International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 5 (2013) pp. 383-404

Intelligent Agent based Student-Staff Scheduling System

Dwayne Parchment[1], Suresh Sankaranarayanan [2,3]
[1]Mona Institute of Applied Science, University of West Indies, Kingston, Jamaica
dparchment@gmail.com
[2]Computing & Information Systems, Institut Teknologi Brunei, Jalan Tutong Link , Gadong, Brunei
[3]Department of Computing, University of West Indies, Kingston, Jamaica
pessuresh@hotmail.com

Abstract: Creating a Schedule that is fair and is suitable to all parties involved is a difficult task at many universities. It is time-consuming and sometimes doesn't work in the favor of all parties involved. People have been using various computerized appointment systems like Microsoft outlook and so to solve the scheduling problem in Universities and colleges. In addition to the use of ICT in appointment system, there is still involvement of human being towards scheduling of appointment. In the field of computing, the use of software agents to automate task has been on the lips of every computer scientist. Multi-agent system technology has become a popular paradigm over the few years in the design modeling and implementation of software solutions. So, based on that there has been a quite an amount of work done on the use of Software agents towards automating the scheduling system from desktop which is evident from the literature. But still there exist drawbacks in the existing system. The use of Mobile Phones has been an ever growing phenomenon since the early 1990s. The main stakeholders in the educational system including lecturer and students have been partakers of this phenomenon where the cell phone has become a part of their daily lives with the inclusion of other ICT devices into the educational sector. So taking this in mind work was carried out towards scheduling appointment for students using Agents from Android handset recently. There were a few drawbacks in the existing system, like no provision of scheduling between lecturer and Lecturer, and it did not take into consideration the time span between the scheduling, rescheduling and cancellation of appointment and the actual start of the appointment. Another drawback of the previous work was the fact that the appointment diary of the lecturer could not be seen. Last but not the least negotiation between Scheduler agent and lecturer agent can be carried out only if the lecturer's mobile handset is on as the fuzzy preference logic for appointment negotiation resides on the mobile side which is a bit of drawback. This research here seeks to alleviate the above mentioned problem by incorporating software agents on Android enabled handset into the educational arena in an effort to solve scheduling appointments woes. Also in this research, it allows the scheduling and cancellation of appointments based on some time period validation. The Smart agents utilize the properties of autonomy and mobility to intelligently schedule appointments on behalf of the lecturer. JADE-LEAP on the latest Android handset has been the selection of choice to develop the proposed system.
Keywords: ICT, Agents, JADE-LEAP, Android

I. Introduction

Education [1] is by far the best investment for any individual in order to have more opportunities to get a job which gives them satisfaction. The fact that education is such a vital aspect of an individual life means that time must be spent in attending to the intricate needs of such process. One extremely important aspect of the education process is setting up and attending appointments with your lecturers and professors. Appointments that are done manually can be tedious and time consuming as seen by experience [2][3]. The pen-and-paper method is the oldest technique of scheduling appointments for most institutions. The scheduling of an appointment between staff and students are done sometimes through a secretary with a central calendar. In this manual method, the staff whose time is being booked must check with secretary or check diary regularly to find out what their schedule is, and to let the secretary know the available times. Also, the students who want to meet with the lecturer are expected to call the secretary or lecturer to check if the appointment is still on. There are times when a lecture has scheduled meeting with a student and has another pressing matter too. When this happens, appointments have to be cancelled or re-Scheduled. It is the responsibility of the lecturer or secretary to notify the student of this. However, this intended notification can become loss in numerous tasks that have to be done on a daily basis and hence student will not be given proper and timely notification. These appointment inefficiencies will definitely frustrate the student and so this method can be untenable in a large university.

Since time to be spent for appointment is a major constraint for any individual these days, an electronic computerized scheduling system will most likely solve some inefficiency in the manual scheduling system. Computerized scheduling systems manage appointments and meetings. A number of solutions have been put forward as it relates to the use of ICT in making the scheduling of appointments in education more efficient. Also each day more educational institutions are coming on board in having ICT managing their appointments. One common software that be used to achieve this is Microsoft Outlook. But still there are some drawbacks which include human involvement towards recording of appointment, cancellation, confirmation and reschedule etc. Of course these systems possess one advantage which is the automatic reminder only. Though ICT has played a good role in appointment scheduling, still there is no automation in appointment scheduling system. Now with these in mind Agent based Scheduling system been developed towards scheduling appointment by means of Agents with very minimal human involvement. These

scheduling system are more towards computer based system.

Recently there has been lot of development towards Mobile technology like Blackberry, IPAD, and Android and so which people use it in day to day life. Students and lectures are no exception in this. So taking these into consideration work has been carried out in the use of Agents in student Scheduling in latest Mobile handset i.e Android called Negotiator which allows students to schedule appointment with lecturer by means of agents automatically but still possess some deficiencies like no provision of scheduling between lecturers, no consideration of time span between cancellation, rescheduling and so. Also appointment diary of lecturer could not been seen and also negotiation of scheduling between lecturer and student can be carried out only if mobile handset is up as fuzzy preference logic lies on the mobile handset only. Taking these into consideration, we here has developed Agent based student-staff scheduling system which overcomes the drawbacks of the system developed before on Android i.e Negotiator.

The paper is organized in sections as follows. Section 2 provides details on Computerized Appointment systems. Section 3 gives details on Intelligent Agents and Agent based Appointment systems . Section 4 gives details on the proposed system. Section 5 gives the implementation details using JADE-LEAP in Android and Section 6 is conclusion and future work.

II. Computerized Appointment System

Nowadays, the trend is clear that technologies are taking place in teaching and learning [4]. Information and Communications Technology (ICT) have become a major part of our educational system. This is often attributed to Electronic and Mobile Learning. Using computers in automating appointments is just one of the major uses of ICT. Saving time, money and increased productivity are some of the benefits of a computerized appointment system.

Computerized scheduling systems [5] are method of using scheduling algorithms and rules to help multiple people manage appointments and meetings. A number of solutions have been put forward as it relates to the use of ICT in making the scheduling of appointments in education more efficient. Also each day more educational institutions are coming on board in having ICT managing their appointments. One common software that be used to achieve this is Microsoft Outlook. At present University of Calgary is offering an online course in Microsoft Outlook 2003. Other Computerized Appointment system includes ScheduleView™ and Appointment schedule software just to name a few. Even though all these systems are in place, still the appointments are not managed properly using this software as there is still human involvement to it towards recording of appointment, cancellation, confirmation and reschedule etc. Of course these systems possess automatic reminder and so. Also smart phones such as Black Berries and IPhone has become a way of life in keeping in touch with friends, colleagues, business and families of the day. So in order to make appointment scheduling system efficient and managed in an intelligent way with very least human involvement, research was carried on using Agent in appointment scheduling using Smart phones. But before we go into those details and our proposed Research, we will look briefly into Intelligent Agents.

III. Intelligent Agents

The idea of an agent was originated by John McCarthy in the mid-1950, and the term was coined by Oliver G. Selfridge a few years later. According to [6] "an agent would be a 'soft robot' living and doing its business within the computer's world." Although this is one definition, there exist numerous definitions or ideas about what an agent is. Agents are looked at as "a persistent software entity dedicated to a specific purpose" [7]; "computer programs that simulate a human relationship by doing something that another person could do for you" [8] ;"a software entity to which tasks can be delegated" [9]. However, an agent is also defined as a self-contained program capable of controlling its own decision, making and acting based on perception of its environment, in pursuit of one or more objectives [10].

The job of agents, range from simple task such as personal Information management, to complex industrial and commercial processes. For an agent to be acceptable, it must possess some of the attributes described in [11] and [12]. Some of the attributes are autonomy, social ability, responsiveness, proactiveness, reactivity, collaborative behavior, knowledge-level, temporal continuity, personality, mobility and adaptivity. The attributes that an agent possesses determines its functionality. Various categories of agents include Integration, Coordination, Mobility, Assistants, Believable and Communicative. Mobile Agents [13] to which our research is mostly geared, can physically travel across a network, and perform tasks on machines that provide agent hosting capability. This fascinating ability allows processes to migrate from computer to computer, for processes to split into multiple instances that execute on different machines, and return to their point of origin [14].

With the advent of computers and in a changing technological area for an educational sector to be efficient and of the highest and competitive standard, it must employ the new trends in Information Technology. One of these new trends is Agent Technology. Agent Technology will greatly provide efficiency and speed in the delivery of educational courses and in educational related tasks by eliminating or reducing the human deficiencies that comes with the use of humans. If these human deficiencies are eliminated or reduced then the education sector can see more than a marginal improvement in education and hence a brighter future for a country. With all these in mind, we will look into the research carried in appointment scheduling using Agents which was the motivation for our research.

A. Agent based Appointment Scheduling System

Various types of software have been designed to solve the problems of scheduling in many industries. Some of these are discussed here now.

PASHA II [15] initiates an appointment by sending all the free time slots for a person to all the people who are intended to be invited to the meeting. PASHA II was

designed to replicate how a secretary manages appointment in the real word. So it was created to act more or less as a personal assistant. A Case Study in designing suitable communication protocols for software agents is available in [16]. Reference [17] is concerned with negotiation which has been described as a process through which multiple self-interested agents can reach agreement over the exchange of scarce resources. Here the focus is on settings where agents have limited or uncertain information, precluding them from making optimal individual decisions. Similarly reference [18] speaks on appointment scheduling as a problem faced daily by many individuals and organizations wherein cooperating agent systems have been developed to partially automate this task. Email was seen as the means to get many persons to be involved and Cosma, which is a fully implemented German language server for existing appointment scheduling agent systems was introduced as a system that can cope with multiple dialogues in parallel, and accounts for differences in dialogue behavior between human and machine agents.

An Agent Notification for Appointment System was developed in order to ease and facilitate user to make an appointment or announcement through a teaching web and be notified via GSM Network using Short Message Services. The Notification Agent was responsible for the notification service on behalf of recipients. Other researchers such as [19-21] all added to the extensive work done in the field of scheduling. All of these researches testify to the fact that the area of appointment and scheduling is of extreme importance and are still being investigated. Recently negotiation using agents to implement a priority based appointment system that allows students to schedule appointment with lecturers in University and colleges has been developed. This [22] was done on latest Android handset. Also development of an appointment scheduling system, for hospital appointment for patients and hospital staff meeting the doctor for general medication has been reported [23]. In this work priority was used towards appointment to be given to senior citizen, disable people and so on. This was of course carried out in Android handset too.

