices reset all the configurations when these devices are released. So the other students can lock these devices for experiments without caring about the previous users' operations. At the same time, the *Daemon Process* running on the background of each wireless device also reports its status to server periodically. The server would response to each device after checking the report. Should there be any unexpected problems, i.e. wrong status or haven't received the report/response for a period, the device will recover the configuration or reset the network module of wireless devices if needed.

Under the structure of two-level operations, the scale of WeFiLab can be easily extended. New wireless devices can be added to the wireless nodes array anytime without changing the first level operations.

3.3 Integration of Roles in WeFiLab

A positive experiment environment for learning needs different roles to work out together, e.g. students, teachers, tutors, other support stuffs and so on. As a system to facilitate students learning on wireless network, WeFiLab tries to involve both students and teachers/tutors during experiments. Besides that each student can do their own experiments, view their operation histories and results, WeFiLab also allows teachers/tutors to view students' operation histories, experiment results and so on, which will give them more knowledge about students' performance and where the common weakness is for all students.

Besides, some experiments may need several students to cooperate together. WeFiLab also allows students to form a group with a selected group leader to coordinate the whole experiments.

We believe that effective learning during experiments should be bidirectional between students and teachers. So in addition to grading students' performance on experiments and questions, teachers/tutors can also give comments to students based on their performance on experiments and their submitted answers/reports. Besides, the WeFiLab offers students chances to give feedback about their experience of doing experiments and using the system to teachers and tutors.

3.4 Capacity of WeFiLab

The capacity of WeFiLab is defined as how many students can be allowed to do experiments during the same time period. Generally, it depends on the number of wireless devices and what kind of experiments students are doing.

Besides taking the advantage of the diversity of students' profile of diversity, WeFiLab also uses the lock restriction on wireless devices and the schedule scheme to ensure the capacity for students.

Students' Diversity: WeFiLab allows students to access at anytime anywhere as long as the Internet service is available. The web-based property of WeFiLab enables students do not need to do the experiments during course/tutorial time. Furthermore, students are also not required to do the experiments in the campus. Actually, according to our observations on the records of the system, more than 37% login records are outside the university campus and about 32% students indicated in their feedback that they do the experiment at their home. Because students can access WeFiLab at anytime anywhere, their login activities are distributed in a large range during the period available for their experiments. These diversities can increase the capacity of WeFiLab to some extent. We will give more detailed information about these diversities in the Section 6.3.

Lock Restriction: According to our observations, for wireless beginners, 2~3 wireless devices are almost enough for most experiments about wireless networking. And most of experiments can be finished in about 10~20 minutes if everything goes on well. To avoid that some students may lock devices too long or forget to unlock them after finishing experiments, the WeFiLab system will retrieve the devices after being locked for more than `MAX_LOCK_PERIOD` (current setting for WeFiLab is 60 minutes, which is enough for students to finish an experiment). Furthermore, when a student login WeFiLab, a *session* will be created for this user on the server. And the users' session will be expired after `MAX_SESSION_TIMEOUT` (current setting for WeFiLab is 30 minutes) since their last activity. The expiration of users' sessions will also trigger the server to retrieve all the devices locked by that user.

Schedule Scheme: Considering that students may spend more time on their experiments, WeFiLab provides two schedule schemes for students to lock devices and do experiments:

- 1) *Reservation Scheme (Scheduled):* The WeFiLab system allows students to reserve the requested devices for specific period (slots) according to their preference. The time slots for each student will be coordinated according to the available devices;
- 2) *FCFS Scheme (Non-Scheduled):* In addition to the Reservation Scheme, wireless devices are assigned to students by WeFiLab based on the First-Come-First-Served (FCFS) rule outside the schedule.

For those experiments that only need to capture some

wireless frames or scanning the environment, sometimes students don't need to lock any devices explicitly. They just need to send a request to server. The server will coordinate all these kinds of requests, capture the desired frames and send back the results to corresponding clients. Besides, each capture request usually takes only several seconds at most, so many students can be served at the same period with a few wireless devices. The *FCFS Scheme* will be enough for this kind of operations.

For those experiments that students need to lock several devices to do experiments, WeFiLab can use the *Reservation Scheme* (scheduled) to ensure that each student have enough time (e.g. a slot of 60 minutes) to do their experiments, and at the same time WeFiLab also uses *FCFS Scheme* to allow those students who failed completing experiments during the scheduled slot to have more chance to do their experiments.

Finally, as to the bandwidth of WeFiLab, the Ethernet connecting the wireless devices to server can be 100Mbps at least, so the bottleneck that can influence the throughput of WeFiLab is the bandwidth from clients to server through the Internet. Due to the structure of two-level operations, most of messages exchanged among Clients, Server and Wireless devices are short commands with only few bytes (within 4 bytes in current implementation), and sometimes one command of first-level operation can be translated to several commands of second-level operations. So the communication overhead needed for the control flow among Clients, Server and Wireless Devices is very low.

4 WEFiLAB EXPERIMENTS

We have carefully designed two kinds of experiments to maximize the benefits of WeFiLab for students. First of all, the experiments introduce the basics of WiFi, for example, letting students know what SSID, BSSID or Channel is and how do they work in connection with other WiFi devices. There are also some advanced experiments which let students decode/analyze wireless frames, enabling students to learn the details of WiFi frame structures and the communication process.

4.1 Knowing the Basics

Compared with wired networking concepts, many students find it harder to understand the fundamental concepts of wireless networking. Students either directly link wired networking with wireless one or have largely degree of misunderstanding about the wireless networking. In this section, we introduce several experiments provided by WeFiLab and describe how they can be utilized to help students know the basics of WiFi mechanism.

Almost in every textbook which introduces WiFi begins with the mode of operation, channels and associations. WiFi allows for both "Infrastructure" mode and "Ad Hoc" mode. A wireless access point (AP) is

required for infrastructure mode operation to bridge wireless subnetwork to Internet or enable communication among wireless client stations (STA). In comparison, Ad Hoc mode is a method for wireless clients to directly communicate with each other. Operating in Ad Hoc mode allows all wireless clients within coverage range of each other to discover and communicate in peer-to-peer fashion without involving an access point.

