***Introduction:***

Vaccines are given on a recommended schedule because the evidence shows that the proper doses at the proper times generate the optimal immune response. This requires several visits to a healthcare provider during childhood and later in life. If we miss one visit or get off the schedule, we may not be fully protected from certain diseases. Luckily, modern life has also brought with it modern technology. There are a few methods that we can use to keep track of our family’s immunization records with reminders on when the next scheduled vaccine is recommended to be given.

There are plenty of smartphone (iPhone, Android, BlackBerry) that are aimed at reminding us about events. All of them have calendars that can look very far into the future, so it’s not a bad idea to just make an appointment or reminder in our phone’s calendar for that date in the future when we need the next vaccine, even if it’s years in the future. Of course, there are dedicated applications (software programs) that will keep track of our immunization records and remind us when the time comes for the next doctor’s visit. One such app is Vaccine Reminder.

***Background:***

The immunization program in Bangladesh was inaugurated on April 7, 1979 (World Health Day).At that time, the service was limited to selected number of Upazila Health Complex, major hospitals and some NGOs. It was revealed through a 1985 survey that the fully vaccinated children constituted less than 2 percent of the child population. To achieve the Global Universal Child Immunization Initiative (UCI), Government of Bangladesh, began a phase-wise process of EPI intensification from 1985-1990. During this period, the EPI was expanded throughout 460 Upazilas (Sub-districts), 84 Municipalities and 4 City Corporations.

**Android** is a Linux-based operating system designed primarily for touch screen mobile devices such as smart phones and tablet computers. Android is open source and Google releases the code under the Apache License. This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Additionally, Android has a large community of developers writing applications ("apps") that extend the functionality of devices, written primarily in a customized version of the Java programming language.

The advent of smartphones has taken the potential for mHealth to a new level. So, we have taken an interest to develop an app on vaccine reminder.

User

Android smart phone

Interface

Notify the User through Reminder and Alarm

Entry for reminder

**Fig: Block diagram for a generalized reminder system**

***APP List:***

Apps have become ubiquitous in many aspects of our lives over the past five years, fueled by the widespread availability of tablet computers and smartphones. Tens of thousands of health, wellness, and medical apps are now available for download to Apple or Android devices from online stores. Some of them are:

* Pediatric on Call Vaccine Reminder
* Vaccine Schedule
* Vaccine Reminders Widget
* Vaccine Tracker
* IVaccine

***Assessment of app functionality:***

Although the number of healthcare apps is large and growing, there is significant variation in capabilities of the current apps, with most having only simple functionalities built into them. An analysis of healthcare apps available to consumers through the different app store resulted in categorization of apps based on whether they could:

* **Inform**: Provide information in a variety of formats (text, photo and video).
* **Instruct**: Provide instructions to the user.
* **Record**: Capture user entered data.
* **Display**: Graphically display user entered data/output user entered data.
* **Guide**: Provide guidance based on user entered information, and may further offer a diagnosis, or recommend a consultation with a physician/a course of treatment.
* **Remind/Alert**: Provide reminders to the user.
* **Communicate:** Provide communication with HCP/patients and/or provide links to social networks.

***Problem statement:***

Some apps cover the full patient journey such as self-diagnosis, finding a physician, filling prescription, compliance. We have developed an app on compliance stage where the app contains following features:

* Provide reminders
* Display information
* Record user entered data for notification

***System Requirement:***

* Software Requirements: Minimum API level 8
* Hardware Requirements: Android Smart Phone
* Supported Operating Systems: Android

***System Analysis:***

The Application requires no Internet as well to keep a track. User needs to press the welcome button of the app and then a list of vaccines is shown for children, adult and female. We have enlisted 18 child vaccines, 3 female vaccines and 5 adult vaccines. Details of the vaccine can be viewed by clicking each vaccine on the list. On the home screen there are three tabs. One is already discussed that is for vaccine list. Second one is for vaccine record entry. User needs to enter the person’s name who has to take vaccine, vaccine name, date against which a vaccine reminder notification will fire. User can also edit the entered record. Another tab is for app information.

***System Design:***

Systems design is the process of defining the architecture, components, modules, interfaces, and [data](http://en.wikipedia.org/wiki/Data) for a [system](http://en.wikipedia.org/wiki/System) to satisfy specified [requirements](http://en.wikipedia.org/wiki/Requirement). Here we have included four system design diagram:

* Use Case Diagram
* Data Flow Diagram
* Control Flow Diagram
* Class Diagram

***Use case diagram:***

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a [use case](http://en.wikipedia.org/wiki/Use_Case).

Children

Details about Vaccines

Women

Vaccine Info List

Adult

Save

C:\Users\Animesh\Desktop\user.PNG

Notification

New Vaccine Entry

**User**

Application Info

Vaccine Entry for Reminder

Edit and Notification

Edit Vaccine Entry

Delete

**Fig: Use Case Diagram for Vaccine Reminder**

***Data Flow Diagram:***

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system), modeling its processaspects.

Person Name

Notification in the Notification Bar

**Database**

Vaccine Name

Date

**Fig: Data flow diagram for Add new vaccine reminder.**

Reminder List

Vaccine Name

Matching

Onclick

Person Name

Get VacId

Edit or Delete

**Database**

Date

**Fig: Data flow diagram for Edit or Delete a vaccine reminder from Vaccine list.**

***Control Flow diagram:***

A control flow diagram (CFD) is a [diagram](http://en.wikipedia.org/wiki/Diagram) to describe the [control flow](http://en.wikipedia.org/wiki/Control_flow) of a [business process](http://en.wikipedia.org/wiki/Business_process), [process](http://en.wikipedia.org/wiki/Process_(engineering)) or program.