Taking all the research that has so far been done, it is seen that the agent development towards appointment scheduling been targeted more towards scheduling of meeting and so on. There has been no work reported in the literature towards agent based appointment scheduling for Educational sector particularly on mobile handset except for the one developed by Cuffe and Suresh [22]. This proposed system builds upon that work of Cuffe and Suresh [22] in using agents to implement a priority based appointment system that allows students to schedule appointment with lecturers and between lecturers, were negotiated based upon priorities and there were guidelines that stipulated the cancellation, re-scheduling of appointments. These are discussed in forthcoming sections.

IV. Intelligent Agent based Student-Staff Scheduling System

Creating a Schedule that is fair and is also suitable to all parties involved, is a difficult task at many universities. It is time-consuming and sometimes doesn't work in the students favor as other engagements or later engagements might take precedence over the already scheduled agreement and hence push back the students' schedule. In the context, the proposed system [24] will take advantage of the use of an intelligent agent to carry out the following tasks so that all stakeholders mutually benefit.

- Scheduling students and lecturers,
- Scheduling HOD's and lecturers as well
- Scheduling lecturers with other lecturers

The main purpose of this proposed system [24] is to reduce the waiting time needed towards appointment scheduling between lecturers and students and also appointment scheduling between fellow lecturers too. The success of this system has been made possible through the use of a mobile phone handset capable of running the Android Operating system and a remote Database Server running Microsoft SQL Server 2005. The proposed system utilizes the functionality of software agents to perform its features. This software agent also performs functions such as scheduling, re-scheduling, update and cancellation of appointments. All these features are implemented using the JADE-LEAP agent development kit while testing the validation of the system function. Now before going into the implementation details, the architecture and algorithm of the proposed system [24] will be discussed.

A. Architecture of Agent based System

Fig.1. Shows the architecture of the proposed Agent based system developed for student staff appointment. It consists of three Agents viz., Student Agent, Lecturer Agent and Scheduler Agent. The details of the functionalities of these Agents and such other things are discussed now which also covers the corresponding tasks of their human counterparts.

- **Student Agent**: This agent is used by the student to authenticate logon request and request appointment with the lecturer agent. This agent resides on mobile handset
- **Lecturer Agent**: This agent is used by the lecturer to view request for appointments and to send request to other lecturers for meetings. This agent also resides on mobile handset
- **Scheduler Agent**: The main purpose this agent is to accept requests for new appointments or rescheduling of appointments already fixed, to determine the validity of the request and to save the changes in the database and informing the requesters, the status of their request.
- **Directory Facilitator (DF)** – The purpose of the directory facilitator is to store information on all the agents presently registered within the environment. In doing this, the scheduler will know which student and lecturer are registered in the system at any point in time.
- **Mobile Device** – A smart phone with the Android Based Operating System will be used as the main interface for our system when it is implemented.

In the agent based system the users i.e. student and lecturer have got to be authenticated by means of their logon credentials. The appropriate agent i.e. Lecturer agent or student agent will take the relevant data to the

scheduler agent to negotiate the task required by looking into general schedules like lectures, meetings, examination, appointment schedules and appointment preference. This appointment scheduling is done in similar fashion for both – student meeting a lecturer and a lecturer meeting a lecturer, for appointment.. There is a very minimal human involvement in the system of lecturer or student rescheduling or cancelling the appointment taking into consideration appointment preferences, timing constraints and so. All the data related to each transaction is stored in a data repository. The successful implementation takes into account a Wide Area Network. This is needed to allow the agents to interact with each other and also the interaction to the database which is located on a Local Area Network on the university campus.

B. Algorithm

The flow charts pertaining to the proposed process have been shown in parts as Figs 2(a), 2(b), 2(c) and 2(d). These will now be used for developing the algorithm used.

- User i.e. the student or the lecturer logs in and based on their status (student or Staff) the appropriate main screen will be presented.
- Upon successful logging in, the user is presented with the option of creating an appointment or viewing the appointment diary to see already scheduled appointments.
- If the create appointment option is selected, the user gets the chance to search the database and select the lecturer to whom appointment is to be made with
- Once the lecturer is selected, user selects the date and time for appointment. If the date and time pass the required validation test then the user selects the reason for the appointment. If the date and time does not pass the required validation test then the user is prompted with a error message and is required to correct date and or time.
- When the validation test is passed and the reason for the appointment is selected, the selection summary screen is displayed for confirmation and then the data is sent to the scheduler agent for negotiation.
- The result of the negotiation is displayed on the lecturer phone and then on the user phone
- If the appointment diary option is selected, the user will be shown the list of the future appointments; here the user can select an appointment for cancellation or re-scheduling.
- When the user selects the appointment for cancellation,
- another validation test is performed. The test checks to see difference between the current time and the schedule start of the appointment is less than two (2) hours; if it is less then the user is prompted with a error message indicating that the appointment cannot be cancelled; if the time is two (2) hours or more then the user is allowed to cancel the appointment. The result of the transaction is displayed on the lecturer phone and then on the user phone.
- When the user selects the appointment for re-scheduling, the option is given to enter a new date and time; after this is done a validation test is carried out to check if the selected appointment can be re-schedule. The time period between current time and start time of new time must be four (4) hours or more; If the validation test is not passed then the user is presented with a error message indicating the error and be required to change date and or time; if the test is passed then the user is taken to the summary selection screen to confirm selections before the data is passed on to the scheduler agent. The result of the transaction is displayed first on the lecturer screen and then on the user screen.

In the algorithm above, the intelligence of the Scheduling System is shown in the actions of the negotiator agent that manipulates the database. Here the agent will determine the allocation of the time slots based on the priority selected by the user. If more than one user selects the same time slot then the user who has the highest priority will be given the preference. The implementation of our system carried out using JADE-LEAP tool is discussed in the forthcoming section.

V. Implementation using JADE-LEAP

The implementation of the proposed Intelligent Agent based Scheduling System for Student Staff Appointment was done using the Java Runtime Environment, Eclipse Helios with Android plug-in, JADE 4.1 to create and manage the agents. JADE LEAP 4.0 was used for the implementation on the mobile device. Android 2.2 SDK was used for creation of the emulator. Microsoft SQL Server 2005 was used as the backend for the storage of relevant data for the system.

One of the most widely used mobile OS these days is ANDROID [25]. Android does a software bunch comprise not only operating system but also middleware and key applications. This Operating System was developed by Android Inc., which was founded in Palo Alto of California, U.S. by Andy Rubin, Rich miner, Nick sears and Chris White in 2003. Later Android Inc. was acquired by Google in 2005. After original release there have been a number of updates to the original version of Android. This powerful Operating system supports a large number of applications in Smart Phones. The Hardware that supports Android is mainly based on ARM architecture platform. Some of the current features and specifications of android include Application Framework, Dalvik Virtual Machine, Integrated Browser, Optimized Graphics, SQLite, Media Support, GSM Technology, Bluetooth, Edge, Wi-Fi, 3G, Camera, GPS, Compass etc.

The presence of an online store (online market) makes Android an excellent Operating System of choice for many developers. Android users can select, and download applications developed by third party developers and use them. Not only can users download applications but Android is available as open source for developers to develop applications which can be further be used for selling in android market.