To help students to distinguish the difference between the two modes, the experiment allows students to operate up to four wireless devices at a time. The procedure of this experiment is:

Step 1: The students are first asked to configure one of the wireless devices to be AP (infrastructure) mode. The remaining devices act as wireless clients (STA mode).

Step 2: Students are then asked to associate the clients with the AP and send text messages among the clients to verify the inter connections.

Step 3: Once this is completed, it is followed by asking students to turn off the connectivity of AP node.

Step 4: Students are then instructed to send text messages among wireless clients again to guess and see what would happen.

Result: Text messages should never be received through the network because the connectivity among all clients are broken due to the lost of AP in the WiFi group.

For Ad Hoc mode experiment, as a comparison, students are guided to

Step 1: Turn one of the wireless nodes into Ad Hoc network initiator and then join other nodes to the network one by one.

Step 2: Students are then asked to send text messages among the clients to verify the inter connections.

Step 3: Once this is completed, it is followed by asking students to turn off the connectivity of the initiator node.

Step 4: Students are then instructed to send text messages among wireless clients again to guess and see what would happen.

Result: Text messages should be received through the network because the connectivity among all clients remain alive due to peer-to-peer connection under Ad Hoc mode in the WiFi group.

By observing the differences during the previous two experiments, students are expected to understand that nodes in Ad Hoc mode communicate peer-to-peer fashion, while infrastructure mode requires a central access point, the AP, for facilitating the communication among associated clients.

In 802.11, each wireless station needs to associate with an Access Point before it can send or receive network-layer data. When a network administrator installs an AP (or wireless router), the administrators must assign the name of the network known as its Server Set Identifier (SSID) that identifies a particular 802.11 wireless LAN. A client device receives broadcast messages (Beacon Frame) from all APs within range advertising their

SSIDs. The client device can then either manually or automatically select the desired network to associate with. The SSID can be up to 32 alphanumeric characters long. Multiple access points can share the same SSID if they provide access to the same network. SSID is the name of a wireless LAN, while the communication between AP and clients are based on the MAC address. So the SSID can be changed after finishing the association. To verify this, students are required to finish following experiment:

Step 1: Students are required to set the SSID for one of the wireless devices and turn it into AP mode.

Step 2: For the other devices, scan and associate themselves with the AP.

Step 3: Send text messages among them to verify the connections.

Step 4: Modify the SSID of AP and send text message again to verify the connection.

Result: The connections remain because the change of SSID does not affect ongoing connections since it is merely an identifier of a wireless LAN.

The administrator must also assign the channel that the AP uses. 802.11b/g operates in the frequency range of 2.4GHz, and there are 14 channels designated in this range spaced 5MHz apart. Any two channels are non-overlapping if they are separated by four or more channels. Consequently, the set of channels, 1, 6, 11, and 14 is recommended to avoid interference. Many students perceive WiFi as just a single radio channel which connect their WiFi capable devices to the internet. The following experiment aims at showing the students that the channels exist and wireless devices can only communicate in the same channel number.

Step 1: Students are asked to turn one of the wireless devices into AP mode.

Step 2: For the other devices, scan and associate themselves with the AP.

Step 3: Send text messages among them to verify the connections.

Step 4: Modify the channel of AP and send text message again to verify the connection.

Result: The connection is lost because WiFi devices can not communicate at different channels.

Through these serial of basic experiments about WiFi, students are supposed to understand the role of SSID and wireless channels during the association and data communication between AP and Client, and the difference of the Infrastructure mode and Ad Hoc mode.

4.2 The 802.11 Frame

For advanced learners, who want to dig deeper about 802.11 knowledge, WeFiLab can be used to go through 802.11 frame structure and study the detail communication between AP and client stations. To do this, reading through the IEEE 802.11 specification [22] is an option.

Some of the wireless devices can work at monitor mode, i.e. a sniffer, to capture all the wireless frames in a specific channel. WeFiLab is also equipped with a WiFi frame analyzer. The benefit of using the analyzer is to help students to go through complicated frame structure with real frame with enough aid of automation. For example, given a beacon frame, the analyzer can highlight the different fields instantly with different color, see Fig 3a. That can significantly stimulate students' interest towards learning. With the assistance of the equipped WiFi frame analyzer, students can decode each frame they captured. Furthermore, with all the captured frames from the sniffer, students can see the entire communication process, such as association, authentication, transmission and so on.

We design the following experiment about WiFi Frames, which mainly focuses on the AP/STA mode. The procedure is:

Step 1: Students are first asked to activate one sniffer to monitor the environment.

Step 2: Students need to capture some WiFi frames of a certain types, such as frames of Beacon, Probing Request/Response, Association Request/Response, Authentication and so on.

Step 3: Then students are asked to interpret the meaning of header/data fields of these frames according to the IEEE 802.11 Specification [22].

Step 4: Several questions will be displayed for each student to answer according the frames that they captured. Students can save and submit their answers on the server.

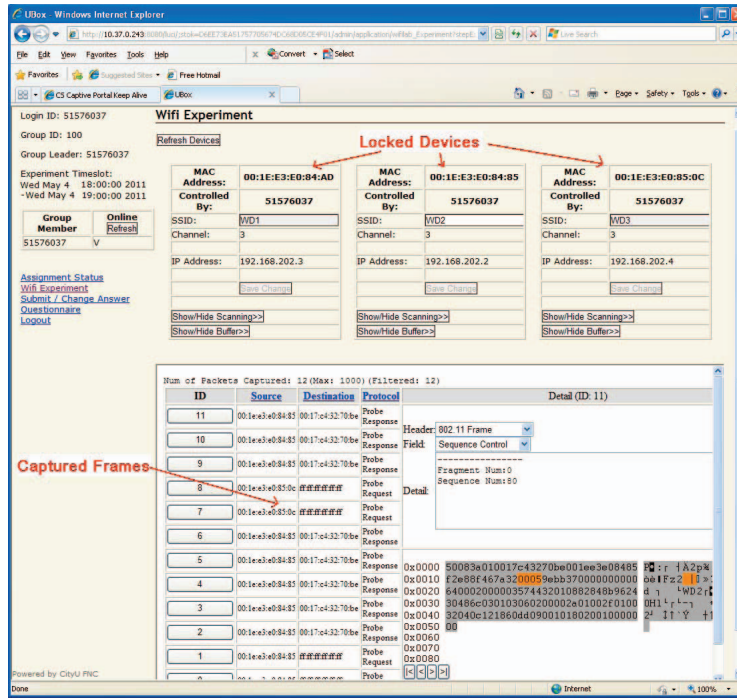
Once completed, all students' answers will be stored for grading and further studying by tutors/teachers. WeFiLab provides an AnswerCheck view for tutors/teachers to grade students' submissions. Also, teachers review the analysis of students' performance.

