Analysis

Existing System

The company is a *dental clinic* called "**Tay Dental Surgery**", which serves patients in the region in the Bukit Timah region of Singapore. The user of the system is the dentist's assistant. The user currently uses a paper diary to keep track of appointment dates; this can be seen in Figure 1 below.

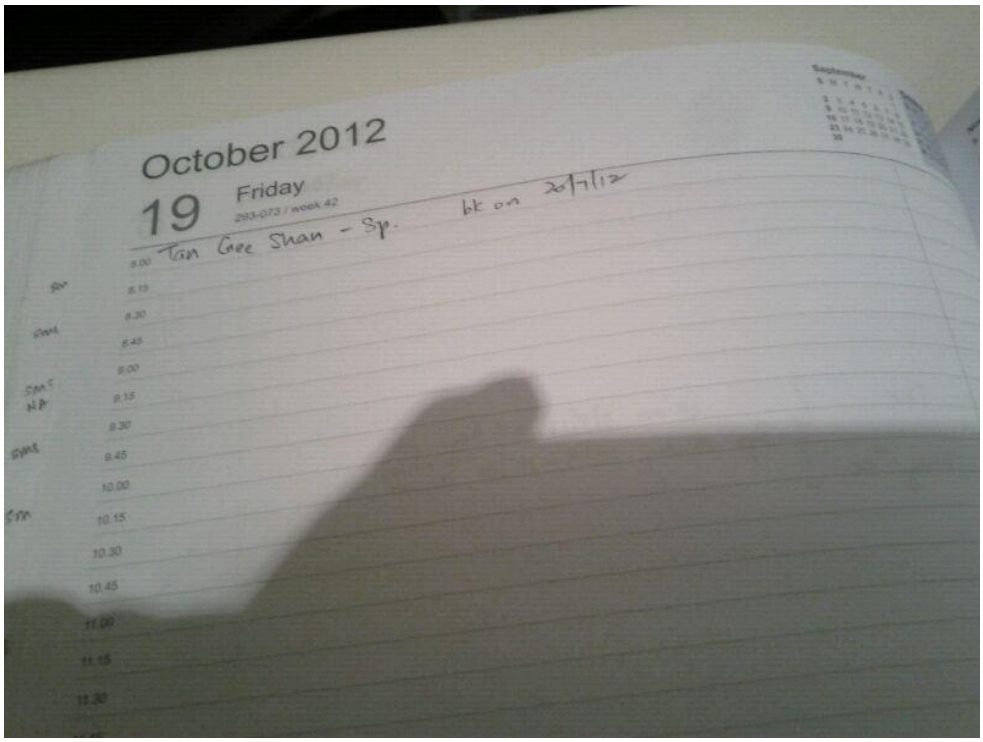

Figure 1 - A picture of the current medium of tracking appointments

To make a new appointment, the patient will call the clinic or will physically go to the clinic and the patient will verbally tell the user the appointment date and the user will write it down in the paper diary. This is to keep track of which patients have an appointment on any given day, so as to not forget about appointments and not to book 2 patients in the same time slot.

If patients want to change or cancel their appointment, they will have to call the clinic. They will then have to remember the date which they previously booked and the user will turn the page of the diary to that day and manually erase the appointment from the book. The new appointment date, if any, will then be added to the diary.

The user currently stores details of existing patients on record cards in a file cabinet as seen below in Figure 2. The file cabinet contains cards which have patient's details on them, and are stored organised according to their name. Before each appointment, the patient's card will be manually searched for and taken out and the appointment date will be written on them. There are a total of 12 of these drawers stored in a cupboard.

Figure 2 - A picture of the organisation of the file cabinet

New patients will have to provide relevant details on a separate card after which the user will manually write details on record cards, as seen below in Figure 3, and place them in the file cabinet. Past appointment dates are also recorded onto the card in order to keep track of when the next regular dental check-up appointment has to be made.

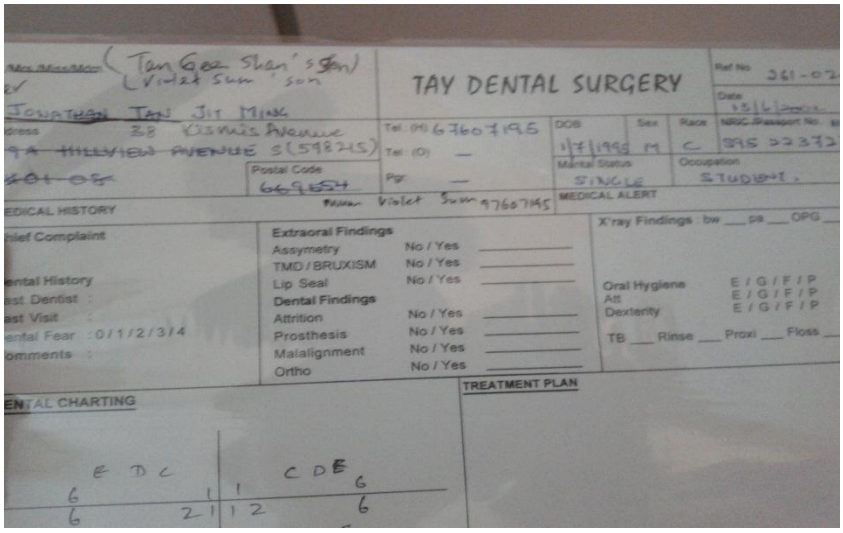

Figure 3 - A picture of the record card which stores the patient's details

Problems

During an interview with the user, the user had highlighted several problems with the current system of details storage and appointment management:

Legibility:

The user makes sure that the details of the appointment are recorded legibly, as they are to be referred to at a later date. With more patients visiting the clinic, the user has trouble writing the details of the appointment legibly. This is a problem because the user will not be sure of the details of the appointment and has to spend additional time to attempt to find out the correct details of the appointment.