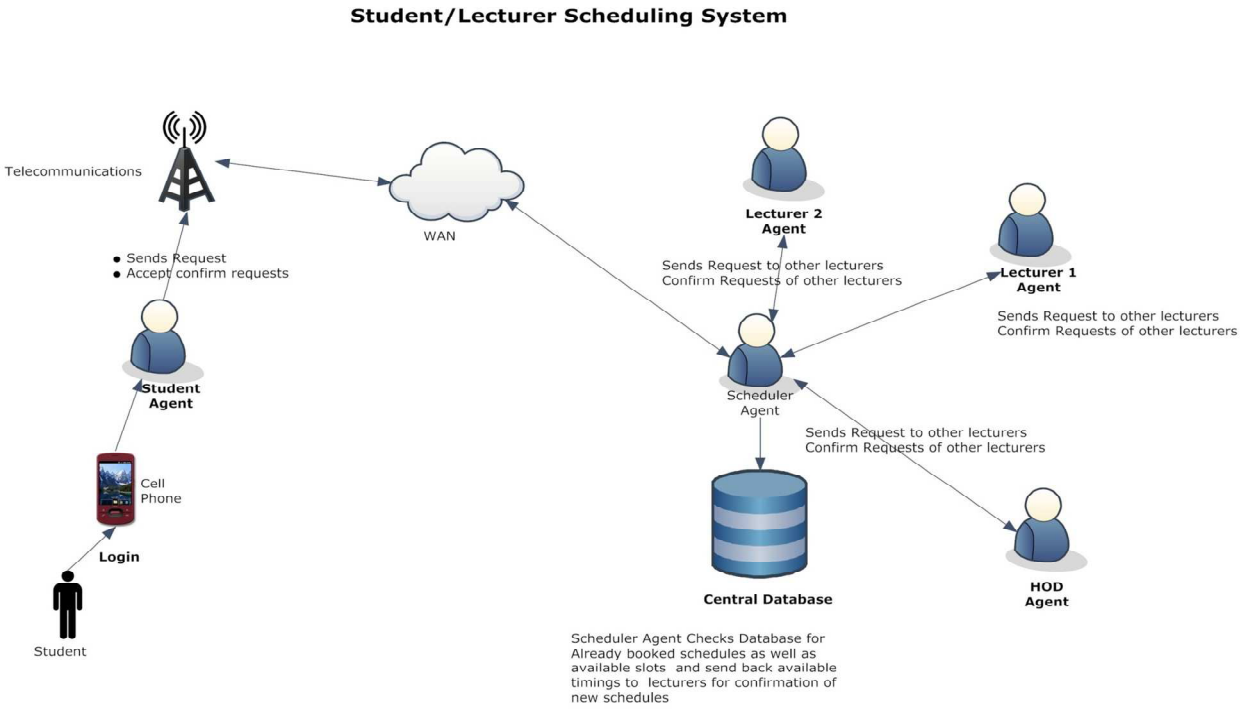

Figure.1 Architecture of Agent based Student- Staff Appointment System

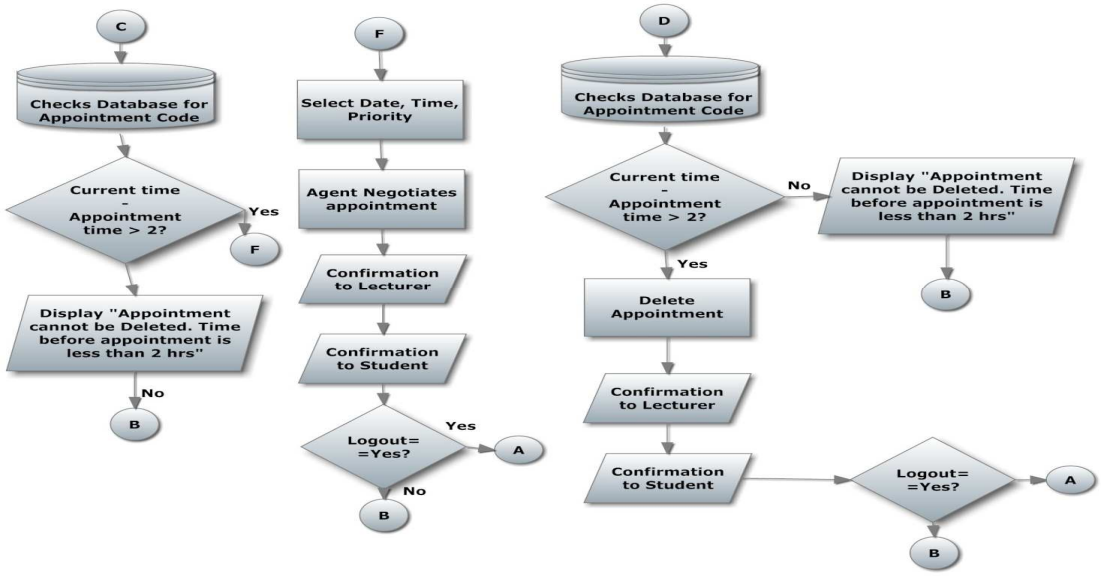

Figure.2(a) Flow Chart of Agent based Appointment System

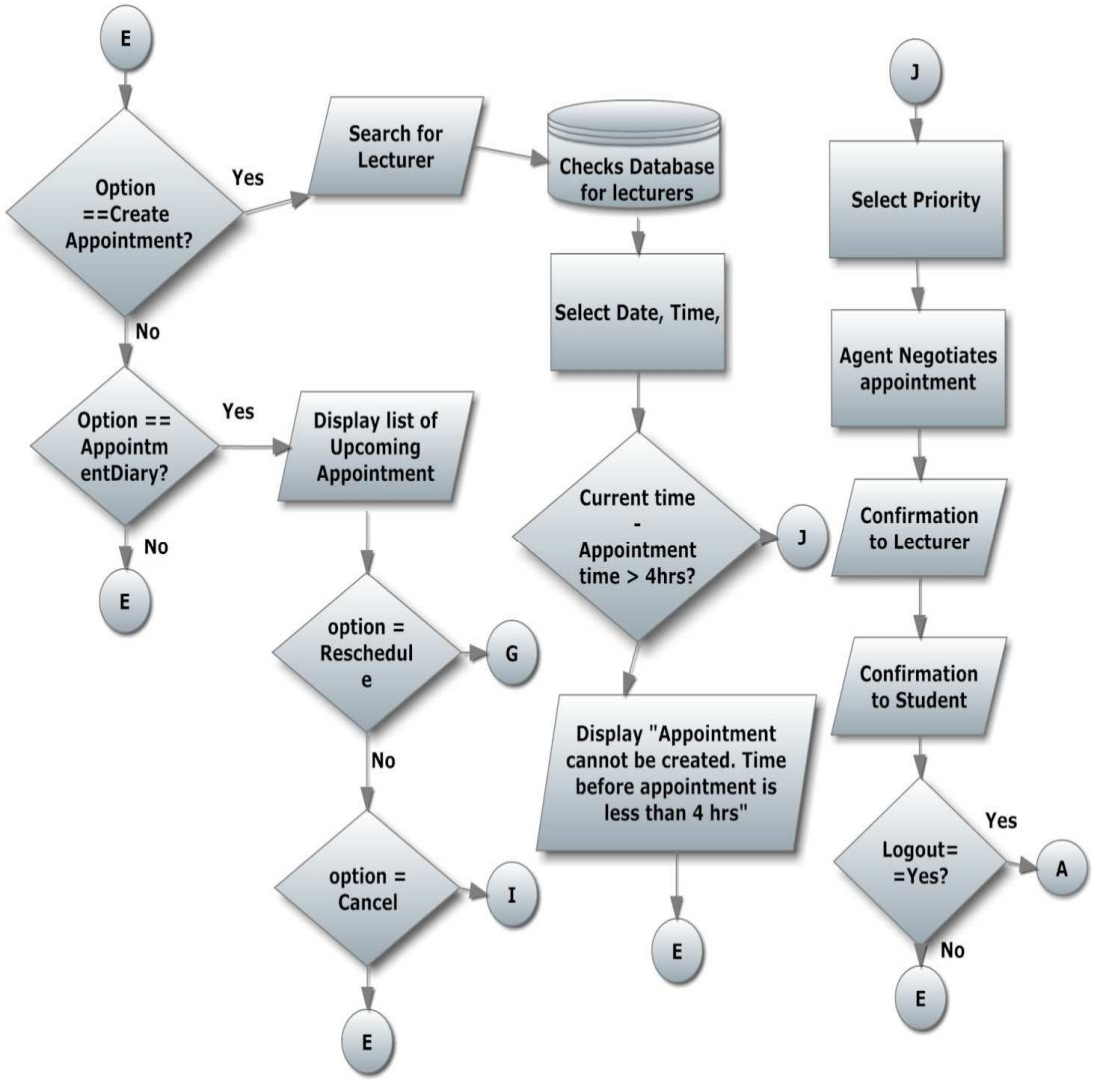

Figure.2(b). Flow Chart of Agent based Appointment System (Continuation)

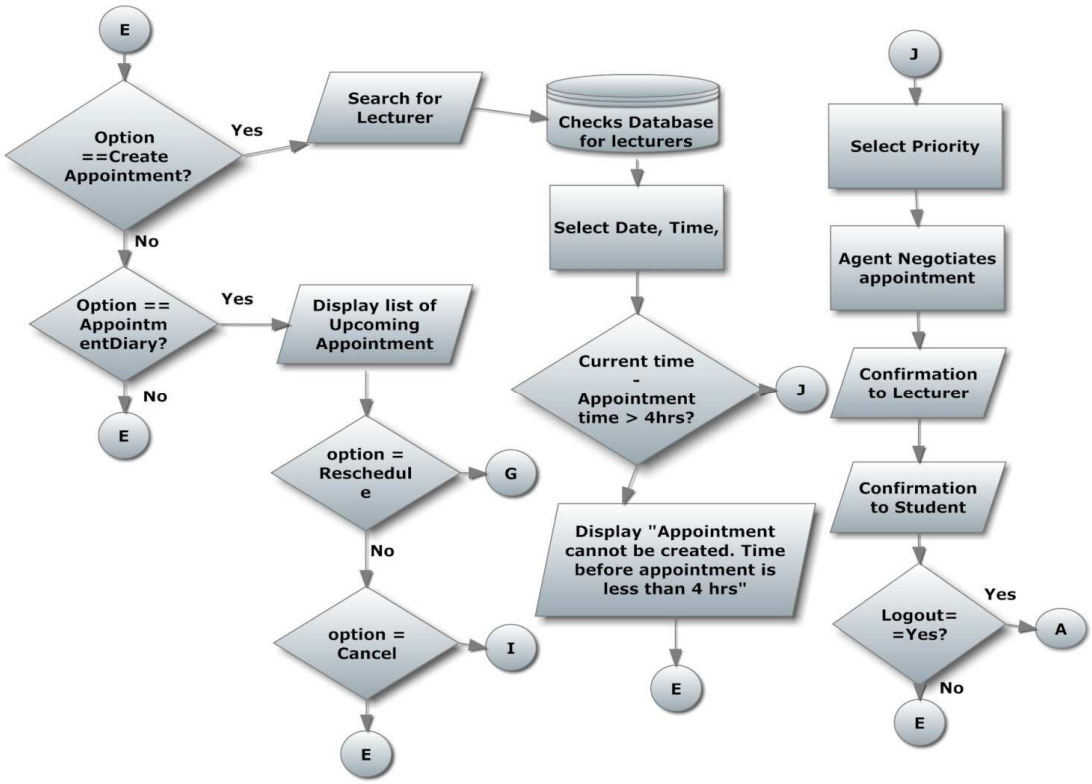

Figure.2(c) Flow Chart of Agent based Appointment System (Continuation)