After this experiment, students are supposed to get a deep insight of the details of WiFi communication, such as the detail structure of WiFi frame, the association/authentication between AP and client, the detailed transmission of data packets and so on. Through the set of questions, teachers can trace students' performance and locate the common mistakes and misunderstandings.

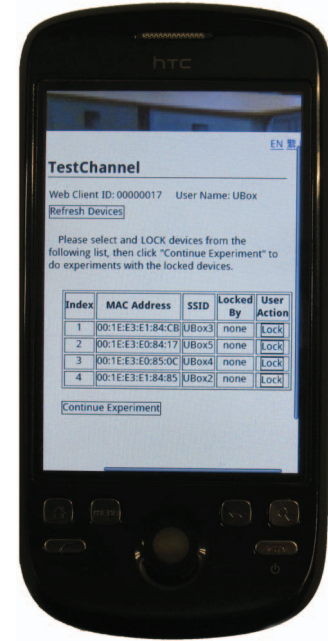
5 IMPLEMENTATION

We have successfully built the prototype of WeFiLab, which has been used for conducting experiments of a course for two years.

In our current implementation, we use 16 wireless routers to form the wireless nodes array. In order to enable efficient message transfer between server and wireless nodes, all these wireless routers are connected to the server through a switch on Ethernet. Ported with a comprehensive OpenWRT embedded Linux OS, the wireless device adopts Broadcom's BCM5354, which is a 802.11b/g Router System-on-Chip, as the core of the



(a) Web Interface Desktop PC Browser



(b) Web Interface on HTC Magic Smart-phone

Fig. 3: WeFiLab Accessed Through Desktop PC and Smartphone

wireless node. It can work on different modes, such as AP/STA, Ad Hoc and so on. Since this chip comes with a USB2.0 host controller, the wireless devices can be extended for some advanced learners, such as 3G, wireless sensor network and so on. Various wireless devices can be used here as server will hide the difference from users through the two-level operations.

The server is implemented as a bridge between user and wireless nodes array. It is connected to the wireless node array through an Ethernet switcher. The server coordinates wireless devices according to users' requests and forwards the response of wireless devices to correspondent clients.

WeFiLab adopts Ajax to build the user-friendly GUI for enabling sufficient interactivity students and the systems. Ajax allows every user with capable web-browser to use WeFiLab without additional plugins, and makes the anywhere anytime wireless networking laboratories to mobile learners come true. Currently, we have tested WeFiLab successfully on several web browsers, such as Google Chrome 9 or above, Firefox 3.6.X or above, IE 8.0 or above, Safari 5. Fig. 3a shows the snapshot of WeFiLab on a web browser of PC. In this figure, we can see the three locked wireless devices and a panel in the bottom which can display the captured frames in real time. WeFiLab can also be accessed through smartphone. Fig. 3b displays a view of WeFiLab on a HTC G2 mobile phone.

WeFiLab uses MySQL as the database server to facilitate the whole system, e.g. storing user information, schedule information, wireless device information and so on. It can also record all the data produced during students' experiments such as operation history, captured

wireless frames, students' answers to the questions and so on.

6 CASE STUDIES

To investigate to what extent WeFiLab can facilitate students' understanding of WiFi and help teachers during the course, we have evaluated WeFiLab through two study cases, where WeFiLab was used as a complement of assignments for two courses in City University of Hong Kong: the course *Mobile Computing* [12] for two semesters: CS4284 2010B (Semester B 2009/10) and CS4284 2011B (Semester B 2010/11), the course *Pervasive Computing* [23] for one semesters: CS4289 2011A (Semester A 2011/12).

In the semester CS4284 2010B, 146 undergraduate computer science students are involved into the experiment about 802.11 Frame, which is described in Section 4.2. After we optimized the design and implementation of WeFiLab, we used it again for the experiment of the course in semester CS4284 2011B and CS4289 2011A, which involves 116 and 53 undergraduate students respectively from computer science.

Both of the study cases in the two semesters are designed following the OBTL scheme [24] in City University of Hong Kong.

6.1 The OBTL Scheme

OBTL (Outcome-Based Teaching and Learning) [24] is a student-centered approach for the delivery of educational programmes where the curriculum topics in a programme and the courses contained in it are expressed clearly as the intended outcomes for students to achieve.

Teaching is then designed to directly encourage students to achieve those outcomes and reflect on the learning process with assessments undertaken. In this approach, teachers act as facilitators, and students should take responsibility and participate actively. It has been widely adopted in universities across the world.

Some glossaries of OBTL that will be used in the following description of the two cases in this section include:

Intended Learning Outcomes (ILO): ILOs state what students are expected to be able to do at the end of the courses/experiments according to a given standard of performance. ILOs are the goals to be achieved through the learning activity.

Teaching/Learning Activities (TLA): TLAs in constructive alignment refer to situations that elicit the appropriate learning activities, whether teacher, peer, or self-initiated; In the two cases of this section, we provided students short sessions of introducing the experiments, demonstrations, some related materials for students learning and let them do the assigned/self-motivated experiments on WeFiLab.