Time Consuming:

The user also has to search in a file cabinet for the patient's record. As seen in the diagram above, the file cabinet has many records and looking for an individual record can be time consuming. Moreover, if the user makes a mistake, it would take even longer in order to find the correct record and put back the incorrect record.

The process of replacing an empty slot for an appointment with another is time consuming as the user will look through all recorded appointments, manually looking through the details each appointment, and finding out of the patient has a preferred time slot which had just turned empty.

Expensive storage of patient details:

With more new patients visiting the clinic, more record cards are needed. There is also a need for more file cabinets as the current file cabinets get full. As a result, the user will have to buy a new file cabinet and re-organize the current system in order for the new file cabinet to be of use. This process costs the clinic money and is very time consuming.

Criteria for success

Objective	Description	Evidence
A	Handwriting in the paper diary is illegible	1. View appointments for today User input:1 John Appleseed – Wednesday 10/10/12 4:00pm – 6:00pm
B	Searching for patients' details in file cabinet is time consuming	5. Search for an existing patient User input: 3 Enter name: User input: John Appleseed John Appleseed ********** Address: Date of birth: Allergies:
C	Replacing empty slots with patient's preferred time is time consuming	Are you sure you want to delete this appointment? User input : Y Appointment deleted Would you like to replace this time slot with another appointment? User input : Y New appointment details: John Appleseed – Wednesday 10/10/12 3:00pm – 5:00pm **or** No patients prefer this time
D	Storage of patient's records is expensive	4. Add a new patient User inputs patient's details Patient added

Prototype Solution

The main menu of the program will look like this:

```
TAY DENTAL SURGERY DATABASE
***************************

Please select what you want to do:
1.      View appointments for today
2.      Make a new appointment
3.      Search for an existing appointment
4.      Add a new patient
5.      Search for an existing patient
Enter command :
```

At this point, the user will enter the number which corresponds to the action they want to do. If the user enters command “1”, all appointments for the day would appear, in order of time.

After the operations required for the particular option has been completed, the main menu will be displayed again, allowing the user to select other options.

If the user wants to make a new appointment, they enter the command “2”. A menu, as seen below, will then appear prompting for user input of the details of the appointment.

```
Enter command : 2

Please enter name of patient:
USER INPUT
Please enter date of appointment
USER INPUT
Is it a major surgical operation?
USER INPUT
Please enter appointment start time
USER INPUT
Appointment added
```

Each option will only appear after the user has input the details. This makes it less confusing for the user and makes it a more easy-to-use interface.

If the user wants to search for an appointment enters the command “3”, a menu pops up, which allows the user to search for an existing appointment.

```
Please enter name of patient:
USER INPUT
APPOINTMENT DETAILS
Change or Cancel or Do Nothing?
USER INPUT
Would you like to replace the empty slot with another patient?
USER INPUT
Time has been filled with APPOINTMENT DETAILS
```

The user will then decide if they want to change, cancel or do nothing to the appointment found. The user will then have a choice to fill the empty slot with another patient, which the program will automatically do for them.

If the user wants to add a new patient, they will press “4”, which will bring a menu where the user will enter the patient’s details.

```
Enter command : 4

Patient name:
USER INPUT
Address:
USER INPUT
Allergies:
USER INPUT
Patient Added
```

If the user wants to change a patient’s details, they would select option 5, which will bring up a menu like this:

```
Please enter name of patient:
USER INPUT

PATIENT DETAILS

Change or Do Nothing?
USER INPUT
Which details do you want to change?
USER INPUT
Patient details successfully saved
```

The user will input the field which they want to change, after which will then type in the updated details and will automatically get saved into the patient’s record.

User feedback on prototype

During an interview, the user was provided with multiple screenshots of the prototype solution. The user seemed to be quite open to the idea of a new system.

The user has been using the same system for over 3 years and is not trained to use a computer. The user said that they “would be willing to try the system” if they were given the proper training on how to use a computer as well as training of the operation of the program.

However, the user also commented that the computer system would not give them as much flexibility as the current system does. If the user wants to add an extra feature to the current system, the current system is flexible enough to do so. However, with a computer program, this will be hard as the program would need to be constantly improved upon.

Moreover, the user made it a point that the clinic does not currently have any computers and therefore the implementation of a new system would be costly. The interviewer then said that the computer required does not have to be very powerful as it is running a command-line interface. The user responded by saying that they think the computer would cost more than the extra money they would make in the long run by implementing the system.