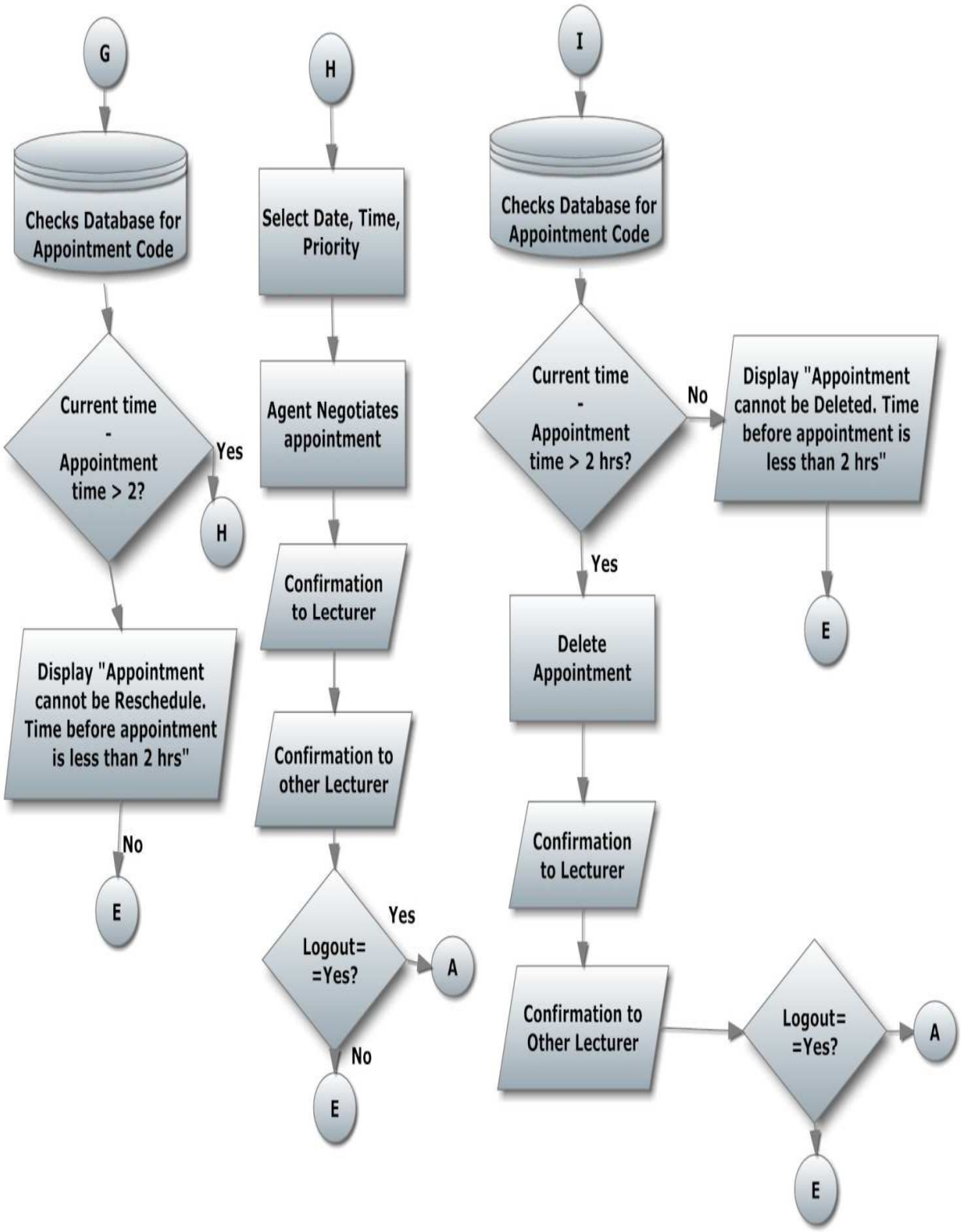

Figure.2 (d) Flow Chart of Agent based Appointment System (Continuation)

The Java programming language is used to create applications for the Android environment. Android relies on Linux version 2.6 for core system services such as security, memory management, process management, network stack, and driver model. For software development, Android provides Android SDK (Software development kit).

JADE (Java Agent DEvelopment Framework) is a software Framework fully implemented in Java language. It simplifies the implementation of multi-agent systems through a middle-ware that complies with the FIPA specifications and through a set of graphical tools that supports the debugging and deployment phases. The agent platform can be distributed across machines (which not even need to share the same OS) and the configuration can be controlled via a remote GUI. The configuration can be even changed at run-time by moving agents from one machine to another one, as and when required. JADE is completely implemented in Java language and the minimal system requirement is the version 1.4 of JAVA (the run time environment or the JDK). An add-on to JADE creating a modified environment called JADE-LEAP[1] allows the implementation of agents in mobile devices with limited resources. The synergy between the JADE platform and the LEAP libraries allows obtaining a FIPA-compliant agent platform with reduced footprint and compatibility with mobile Java environments down to J2ME-CLDC MIDP 1.0. The LEAP libraries have been developed with the collaboration of the LEAP project. The Intelligent Agent based Scheduling of Student-Staff Appointment been implemented using JADE-LEAP on Android.

Fig. 3 shows Agent environment on how the agents will interact with the system. Agents will be registered within the system using their telephone numbers to identify them. The successful scheduling of appointments was done using preferences strategy. Each appointment is referenced by an appointment code within the Microsoft SQL Server Database. The appointments priorities were assigned numerical values where the lower the value, the greater you priority will be and given preference. Priorities start at the value 1. The priorities were grouped into three categories specifically High, Medium and Low as shown in Table 1 and 2.

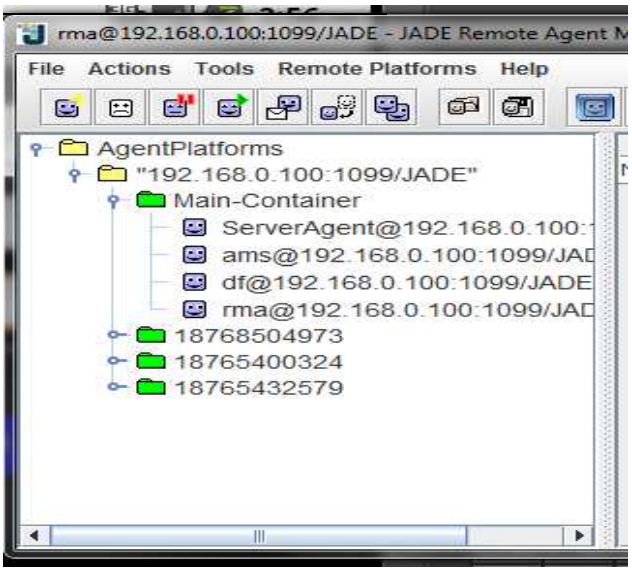

Figure. 3 Agents in JADE-LEAP

Database Code	Priority
	High Priority Class
1	Students failing the course and forced to submit their thesis by certain deadline
2	Student – final exam/Assignment/midterm query just one week to 3 days before the deadline
	Medium Priority class:
3	Reference Submission/Scholarship Application
4	Research Thesis/project discussion
5	Research thesis/project writing
	Lowest Priority class
6	Course work students- lectures/tutorials/labs
7	Others

Table 1. Student Preferences and Code

Database Code	Priority
	High Priority Class
1	Student related matters - Performance of course work/thesis student
2	Timetable issue
	Medium Priority class:
3	Examination paper discussion
4	Research Grants/Funded project discussion
5	Course Delivery Strategies
	Lowest Priority class
6	Administrative matters
7	Summer/Intern, Scholarship
8	Other Matters

Table 2. Staff Preferences and Code

The Teaching periods of staff were taken into consideration and hence were added to the database with a database code of zero (0). This indicates the lowest possible code and hence the greatest priority and therefore could not be replaced with no other schedules. The validation of our system will be discussed with various scenarios showing the intelligence developed in our system toward Appointment, Cancellation of Appointment, and Appointment Diary Viewing.

A. Appointments

Let us consider a scenario where three Students requesting the same date and time with but with different level of priorities. Before initiating the appointment, user will be presented with a screen to login as shown in Fig.4. Users are expected to enter their ID number, password and telephone number towards logging. Upon pressing the ***connect*** button, the agent will be created. The validation of the Username and Password will be also done when the ***login*** button is pressed. The information will be carried by the agent to the database and tested against the stored data.

Upon successful validation, the user will be taken to the main menu. It is important to note that the main menus are based upon the category of the user login. Fig. 5 shows a student main menu which is different from that of a staff. After the user logs in, the choice is given to create an appointment or to view already existed appointments. If the student or staffs wants to modify an appointment, whether to delete or re-schedule, then the ***Appointment Diary*** button is clicked.

Figure. 4 Login Screen for Student and Staff

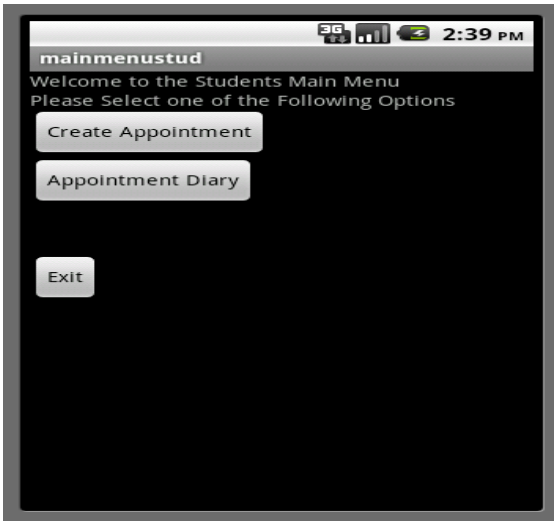

Figure .5 Student Main Screen

When the Create Appointment button is clicked, the student has the option of searching for the lecturer with whom appointment is to be made as shown in Fig.6 . The option is given where the student just has to enter only the first initial of the last and first name. After the data is entered, the continue button is clicked; here the student agent will take the data and pass it onto the database to check if staff members with those criteria exist. If no staff is found, then an error message will be displayed, otherwise the list of staff will be displayed and the student will now have the option of selecting the staff the he or she needs. These are shown in Figs 6 and 7. Figs 8 and 9 show, the student has selected the date and time. However, both date and times cannot be the same and hence an error message is displayed as shown in Fig 10 which is corrected and the student is taken to the next screen to select the reason for the appointment as shown in Fig 11. In the example shown the Student chooses the appointment preference which is Reference Submission/scholarship application which is highest priority in medium priority class as shown in Table-1. When the priority is selected and the continue button is selected, the student is now given the option of viewing the data that were selected. This is done through the summary user interface as shown in Fig 12. If corrections are to be made, then this is done by clicking the back button. If the continue button is clicked, the agent will take the data and pass it on to the Scheduler agent for negotiation. If the requested time is valid, meaning it is free and not within an already specified schedule, then it is given to the user and a confirmation message is sent to the staff and student phone as shown in Figs. 13 and 14, through the staff and student agent. The appointment information is recorded in the database.