Assessment Tasks (AT): Assessment Tasks are what teachers ask students to do to demonstrate evidence that a particular Learning Outcome has been achieved. Some ATs provided in WeFiLab include letting students conduct an experiment (detecting based on the captured frames on the sniffer), analyze WiFi frames, answer questions at the end of experiments, etc.

All the ILOs, TLAs and ATs of the experiments are aligned, and assessment rubrics (criteria) for the experiment are also provided in the two cases, so students know what they need to do and how they will be assessed for the experiments using the WeFiLab.

6.2 Case 1: 802.11 Frame Capture and Analysis

The first case is the experiment for CS4284 in semester 2010B. This experiment mainly focused on the infrastructure mode of IEEE 802.11 WLAN. The participants of this case study include 146 undergraduate computer science students. They all attended the experiment about 802.11 Frame capture and analysis, which is described in Section 4.2.

6.2.1 Overview

The whole experiment includes two parts: First, students need to capture a certain types of wireless frames by WeFiLab; Second, students need to answer a group of questions according to the captured frames.

The *ILOs* of this experiment is:

- 1) Students will be able to capture wireless frames through WeFiLab;
- 2) Students will be able to decode WiFi frames in Wireless LAN;
- 3) Students will be able to analyze both the information elements and non-information elements in WiFi Frames Body;

The *TLAs* of the experiment includes a session for introduction and demonstration to let students know how to capture real wireless frames, decode & analyze these frames. Before the experiment was officially released, teachers/tutors had explained the issues in Wireless LAN and the IEEE 802.11 frames both in the lectures and tutorials. Some related materials, e.g. one chapter (Chapter 7) in the IEEE 802.11 Specification Document [22], were recommended to students for their self-learning and reference while doing the experiment. A short manual of WeFiLab for this experiment was also provided to ensure that students understand the whole experiment.

To evaluate whether the *ILOs* are achieved for students, the *ATs* of this experiment is to let students answer some questions, which are designed based on the frames captured by them during the experiment, and grading them based on their performance.

6.2.2 Experiment Details

As in this experiment, students only need to capture some frames and they don't need to lock devices explicitly, the *FCFS Scheme (Non-Scheduled)* is used, referred to Section 3.4. When students send to request to the server, the server will assign and activate a sniffer to students. The sniffer will capture all wireless frames on the specified channel. In case that the channel is idle, i.e. no traffic occur within a predefined time period, the server will control a wireless device to take activities such as scanning, association, authentication or transmission in order to generate some wireless traffic.

During this experiment, students were asked to:

Step 1: Login the WeFiLab system and activate a wireless sniffer.

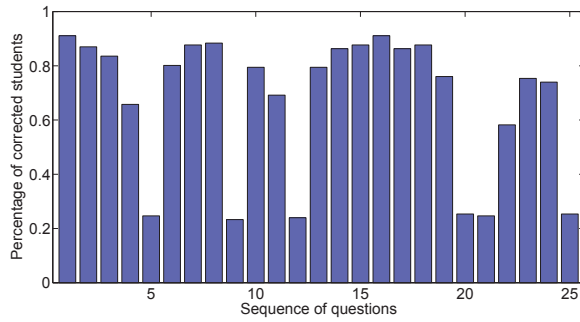
Step 2: Capture a certain types of WiFi frames through the wireless sniffer. The desired frames include Beacons, Probing Request/Response, Association Request/Response, Authentication Request/Response, RTS/CTS, DATA and ACK.

Step 3: Interpret the headers/body of the captured frames using the embedded frame analyzer in WeFiLab and according to the IEEE 802.11 Specification.

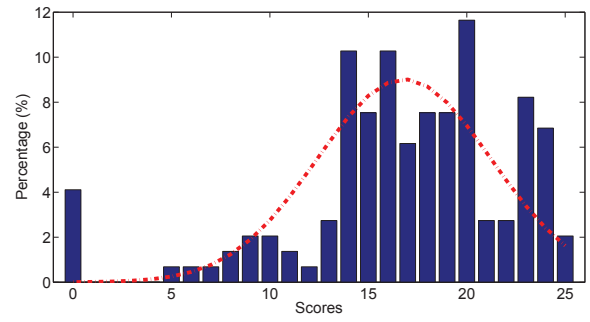
Step 4: Answer 25 questions generated for them according to the frames that they captured and submit the answers through WeFiLab after finished.

Students were allowed to try the whole experiment again whenever they needed before the deadline. The system will store all their submissions but only choose the last version as their final answers for grading.

These 25 questions provided for students include topics such as the association process, data transmission and other issues about Infrastructure mode in Wireless LAN. Table 1 shows the general topics of these 25 questions and some related frame fields. These fields include both information element and non-information elements in IEEE 802.11 frames. As all the wireless frames are captured by students on the air dynamically, the answers of the same question for each student would be different.



(a) Correct Answers of Each Question



(b) Students' Scores Distribution

Fig. 4: Results of the 802.11 Frame Experiments

TABLE 1: Topics and Examples of Corresponding Knowledge Points of Questions in 802.11 Frame Experiment

Questions	Topics	Frame Fields Examples
Q1	WiFi Frames Types	Frames Control Field
Q2-Q6	Beacons of AP	Beacon interval
Q7-Q13	Probing Process	Supported Rates
Q14-Q16	Authentication Process	Authentication Type
Q17-Q22	Association Process	Association ID
Q23-Q25	Data Transmission	Data Duration

6.2.3 Results and Analysis

The whole experiment, including both capturing frames and answering questions, continued for two weeks (April 12~April 26, 2010). All the 146 students in the CS4284 2010B were able to use WeFiLab to complete the whole experiment and submitted their answers successfully before the deadline. Some students didn't complete the experiment first time. According to our system records, most of students tried 2~3 times of the experiment before the deadline, with each student try 2.74 times on average.

Fig. 4 gives the results of students' performance on these 25 questions based on the records of WeFiLab. Fig. 4a shows the percentage of students who provide correct answer on each question. Based on these statistic results, teachers can easily find out on which questions students' knowledge are weak. Using the WeFiLab, teachers can view these students' answers on these questions and locate the weak points of them, which will help teacher to give more explanations to students on these topics in the lecture.

For example, the result in Fig. 4a shows that there are nearly 80 percent students failed on the questions 5, 9, 12, 20, 21 and 25. Among these questions that many students failed, all the first five questions are about the supporting rates in the Beacon/Probing/Authentication/Association frames. The supported rates carried in these frames describe the rates that the particular wireless LAN supports. This information field is encoded as 1 to 8 octets, where each octet describes a single Supported Rate. According to the students who failed on these questions, most of them still unclear about how to calculate the supported rates after reading the IEEE 802.11 Specification. So

the teacher needs to explain the calculation during the lecture course.

The question 25 is about the four-way handshake of data transmission procedure between AP and client, which involves exchanging frames of RTS-CTS-DATA-ACK. This question is to test students understanding of the inter frame space (IFS) and NAV (Net Allocation Vector), such as DIFS, SIFS and duration field in the frames. According to students' answers, many students failed on this question because they don't understand the time spaces and duration clearly. So the teacher may need to reemphasize students understanding on the mechanism of CSMA/CA with RTS-CTS.

Fig. 4b also shows the distribution of all students' final scores of the experiment. The blue bar is the distribution of raw scores. The red dotted curve shows the normal distribution based on the mean and standard variation of the raw scores. It helps teacher to obtain an overview of all the students' performance. We can find that the distribution is almost the classical "bell curve" [25]. According to this result, the teacher can know that most students got 15~21 correct answers. As most students successfully captured the required wireless frames and answering the questions about decoding/analyzing these frames, the two *ILOs* for this experiment are achieved.

6.3 Case 2: 802.11 Ad Hoc Experiment

The second case involves assignments for CS4284 in semester 2011B and CS4289 in semester 2011A. This experiment in the assignment combines part of the experiment in Section 4.1 and the 802.11 Frames in Section 4.2. Totally 169 undergraduate computer science students attended this experiment (116 from CS4284 2011B and 53 from CS4289 2011A).

6.3.1 Overview

The *ILOs* of this experiment is:

- 1) Students can deploy Ad Hoc networks and check the connectivity of the network in WeFiLab;
- 2) Students will be able to decode and analyze IEEE 802.11 frames produced among Ad Hoc network;
- 3) Students will be able to understand both the information elements and non-information elements in IEEE 802.11 Frames;

The *TLAs* of this experiment mainly includes two parts: 1) Deploy an Ad Hoc network in WeFiLab cooperating with other group members; 2) Decode & analyze the frames generated during the experiment and answer some related questions. A detailed manual of the system was also provided for students. The IEEE 802.11 Specification Document [22] was recommended to students for their self-learning. At the end of the experiment, we also give students a chance to provide feedback/questionnaire about the experiment.

The experiment allows students to finish it in a group. Each group is formed on students' own and then their grouping information will be registered to the system. All group members must collaborate with each other to complete the experiment, e.g. controlling one device to work as AP for other members, capturing other members' packets. For each group, a group leader is elected to coordinate the whole experiment among group members. In the course CS4284 2011B, the 116 students are divided into 45 groups by students themselves. Most of the groups included three students, others include two or only one student (29 groups have 3 members, 13 groups have 2 members and 3 groups have only 1 member). In the course CS4289 2011A, the 53 students are divided into 31 groups, including 12 one-member groups, 16 two-member groups and 3 three-member groups.

The *ATs* of this experiment is to let students answer the questions, which are generated according to their captured frames during the experiment, and grading them based on their performance.

6.3.2 Experiment Details

As in this experiment, the number of students are far more than the current wireless devices resources of WeFiLab, both the *Reservation Scheme (Scheduled)* and the *FCFS Scheme (Non-Scheduled)* are used, referred to Section 3.4. For each group, WeFiLab reserved an one-hour slot and three devices based on all students' preference for the experiment. So each group has enough time to do the experiment. And for the time outside the reserved schedule, we use *FCFS Scheme* to allow students who failed competing for the available resources to do the experiment.

The manual provides students detailed instructions during the experiment. Some of the main procedures include:

Step 1: All members login the WeFiLab system. Each member of a group can see the online status of the other members.

Step 2: Group Leader request and lock 3 devices. If the group locked the three devices successfully, the system will also assign a sniffer to the group. The sniffer will capture all the frames generated by the locked devices.

Step 3: The three locked devices will be assigned to each member of the group automatically. Each device will be controlled by only one member. Some members

may need to control more than one devices for those groups with less than three students.

Step 4: Set SSID for each devices; Use broadcast/unicast packets to check the connectivity of the three devices;

Step 5: Choose one device as initiator of the Ad Hoc network, the other devices join the initiator to constitute an Ad Hoc network; Use broadcast/unicast packets to check the connectivity of the three devices;

Step 6: Set new SSID for the initiator and other devices; Use broadcast/unicast packets to check the connectivity of the three devices;

Step 7: The sniffer will capture all the generated frames. The server will simply analyze these frames to get the status of students' experiment. Once the system detected all the operations are finished, it will notify students.

Step 8: Interpret the headers/body of the captured frames using the embedded frame analyzer in WeFiLab and according to the IEEE 802.11 Specification.

Step 9: Answer some questions generated for them according to the frames that they captured and submit the answers through WeFiLab after finished.

A questionnaire is also provided in WeFiLab to let students give their feedback and rate the WeFiLab system after they complete the whole experiment.

Those questions provided for students include topics such as the format of "FrameControl" Field in Frame Header, locating and decoding information elements(e.g. SSID, supported rates, working channel, etc.), the association process among stations, data transmission, the setting of NAV value and other issues about Infrastructure mode in Wireless LAN. Students were allowed to submit the answers whenever they needed before the deadline of the experiment. The system will store all their submissions but only choose the last version as their final answers for grading.