Figure.6 Lecturer Search Screen

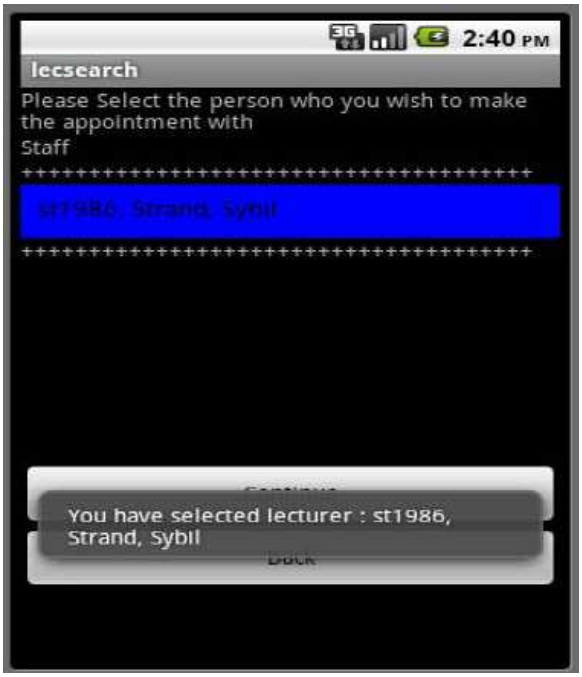

Figure.7 Lecturer Selected

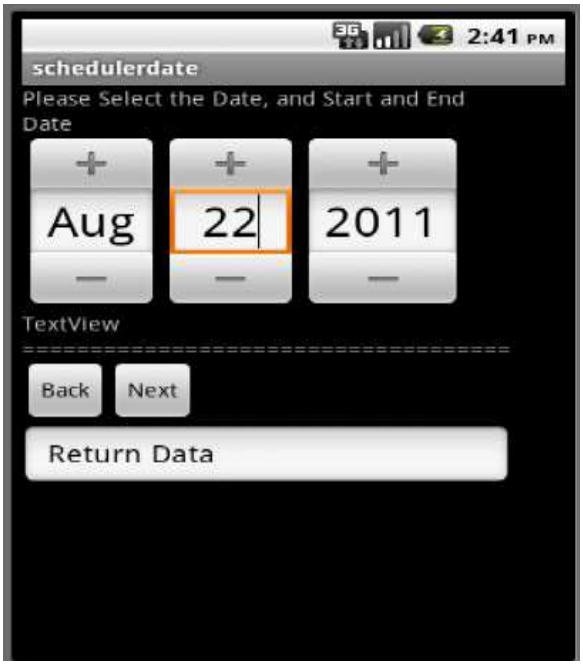

Figure.8 Date Select Screen

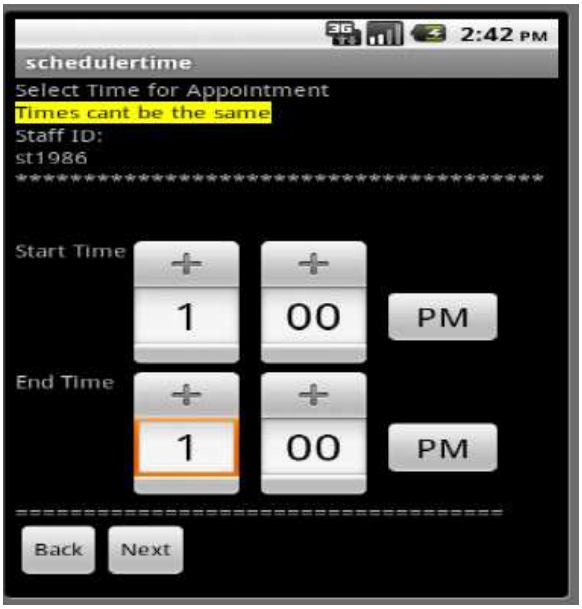

Figure.9 Time Select Screen

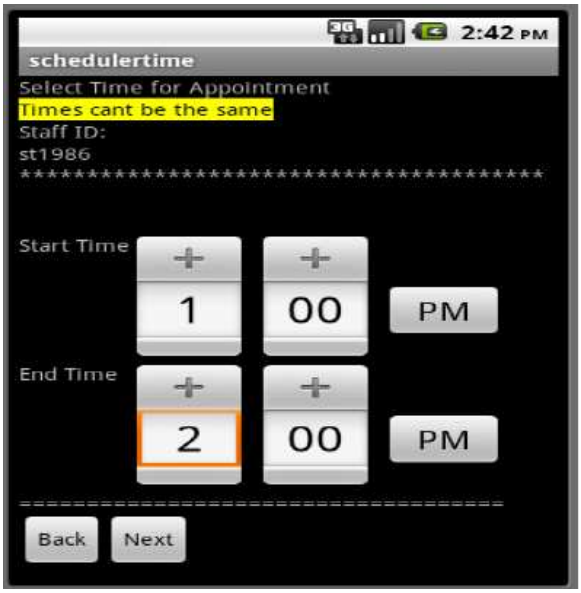

Figure.10 Start and End Time Select Screen

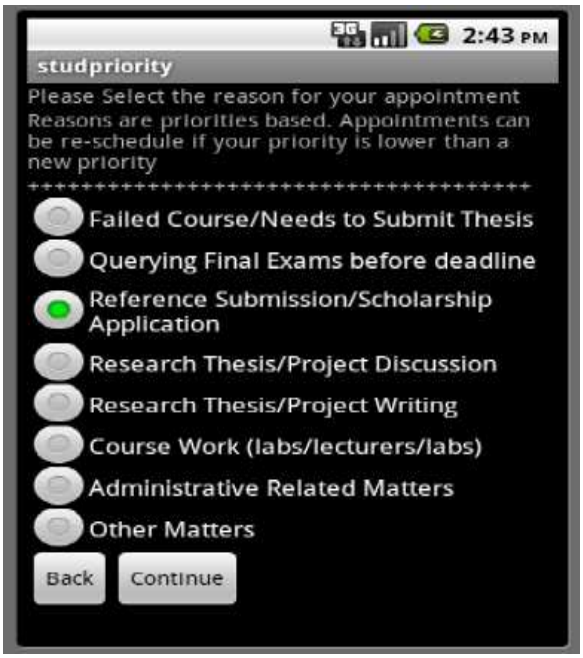

Figure.11 Appointment Preference Student-1

Figure.12 Appointment Data Summary

Figure.13 Confirmation on staff Phone

Figure.14 Confirmation on student-1 Phone

Now let us say student2 makes appointment for the same date and time say, 22nd August 2011 between 1:00- 2:00 PM with the same lecturer but with a higher priority reason. We will now see how the agent intelligence has been used in negotiating the appointment for student2 with the same lecturer. Similar to student 1, student2 also logs on with his credentials and looks for the appropriate lecturer. Student2 wanting to meet the same lecturer, with whom the student1 has already made an appointment towards failed course/ Needs to submit Thesis subject has the highest priority as shown in Table-1. The request summary is now presented to student2 as was done to student-1. Again as was done with student1, the option is given to student2 to make changes or continue to pass the information to the Scheduler agent Now based on the scheduling algorithm, the system is expected to reschedule the Student1 appointment and give the time slot to student2 as the appointment preference of student2 has higher priority compared to that of student1 which is shown in Fig 15. Fig.16 shows the student2 receiving the confirmation on mobile handset about the appointment.

Figure.15 Confirmation for Lecturer- student-1 & 2

Figure.16 Student-2 Confirmation

Now consider another Student3 makes appointment for the same date and time i.e. 22nd August 2011, 1:00- 2:00 PM with the same lecturer but with a different priority reason compared to the previous two students i.e. Research Thesis/Project writing which is lower priority compared to appointment of student1 and student2. Again student3 also logs on with his credentials similar to previous two students. Here the appointment selected has a lower appointment priority value in the database. The summary of the user selection are then displayed when the user clicks continue. The user i.e student3 can select the continue button to pass the information to the scheduler agent or back button to make changes to the selection. The agent realizes that the requested time i.e 1:00- 2:00 PM on 22nd August 2011 is already reserved to student2 with a higher priority followed by student1 with a medium priority for 2:00 to 3:00 PM. Now scheduler agent negotiates another time slot for student3 with the same lecturer taking the appointment preference into consideration. The result of the negotiation is displayed on the lecturer screen and then on the user screen i.e. student as shown in Figs. 17 and 18.

Figure.17 Lecturer Confirmation- Student -3

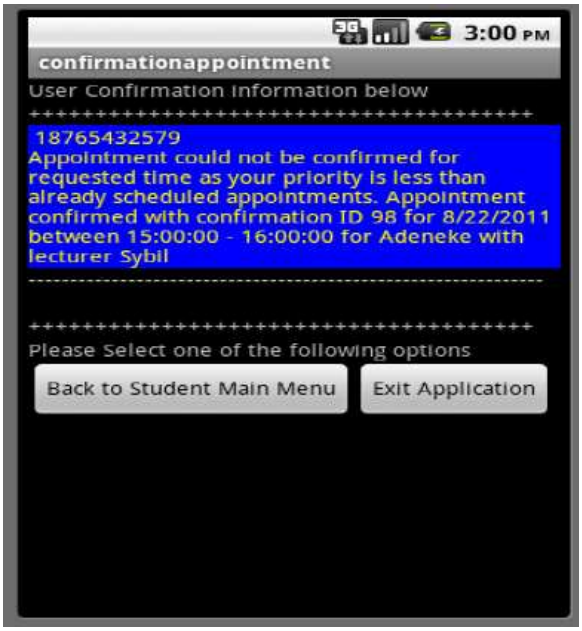

Figure.18 Student-3 Confirmation

Let us consider another scenario in which three lecturers log in and request the same date and time with different reasons for appointment. The log in screen is displayed which is similar to that of the student where the first lecturer logs in with his username, password and phone number for making an appointment as in case of student appointment. The information is passed to the agent and the username and password is queried against the stored one in the database and the main screen is displayed. It is important to note that the main screen is different from that of the student. Once the create appointment button is clicked, the user is taken to the screen to select the lecturer with whom the appointment is to be made with. The user enters the first initial of both the last and first name of the lecturer. When the continue button is clicked the agent again query the database for all the users that meet that criteria and return a list as shown in student appointment. A lecturer is selected and the continue button is clicked to move on to select the date and time. The date and time is selected similar to that of a student and also if an attempt was made to select a start time that is more than the end time then a validation error message would be displayed. Here the user can't move on until this is corrected. The next button is clicked and the reason for the appointment is selected i.e. Research Grants/Funded Project Discussion as shown in Fig. 19. Here there is a different screen than that of the student.