Through the simple experiment and further studying of the captured wireless frames during answering the questions, students are supposed to understand some basic characteristics in Ad Hoc network, such as the role of SSID and initiator for Ad Hoc networks, detailed data transmission process, the broadcast which is not guarded by RTS/CTS mechanism and so on.

6.3.3 Results and Analysis

The whole experiment, including the Ad Hoc experiment, frame analyzing questions and the questionnaire, continued for three weeks (CS4284 2011B: March 22~April 11, CS4289 2011A: September 26~October 14). All students in the two courses were able to use WeFiLab to complete the Ad Hoc network experiment and answer the required questions before the deadline successfully. Table 2 shows more details about the students' information in the two courses.

According to the results, about 88.9% groups from CS4284 2011B and 87.1% groups from CS4289 2011A complete the Ad Hoc Network experiment during their

TABLE 2: Statistics of the Ad Hoc Experiment

	Num. of Students/Groups	
	CS4284 2011B	CS4289 2011A
Num. of Students	116 students	53 students
Num. of Groups	45 groups	31 groups
Complete Ad Hoc Experiment	45 groups (100%)	31 groups (100%)
Complete Experiment During the Schedule Slot	40 groups (88.9%)	27 groups (87.1%)
Answer Questions	116 students (100%)	53 students (100%)
Finish Questionnaire	108 students (93.1%)	53 students (100%)

TABLE 3: Students Try Count and Time Spent on Experiment

	No. of Attempts		Total Spent Time(s)	
	2011B	2011A	2011B	2011A
Min	1	1	253	283
Max	16	10	5149	8342
Average	3.87	3.68	1491	2040
Median	3	3	982	1634

reserved one hour slot (Scheduled). The others finished the experiment using the *FCFS Scheme*. Table 3 also shows how many times students retried to do the experiment and the total time period they locked the wireless devices. Most of the groups tried the experiment for 3~4 times and finished the Ad Hoc experiment using 15~30 minutes.

Fig. 6 is the overall statistics of Students and the Clients information for both the courses CS4284 2011B and CS4289 2011B. Fig. 6a shows the places where students do the experiment according to the questionnaire. It is worth to note that about 38% students do their experiment at home, and the 4% of “Others” in the figure includes companies (part-time students), students’ residences and so on. Besides the questionnaire that students provided, WeFiLab also record students login IP, from which we find that about 37.3% students’ login activity are taken place outside the university campus. Furthermore, Fig. 5 shows the distribution of login time of all students based on the hour of a day (00:00~23:59) during the whole three weeks. Though around 50% students’ login activities appears during 13:00~16:00, their activity still distributed in a large range of a day. All these phenomena are mainly because that WeFiLab enables students to study anytime anywhere as long as there is Internet service. And the diversity both on the place and time dimension increase the ability of WeFiLab to reuse wireless devices resource for more students.

Fig. 6b and 6c are the web browsers and client OS that students used while doing the experiment. All these information are captured through the HTTP User Agent [26]. The various of web browsers and OS used by students during doing the experiment, which contain most of the current popular browsers and OS, shows that WeFiLab has much compatibility.

Fig. 6d shows the type of clients based on the questionnaire. The goal of WeFiLab is to let students do experiments on real devices through web page. It allows

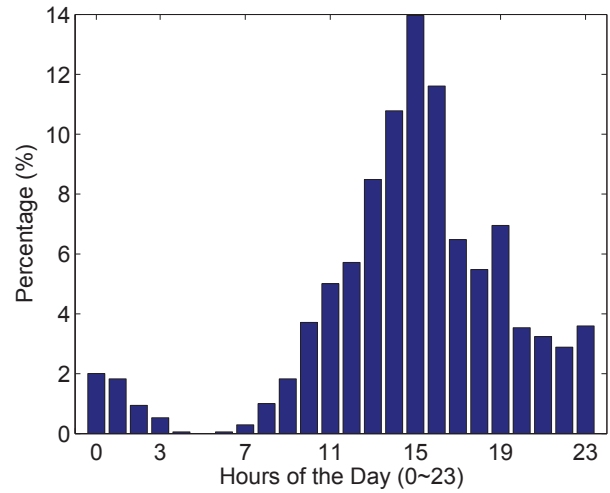


Fig. 5: Overall Diversity of Students' Login Time for Both Courses

students to use both desktop computers and mobile devices to access the system. The results show that most of the students used desktop/laptop PC, or Mac to do the experiment. However, there are still 1% records which students used WeFiLab on iPad and smartphones, i.e. iPhones. The iPad has the size and weight fell between those of smartphones and laptops, but shares the same OS as the iPod Touch and iPhone.

To assess the performance and design of the prototype WeFiLab, there is also a group of statements in the questionnaire to let students rate the experiment based on their experience during the experiment on a scale between 0 and 10, where 0 indicates “very bad” and 10 indicates “excellent”. Table 4 shows students feedback on these statements. Most students think that WeFiLab provides very good functions for their study (i.e. the item 3 in Table 4, whose average score is over 7).

TABLE 4: Overall Students' Rates on WeFiLab for Both Courses

Item	Average Scores(0~10)
User Interface	7.11
Response time of each operation	7.20
Functionality	7.07
Interaction with Users	7.01
Design of WeFiLab	6.89
The Team Work in WeFiLab	7.37
Feeling on Online Wireless Lab	7.42

7 DISCUSSIONS AND FUTURE WORKS

Though WeFiLab is designed for students to have hand-on experience of wireless network, its architecture and design rationale, e.g. the two-level operations and schedule scheme, can be applied to other fields that require students to have experience on real devices, e.g. RFID, Internet Switcher/Router experiment, etc.

Some challenging problems and future works about WeFiLab are described as follows:

(a) **WeFiLab on Smartphones:** As we mentioned in the previous sections, the WeFiLab platform adopts Ajax

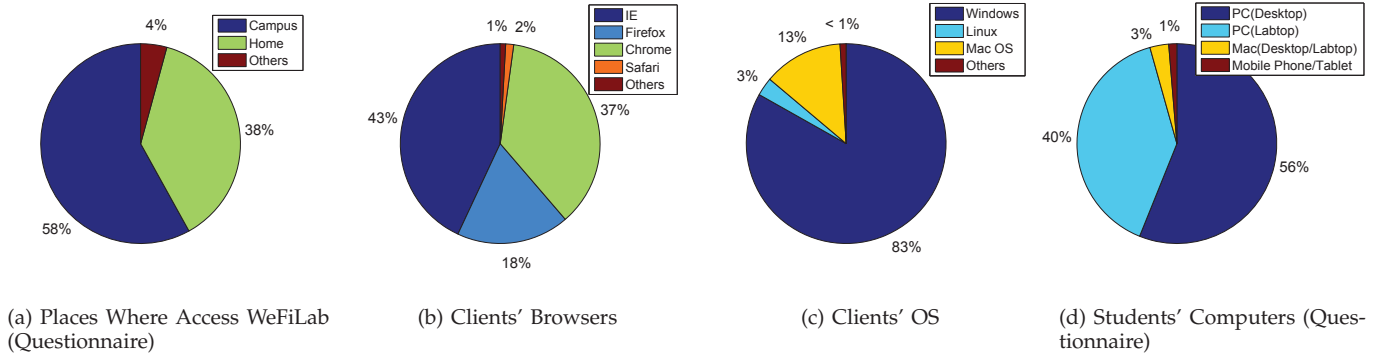


Fig. 6: Overall Statistics of Students' and Clients' Profile for Both Two Courses

for the design of GUI, which is compatible with many current mobile web browsers of smartphones. However, smartphones still have the following limitations, which need to be considered for the further development of WeFiLab: 1) Small screen size: The screen size are varied for different types of mobile phones and smaller compared to PC and laptop. 2) Limited computing and memory resources; 3) Limited network ability: Depending on the condition of mobile phone, they may have low bandwidth and long packet delay. Furthermore, due to the mobility of smartphones, clients may experience unexpected connection interruptions.

(b) Interference: One of the great challenges while developing WeFiLab is the interference among all the wireless devices. We have tried to reduce the transmission power of the wireless module as low as possible without influence students' experiments. Though the interference in WeFiLab is not so obvious under the current settings currently, we believe with the increasing on the number of wireless devices, the interference would be a serious problem and challenge to work on in future. To separate devices into different locations, which are far away enough not interfering with each other, is also an option. More optimization on the design and more efficient schedule are need to be conducted on WeFiLab.

(c) Relationships of Roles: We will continue to focus on improving the relationship among students and teachers/instructors. There are two aspects about this relationship [27]: (1) Student-Student relationship: This relationship is about the collaboration among students who might also be remote from each other. Currently, WeFiLab already allow students to collaborate with each other while doing the experiments. WeFiLab will further enable students to collaborate more efficiently to finish large experiments together or help each other during experiments. (2) Student-Instructor relationship: Similar to the relationship between students, we aims to provide more interaction among students and teachers/instructors when the experiments are carrying on, rather than just providing the results and histories of students' performance after the experiments. For example, it would be very useful to let teachers trace students' real-time activities and status during doing experiments. The sufficient information can enable teachers find the

weak points of students or providing some suggestions before students making mistakes.

(d) More Technologies: Some more complex configuration of wireless network can be provided to serve some advanced students for further learning, e.g. graduate students. Also the wireless devices in WeFiLab can be extended (through the USB port) to support other networks, such as the third generation (3G) wireless cellular networks (e.g. HSPA), wireless sensor network to fulfill the interest of students for new technologies.

(e) Education & Research: The WeFiLab is designed as an education platform currently. However, with some extension, it can also be used as a platform for academic researches. The operations would be much more complicated for research purpose compared to education. So some advanced features would be provided, such as scripts-support for the control of each step of experiments, programming on wireless devices (compilation/uploading/running), more scientific process on experiment data and so on. Furthermore, a group of APIs for the system may also need to be developed and provided in order to allow access and reuse by third party educational applications.

8 CONCLUSIONS

In this paper, we present the platform of WeFiLab, an on-line web-based laboratory for undergraduate students. WeFiLab mainly focuses on the wireless networking technologies (WiFi). It aims to provide undergraduate students for hand-on experience of doing wireless network experiments on real devices. With the web-based GUI, it enables students to do experiment on web page via Internet anytime anywhere. WeFiLab uses the structure of two-level operations to coordinate the communication between clients and wireless devices, allowing students' experiment on real wireless devices. The Schedule Scheme of WeFiLab also facilitating the efficient use of these devices.

We've also implemented a prototype of WeFiLab system and evaluated it in two study cases. In both of the two study cases, which totally involved 315 undergraduate computer science students, WeFiLab was successfully used as a complement for the course of Mobile Computing and Pervasive Computing in City

University of Hong Kong. The case study results show that WeFiLab not only improve students' understanding about the WiFi but also help teachers to locate the weak areas of students' knowledge and reemphasize these topics in the lecture course.

9 ACKNOWLEDGEMENT

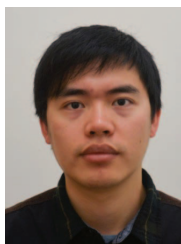
This work is supported in part by CityU Teaching Development Grant (TDG(CityU)) No. 6000325; CityU Applied R & D Funding (ARD) Nos. 9681001, 6351006 and 9667052; and NSF (China) No. 61070222.