The summary screen of the user selection is displayed as shown in student appointment and the Lecturer as in the case of the student, can decide to go back to make changes or click continue to pass the data on to the agent to negotiate appointment slot in the database. The Scheduler agent finds that the slot is free and hence it is given to the lecturer who is making the appointment. This is displayed first on the lecturer's phone to which the appointment is being made as shown in Fig. 20 with and then on to the lecturer's phone who is making the appointment as shown in Fig .21.

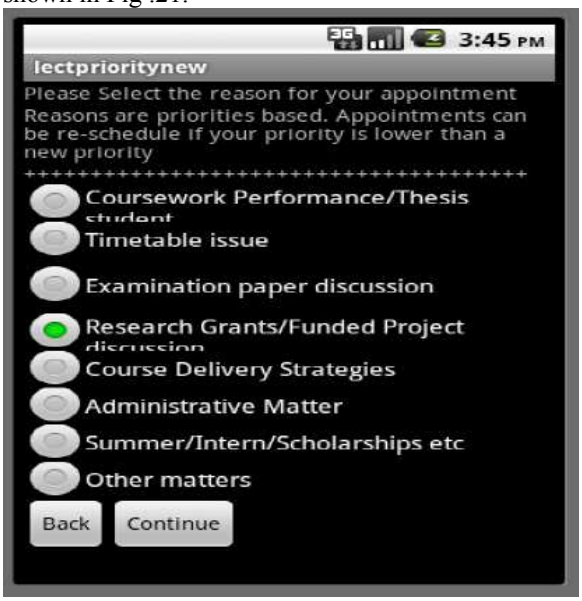

Figure.19 Lecturer-1 Appointment Reason

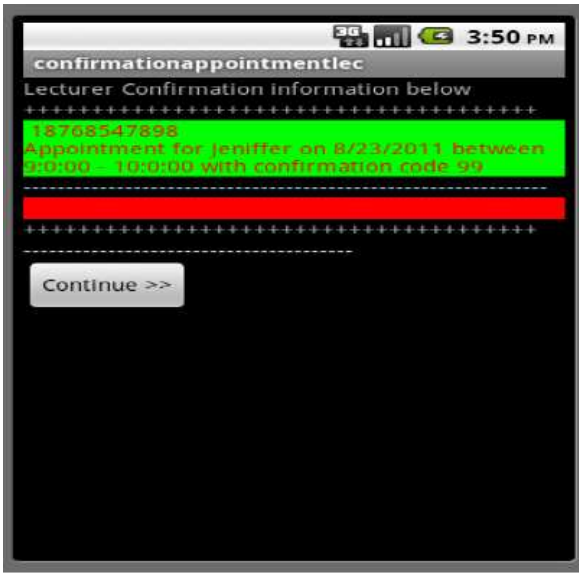

Figure.20 Confirmation Screen of Lecturer

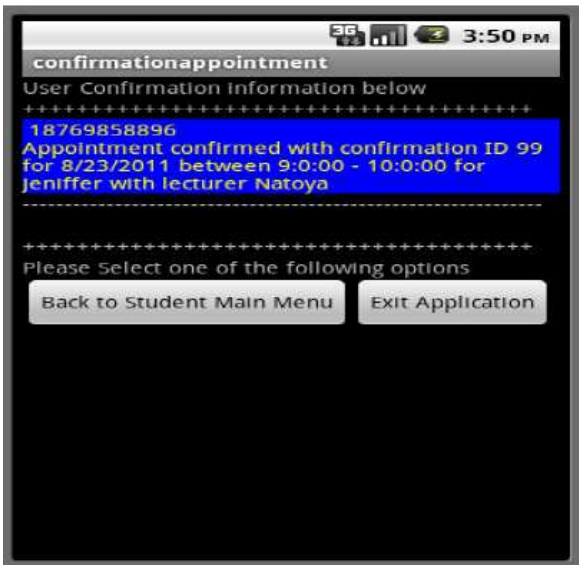

Figure.21 Confirmation Screen of Lecturer-1

Let us consider Lecturer2 who signs in and makes request for Appointment for same date and time but with a different reason. The data is validated against that stored in the database and the user is taken to the same main screen as before. The create appointment button is clicked and the user selects the same lecturer as lecturer1. The same date and time is selected similar to lecturer1 i.e. 23rd August, 9:00-10:00 AM. The next button is clicked and the user is taken to the appointment reason screen similar to one shown in Fig.19 where the Lecturer2 selects the appointment reason as Administrative matters, which is of lower priority compared to that of Lecturer1 appointment reason. Here the scheduler agent could not give the requested time for Lecturer2 as the requested time has already been scheduled for Lecturer1 towards Research project discussion which is higher priority. So according to Scheduling algorithm Lecturer2 appointment reason is administrative reason which is lowest priority and so would be granted the next time slot only which is 10:00 - 11:00 AM instead of 9:00-10:00 AM and rescheduling of Lecturer1 time is not possible. This is confirmed first on the lecturer's phone to which appointment detail is being made with and then on lecturer2's phone who is making the appointment as shown in Fig. 22 and 23.

Let us now consider the scenario where Lecturer3 signs in and makes request for Appointment for same date and time but with a different reason similar to previous two lecturers. As customary, lecturer3 has to sign in and an agent created for access to be given to the system similar to previous two scenarios. Also lecturer3 searches for the same lecturer which

lecturer1 and lecturer2 made appointment with. Lecturer3 is taken to the screen to select the reason for the appointment which is Course work performance/Thesis student but this appointment would take priority over the previous two appointments as it is the highest priority. Once the continue button is clicked the summary screen is displayed so that the selected data can be viewed. Once the continue button is clicked, the Scheduler agent takes the data to the database to be negotiated. The fact that this priority is higher means that rescheduling would take place with Lecturer1 and Lecturer2 towards their appointment i.e. Lecturer3 would take 9:00-10:00 AM slot followed by Lecturer1 who takes 10:00-11:00 AM and finally Lecturer2 which takes 11:00-12:00 PM slot. The result of the negotiation is displayed on the requester's phone as shown in Fig 24 and 25.

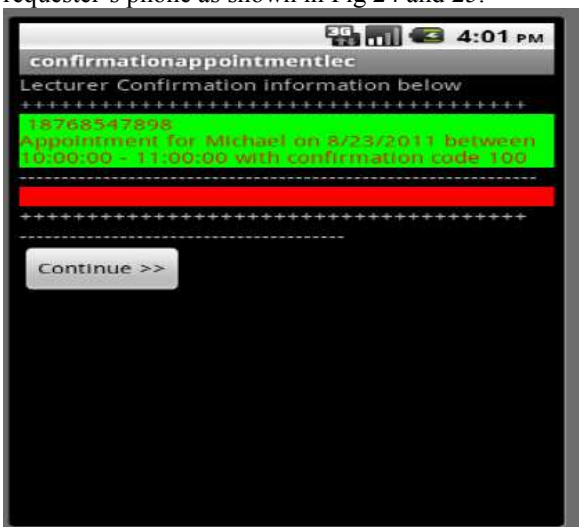

Figure.23 Confirmation Screen of Lecturer

Figure.24 Confirmation Screen of Lecturer

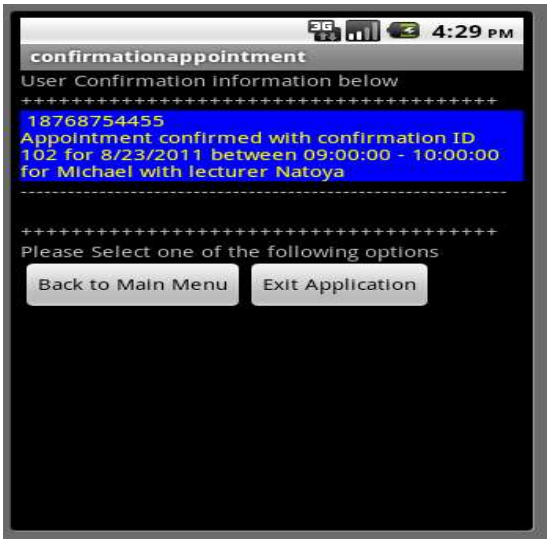

Figure.25 Confirmation Screen of Lecturer-3

B. Cancellation of Appointments

Till now we have seen on how appointments are made by students and fellow lecturers with lecturers taking into account the appointment preferences. We will now see as how appointments are cancelled.

The user i.e. Lecturer then logs into the system with his id st654 and the agent is created. The main screen is displayed to the user to which the appoint diary button is selected. The appointment diary option allows the user to see both the current and future appointments. The user has the option when these are shown in Fig.26 to either cancel appointments or re-schedule appointments. It must be noted however that there is a prescribe time period that allows for these actions to take place i.e. cancellation has to be done 2 hrs before the appointment time. In the list of appointments, we see two appointments being displayed in the list i.e. 72 and 75 as shown in Fig.26. From the list of appointments, one was selected and the next button is clicked. The summary screen is displayed indicating the appointment details to which the transaction will be done. The two possible transactions are Delete Appointment and Re-Schedule Appointment. Once the Delete Appointment is clicked, information is passed to Scheduler agent who checks to see that the present time is less than 2 hours before the appointment time, and hence it cannot be cancelled. So an error message is displayed as this action is not valid as shown in Fig. 27.The continue button is clicked and the user is taken back to the main menu.

Let us consider another scenario of appointment deletion where the same user with id st654 logs in to

the system. Once the user clicks appointment diary button, the list of appointments are displayed again. In this case only one appointment is displayed based on the time which is appointment number 78..This one appointment is selected for deletion Once the delete Appt button is clicked, the appointment information is passed to scheduler agent for deletion i.e appointment code 78 for 28th August 2011 between 4:00- 5:00 p.m. The scheduler agent checks to see if the present time i.e 1:31 p.m is less than 2 hours before the appointment time which is so in this case and the appointment is deleted from the database. This is confirmed on the lecturer's phone to whom the appointment was made with and then on the requester's phone. These two screens below are shown in Figs 28 and 29.