REFERENCES

- [1] N. Sarkar, "Teaching computer networking fundamentals using practical laboratory exercises," *IEEE Transaction on Education*, vol. 49, no. 2, pp. 285–291, 2006.
- [2] N. Linge and D. Parsons, "Problem-based learning as an effective tool for teaching computer network design," *IEEE Transaction on Education*, vol. 49, no. 1, pp. 5–10, 2006.
- [3] A. Holzinger, A. Nischelwitzer, and M. Meisenberger, "Mobile phones as a challenge for m-learning: Examples for mobile interactive learning objects (MILOs)," *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 307–311, 2005.
- [4] "Emulab." [Online]. Available: <https://users.emulab.net/trac/emulab/wiki/wireless>
- [5] L. Fabrega, J. Massaguer, T. Jove, and D. Merida, "A virtual network laboratory for learning ip networking," *Proceedings of the 7th annual conference on Innovation and technology in computer science education*, pp. 161–164, 2002.
- [6] S. Liu, W. Marti, and W. Zhao, "Virtual networking lab (vnl): its concepts and implementation," *Proceedings of the 2001 American Society for Engineering Education Annual Conference & Exposition*, 2001.
- [7] J. P. Blazquez, J. A. Moreno, and J. H. Joancomarti, "An integrated structure for a virtual networking laboratory," *IEEE Transactions on industrial electronics*, vol. 55, no. 6, pp. 2334–2342, 2008.
- [8] J. DeHart, F. Kuhns, J. Parwatikar, J. Turner, C. Wiseman, and K. Wong, "The open network laboratory," *Proceedings of the 37th SIGCSE technical symposium on Computer science education*, vol. 38, no. 1, pp. 107–111, 2006.
- [9] "Open network laboratory." [Online]. Available: <https://onl.wustl.edu/>
- [10] X. Chen, G. Song, and Y. Zhang, "Virtual and Remote Laboratory Development: A Review," *Proceedings of Earth and Space 2010*, pp. 3843–3852, March 2010.
- [11] B. Alhalabi, M. K. Hamza, S. Hsu, and N. Romance, "Virtual labs vs remote labs: Between myth & reality," *Proceedings of Florida Higher Education Consortium 7th Statewide Conference*, Nov 1998.
- [12] "CS4284 Mobile Computing." [Online]. Available: <http://eportal.cityu.edu.hk/bbcswebdav/institution/APPL/Course/Current/CS4284.htm>
- [13] R. Mashburn, "Virtualized computer labs and the software tools to make it so," *Innovative Techniques in Instruction Technology, E-learning, E-assessment, and Education*, pp. 295–303, Aug 2008.
- [14] "List of network simulator." [Online]. Available: <http://www.idisia.ch/andrea/sim/simnet.html>
- [15] G. Canfora, P. Daponte, and S. Rapuano, "Remotely accessible laboratory for electronic measurement teaching," *Computer Standards and Interfaces*, vol. 26, no. 6, pp. 489–499, 2004.
- [16] W. Hutzler, "A remotely accessed HVAC laboratory for distance education," *International Journal of Engineering Education*, vol. 18, no. 6, pp. 711–716, 2002.
- [17] M. Cooper, A. Donnelly, and J. Ferreira, "Remote controlled experiments for teaching over the internet: A comparison of approaches developed in the pearl project," *Proceedings of Annual Conference Australian Soc. Computers in Learning in Tertiary Education*, 2002.
- [18] L. D. Feisel and A. J. Rosa, "The role of the laboratory in undergraduate engineering education," *Journal of Engineering Education*, vol. 94, pp. 121–130, 2005.
- [19] N. Lehlou, N. Buyurgan, and J. R. Chimka, "An online rfid laboratory learning environment," *IEEE Transactions on learning technologies*, vol. 2, no. 4, pp. 447–452, Oct 2009.
- [20] M. A. Bochicchio and A. Longo, "Hands-on remote labs: Collaborative web laboratories as a case study for it engineering classes," *IEEE Transactions on learning technologies*, vol. 2, no. 4, pp. 320–330, 2009.
- [21] D. Lowe, S. Murray, E. Lindsay, and D. Liu, "Evolving remote laboratory architectures to leverage emerging internet technologies," *IEEE Transactions on learning technologies*, vol. 2, no. 4, pp. 289–294, 2009.
- [22] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications." [Online]. Available: <http://standards.ieee.org/getieee802/download/802.11-2007.pdf>
- [23] "CS4289 Pervasive Computing." [Online]. Available: <http://eportal.cityu.edu.hk/bbcswebdav/institution/APPL/Course/Current/CS4289.htm>
- [24] "OBTL at City University of Hong Kong." [Online]. Available: <http://www6.cityu.edu.hk/obtl/index.asp>
- [25] S. A. Miller, "What does it mean to curve grades?" [Online]. Available: <http://academics.hamilton.edu/biology/smiller/curve.html>
- [26] "User Agent." [Online]. Available: http://en.wikipedia.org/wiki/User_agent
- [27] D. Lowe, S. Murray, E. Lindsay, and D. Liu, "Evolving remote laboratory architectures to leverage emerging internet technologies," *IEEE Transaction on Learning Technologies*, vol. 2, no. 4, pp. 289–294, 2009.



Lin Cui is currently a PhD candidate in the Department of computer Science of City University of Hong Kong. He received his Bachelor degree from Shandong University in 2007, and the master degree from Harbin Institute of Technology in 2009. His general research interest falls in the area of mobile computing systems, wireless ad hoc/sensor network.



Fung Po Tso is currently a SICA research fellow in the School of Computing Science, University of Glasgow, UK. He received his BEng degree in Computer Engineering, MPhil and Ph.D degrees in Computer Science from City University of Hong Kong (CityU HK), in 2005, 2007 and 2011 respectively. His research interests include cloud data center networks, mobile computing, distributed computing and cyber-physical system.



Di Yao received the BSc and MSc degrees from Northeast Dianli University and National University of Defense Technology in 2005 and 2008 respectively. She is currently working in City University of Hong Kong as RA. Her research interests include the design and applications of wireless sensor network systems, system performance measurement and optimization.



Prof. Weijia Jia is currently a full Professor in the Department of Computer Science and the Director of Future Networking Center, ShenZhen Research Institute of City University of Hong Kong (CityU). His research interests include next generation wireless communication, protocols and heterogeneous networks; distributed systems, multicast and anycast QoS routing protocols. In these fields, he has a number of publications in the prestige international journals (IEEE Transactions, e.g., ToN, TPDS, TC, TMC etc.), books/chapters and refereed international conference proceedings (e.g. ACM CCS, WiSec, MobiHoc, SenSys, ICDCS, INFOCOM etc.).