Figure.26 List of Appointments

C. Rescheduling of Appointments.

Let us consider a scenario as how appointments can be rescheduled. In here the constraint is that rescheduled can be carried out at a minimum time period of 4 hours before the scheduled appointment time. The user here logs on to the system with the user name and password which is validated as seen in previous scenarios. Once validated, the main screen is displayed which shows the appointment diary option to see list of appointments as shown in Fig 30.

From the list of appointment only two appointments are valid and displayed. One of them is selected say appointment code 87 and the rescheduled button is clicked and the rescheduled summary screen is displayed as shown in 31 and 32.

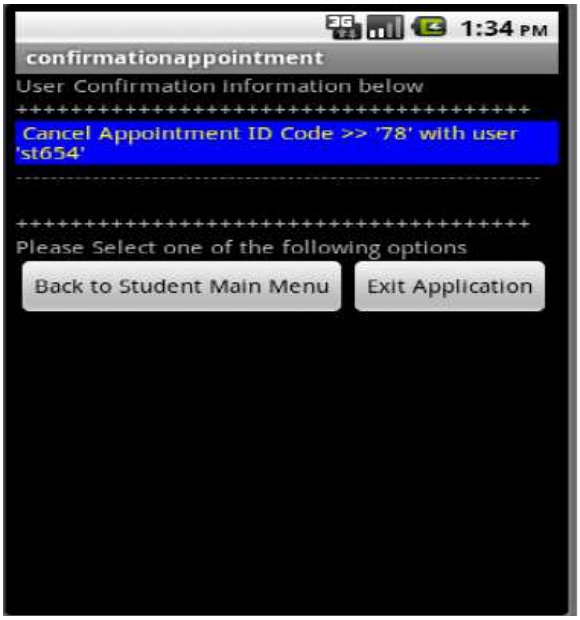

Figure.27 Appointment Deletion

Figure.28 User Confirmation of Cancellation

The continue button is clicked and the user is taken to another screen to validate if this transaction is possible. A read only screen is displayed and the user has to click validate as shown in Fig.33. On pressing validate the reschedule appointment information is passed to scheduler agent to check the schedule start time of the appointment and the current time.

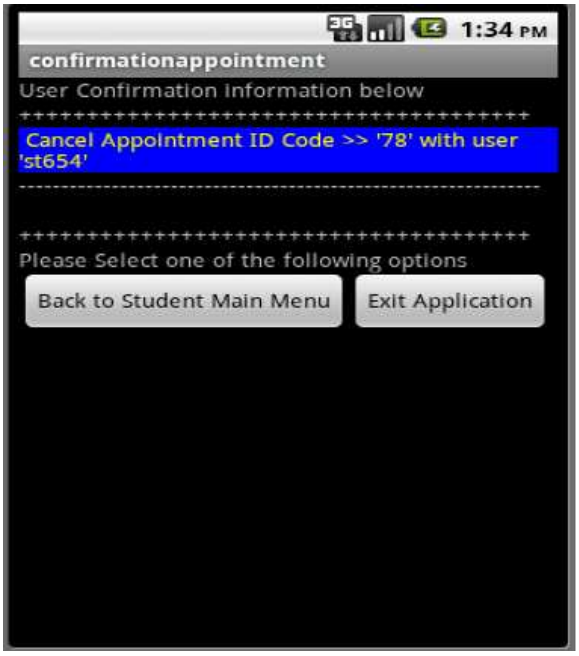

Figure.29 Lecturer Confirmation of Cancellation

Here the reschedule is not possible as the selected appointment time period is just 50 minutes before the scheduled start of the appointment and the system needs at least 4 hours before starting time to re-schedule appointment. User has to click back to return to the previous screens to select a different appointment. Note that the continue button is only enabled when the action is valid.

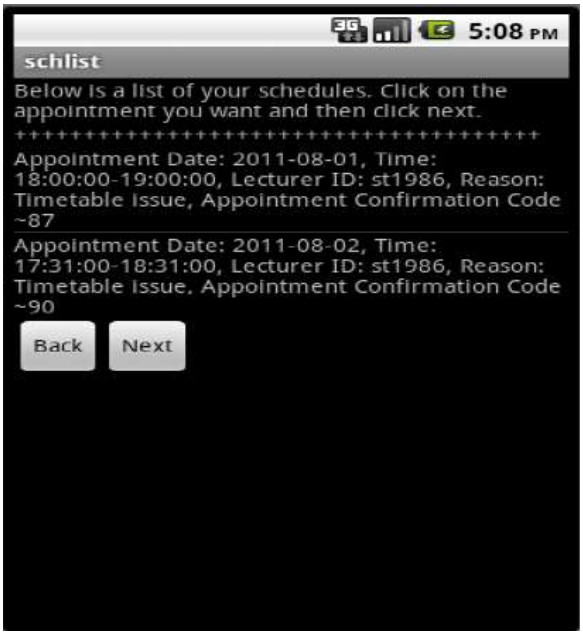

Figure.30 List of Appointments

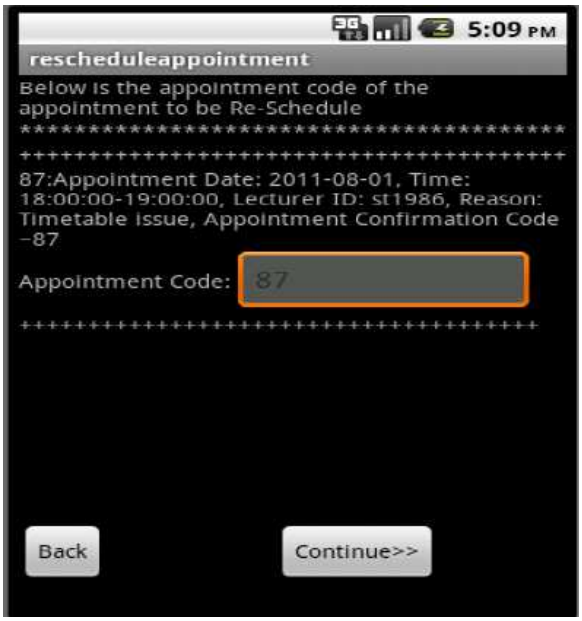

Figure.31 Appointment Selection

Fig.32 Appointment Validation

Let us consider another scenario towards reschedule of appointment which is valid. The same as before the user logs on to the system to view the list of appointments by clicking the appointment diary option as shown before. In here appointment code 90 is selected by user. The re-schedule button is clicked and the user is taken to the summary sheet .The user clicked *the continue* button and the validation screen

is displayed as before The validate button is clicked and the check is made by scheduler agent with the database to see if the action is possible based on the fuzzy rules as shown in Fig.34. The result of check came back and it noted that the action is possible. Here the continue button is now enabled and the user clicks it tocontinue with the transaction. The user now has the option to select the new date and time which must be at least 4 hours before the schedule start time. A new date and time is selected which is now a valid one which is more than 4 hrs before the requested start. The next button is clicked and the user is taken to the summary screen before passing data to the agent. The appointment summary is present as and the student/staff agent now takes the data to the scheduler agent to negotiate appointment by looking into appointment database. The result of the negotiation is presented on the lecturer's phone of which the appointment was already made with as shown in Fig 35 and then on the requester's phone as shown in Fig 36.

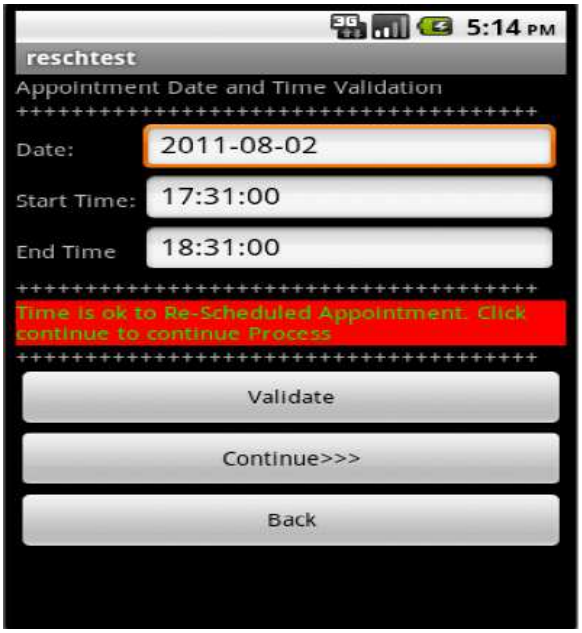

Figure.34 Validation Result

Figure.33 Validation Result

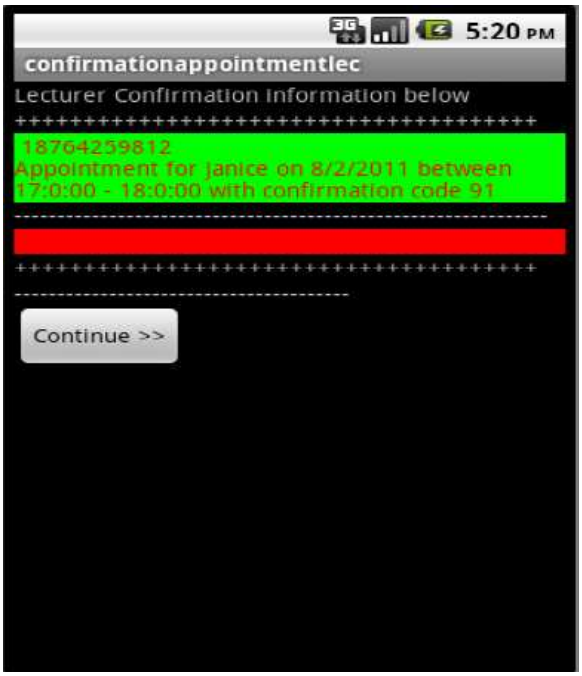

Figure.35 Lecturer Confirmation

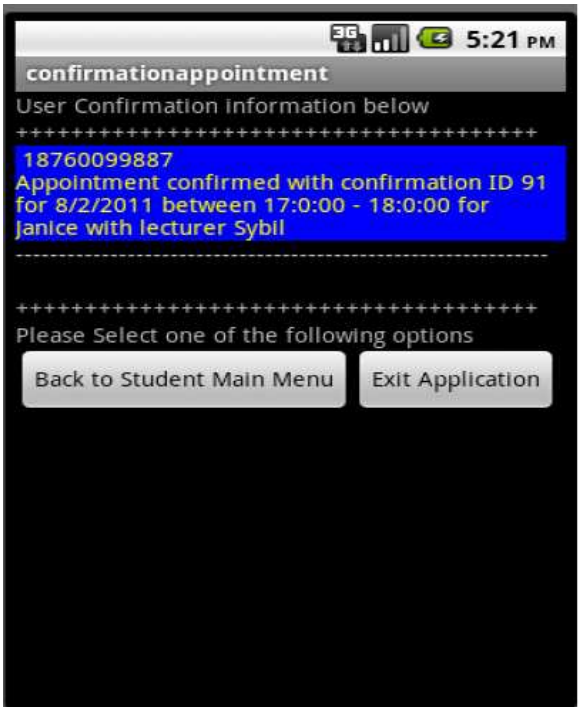

Figure.36 User Confirmation

VI. Conclusion & Future Work

The proposed system development presented clearly shows that the use of Intelligent Agent based systems is continually and rapidly making a foot-hold in the Information Communication Technology sector. Research has already been done in developing intelligent agent for appointment scheduling in Education to health on mobile handset which is evident from the literature. The research has enhanced the previous Agent based student appointment system i.e Negotiator 1.0 developed which has some drawbacks. The previous system had limited appointment preferences and also had no option of staff making appointment with the fellow staffs. Also the system stored the appointments of the staff on their personal mobile handset which is big drawback as loss of signal and so can lead to failure in appointment negotiation.

So we here have developed the system that allows not only students to make appointment but also staff. In addition the appointment negotiation takes places in central database and also all appointments of staffs stored in central database only. The appointment information are sent to both the staff and user requesting the appointment. All negotiation takes place at central database only. The system allows viewing the appointment diary by querying the database to delete, reschedule appointments taking into consideration the timing constraint. The System been developed using JADE-LEAP with Android add-on on Android mobile devices to make appointments with a lecturer by student and lecturer. The agents combined are both autonomous to operate with little or no human intervention but reliant on each other to coordinate the appointment making process with less manual labor involved and more cost effective. The proposed Intelligent Agent based Scheduling System of Student and Staff Appointment can be extended to allow Heads of Departments to be able to schedule meeting with staff at least 4 hrs prior and this meeting will take precedence above all other meetings. Automatic meeting reminder can be included to be sent to both staff and student through SMS and e-mail.

References

1. Education in Society. Retrieved from http://www.learningcity.org/society.html
2. Appointment scheduling , Retrieved from http://www.taskmanagementguide.com/solutions/articles/appointment-scheduling-software-for-office-staff-guidance.php
3. Appointment-Plus, Retrieved from http://www.appointment-plus.com/online_scheduling_software/mobile_device_online_scheduling_software.php
4. Wahab, M and Hassan, N. (2008) "A Web-Based Appointment System through GSM Network" Retrieved from http://eprints.uthm.edu.my/165/1/mohd_helmi_abd_wahab%2Cnorlida_hassan.pdf
5. Davidson, E. (2009). What Is a Computerized Vs. Manual Scheduling System? Retrieved August 16, 2011, from http://smallbusiness.chron.com/computerized-vs-manual-scheduling-system-11956.html
6. Kay, A. (1984). Computer Software. Scientific American 251(3); Pp.53–59
7. Selker, T. (1994), "A Teaching Agent that learns" Communications of the ACM 37 (7) pp 92-99
8. Smith, D C et al(1994) "Programming Agents without a programming language" Communications of the ACM 37 (7) pp 55-67.
9. Janca, P.C.,(1995), "Pragmatic Application of Information Agents: BIS Strategic Decisions.", Norwell, United States,
10. Jennings, N. R., and Wooldridge, M. (1998). Applications of Intelligent Agents. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.39.6599&rank=2
11. Etzioni, 0. And Weld, D (1995) "Intelligent Agents on the Internet: Fact, Fiction, and

Forecast", IEEE Expert, vol. 10 (4), pp 44-49..
12. Franklin, S. & Graesser, A(1996) "Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents", Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag
13. Bradshaw, J. (1997) "Software Agents". Retrieved from www.cs.umbc.edu/agents/introduction/01-Bradshaw.pdf
14. Reilly, D (1998), "Mobile Agents - Process migration and its implications", Retrieved from http://www.davidreilly.com/topics/software_agents/mobile_agents/index.htm
15. Schmeier, S and Schumpeter, A (1996)., "Personal Assistant for Scheduling Appointments", Proceedings of 1st International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, London, UK
16. Herrler, R and Christian, H (2002), "Appointment Scheduling among Agents: A Case Study in designing Suitable interaction Protocols" Retrieved from http://ki.informatik.uni-wuerzburg.de/~blandi/work/amcis2002.pdf
17. Koch, F., and Rahwan, I.(2004) "Interest-based Negotiation in Multi-Agent Systems
18. Busemann, S et al, (2007) "Natural Language Dialogue Service for Appointment Scheduling Agents". Proceedings of Fifth conference on Applied Natural Language Processing, Washington D.C, USA. pp.25-32
19. Vermeulen, et al,(2006), "Improving Patient Activity Schedules by Multi-agent Pareto Appointment Exchanging", Proceedings of the 8th IEEE Conference on E-Commerce Technology (CEC' 06), , 9-16
20. Anckar,B. et al (2001) "Overcoming Online Booking barriers with a Software Agent Approach", European Conference Paper, June 2001. Available at: http://is2.lse.ac.uk/asp/aspecis/20010005.pdf
21. Fanty M. et al, (1995), "Automated Appointment Scheduling", In ESCA Workshop on Spoken Dialogue Systems. Vigs
22. Suresh, S and Cuffe, K. (2010). Intelligent Agent based Scheduling of Student Appointment-Android Environment. Proceedings of 5th IEEE International conference on Computer Sciences, Convergence Information Technology, Seoul, Korea,Pp. 46-51
23. Hylton, A. and Suresh, S.(2011), Intelligent Agent based Scheduling of Hospital Appointment System in Android Environment. Accepted in 4th IEEE International Conference on Computer Science and Information Technology (ICCSIT 11), Chengdu, China
24. Parchment, D (2011). Agent Application in Student Staff Scheduling System, M.sc Computer Science Dissertation, University of West Indies, Jamaica
25. Android, retrieved from http://www.engineersgarage.com/articles/what-is-android-introduction#introduction

Author Biographies

Dwayne Parchment started his professional studies at the University of Technology, Jamaica in 2001 where he obtained a Bachelor of Science Degree in Computing and Information Technology. After completing his studies, he became a part-time teacher for computer science at St. Hugh's and then a Senior Teacher in 2006 where he filled the capacity of Consultant for Technical System Services till today. Further to that he went on to The MICO University College where he pursued a Post Diploma in Computer Science in 2006. Recognizing the need to be as marketable as possible, Mr. Parchment did a diploma in Video Production at CPTC and a diploma in System Administration at HEART. He is now completing his Masters in Computer Science at the University of the West Indies, Mona where his research focuses on using agent technology in scheduling appointment system. He is presently attached to the Computer Science and Information Technology Department at St. Hugh's High where he is the Consultant in Technical System Services and also teaches the subjects Information Technology (CAPE & CSEC) and Computer Science (CAPE), Word-Processing, and Electronic Document Preparation and Management. Mr. Parchment also manages the school's database, web site and computer network and is the staff advisor to the Inter-School Christian Fellowship (ISCF). Mr. Parchment is also an adjunct lecturer at The MICO University College where he delivers the courses Advanced Computer Networking, Advanced Database Management, Fundamentals of Computer Technology and Comparative Programming language; As service to his country in developing the ICT needs of students, Mr. Parchment serves as a member of the Subject Advisory Group for Information Technology in the Elearning Project since December 2010.

Dr. Suresh Sankaranarayanan holds a PhD degree (2006) in Electrical Engineering with specialization in Networking from the University of South Australia. Later he has worked as a Postdoctoral Research Fellow and then as a Lecturer in the University of Technology, Sydney and at the University of Sydney, respectively during 2006-08. He is the recipient of University of South Australia President Scholarship, towards pursuing the PhD degree programme and has also bagged the IEEE travel award in 2005. He is also Senior Member of IEEE computer Society and Computer Society of India too. He was working as a Lecturer (Asst. Prof. Status) in the Department of Computing and lead the Intelligent Networking Research Group, in the University of West Indies, Kingston, Jamaica, during 2008-11.He has also worked as a Professor, School of Computer Science and Engineering, Vellore Institute of Technology (VIT University), Chennai Campus, India, for a short period during 2011. He is now working as Associate Professor, Department of Computer & Information Systems, Institute of Technology, Brunei (ITB – A technological university). Currently he is also functioning as a Visiting Professor, Department of computing, Faculty of Pure & applied Science**,** University of West Indies, Mona Campus**,** Kingston-7, Jamaica, West Indies. He has supervised 28 research students leading to M.Sc, ME, M.Phil and M.S degrees and currently supervising 6 students leading to M.Sc, M.Phil and Ph.D degrees respectively. He has got to his credit, as on date, about 50 fully refereed research papers published in the Proceedings of major IEEE international conferences, as Book Chapters and in International Journals. He is also a Reviewer and Technical Committee member for a number of IEEE Conferences and Journals. He has conducted many tutorials, workshops and also given Guest Lectures in Networking in various Universities and Colleges. He also managed a collaborative research programme with Oakland University, Rochester, USA. His current research interests are mainly towards Mobile and Ubiquitous Computing- Wireless Sensor Networks in Health & Engineering , Intelligent Agents, Cloud Computing, Mobile commerce.