Vaccine Reminder List

Person Name

Date

Date

Person Name

Vaccine Name

Vaccine Name

**Database Update**

No

Has the due date arrived?

Continuous monitoring using PendingIntent

Generate Notification with Alarm

Yes

**Fig: Control Flow Diagram for Vaccine Reminder**

***Class Diagram:***

A class diagram in the [Unified Modeling Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language) (UML) is a type of static structure diagram that describes the structure of a system by showing the system's [classes](http://en.wikipedia.org/wiki/Class_(computer_science)), their attributes, operations (or methods), and the relationships among objects.

normalPanel

first

1. ~~TabActivity~~

2. Resources

3. Tabhost

4. TabSpec

5. Intent

1. getResources()

2. getTabHost()

3. newTabSpec()

4. setIndicator()

5. setContent()

6. setClass()

7. setDrawable()

8. addtab()

9. setContentView()

10. getMenuInFlater()

MainActivity

1. Activity Class

2. ImageButton

3. Intent

1. onCreate()

2. setContentView()

3. OnClickListener()

4. startActivity()

5. getMenuInFlater()

homeActivity

aboutActivity

aboutActivity

1. Activity Class

2. TextView

1. onCreate()

2. setContentView()

normalPanel

1. Activity Class

2. ListView

3. TextView

4. ArrayAdapter

5. Intent

1. OnItemClickListener()

2. onCreate()

3. setContentView()

4. findViewById()

5. setAdapter()

6. AdapterView()

7. startActivity()

8. putExtra()

ListActivity, ListActivity2, ListActivity3

1. Activity Class

2. TextView

3. Intent

1. getIntent()

2. onCreate()

3. setContentView()

4. findViewById()

5. getIntentExtra()

6. setText()

**Fig: Class Diagram (1) for Vaccine Reminder**

editVac

1. Activity Class

2. ImageButton

3. EditText

4. Calender

5. PendingIntent

6. Notification

7. HashMap<String,String>

8. AlarmManager

1. OnClickListener()

2. onCreate()

3. setContentView()

4. findViewById()

5. get()

6. setText()

7. DatePickerDialog()

8. OnDateSetListener()

9. getTimeInMills()

10. showDialog()

11. startActivity()

12. putExtra()

13. getSystemService()

14. getText().toString()

15. callHomeActivity()

newVac

1. Activity Class

2. ImageButton

3. EditText

4. Calender

5. PendingIntent

6. Notification

7. HashMap<String,String>

8. AlarmManager

9. Button

10. Dialog

11. Intent

1. OnClickListener()

2. onCreate()

3. setContentView()

4. findViewById()

5. get()

6. setText()

7. DatePickerDialog()

8. OnDateSetListener()

9. onCreateDialog()

10. getTimeInMills()

11. startActivity()

12. putExtra()

13. getSystemService()

14. getText().toString()

15. callHomeActivity()

16. showDialog()

17. getInstance()

DBController

1. SQLiteOpenHelper

2. SQLiteDatabase

3. PendingIntent

4. Notification

5. HashMap<String,String>

6. ContentValues

7. ArrayList

8. Cursor

1. onCreate()

2. execSQL()

3. onUpgrade()

4. get()

5. put()

6. insert()

7. close()

8. update()

9. getWritableDatabase()

10. Cursor.moveToFirst()

11. Cursor.moveToNext()

MyNotificationService

1. Service Class

2. IBinder

3. Uri

4. Intent

5. PendingIntent

6. NotificationManager

7. Notification

1. onBind()

2. onStartCommand()

3. getActivity()

4. SystemService()

5. build()

6. setContentTitle()

7. setTicker()

8. setWhen()

9. setSmallIcon()

10. setContentIntent()

11. setSound()

12. setDefaults()

13. notify()

homeActivity

1. ListActivity Class

2. ListView

3. TextView

4. ArrayList

5. Intent

6. ListAdapter

7. HashMap<String,String>

1. setOnItemClickListener()

2. onCreate()

3. setContentView()

4. findViewById()

5. setListAdapter()

6. getText().toString()

7. startActivity()

8. putExtra()

**Fig: Class Diagram (2) for Vaccine Reminder**

***Methodology:***

* In the beginning of the App, if one clicks the App icon, an image button will be on the screen.



Fig: The welcome Screen.

* If one clicks the image button, a Vaccine List will appear. It is the First tab. On the home screen there are three tabs. There have been enlisted 18 child vaccines, 3 female vaccines and 5 adult vaccines. Details of the vaccine can be viewed by clicking each vaccine on the list.

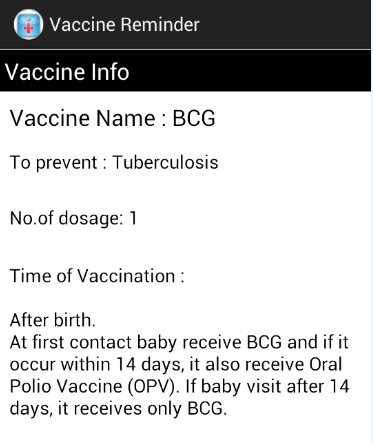
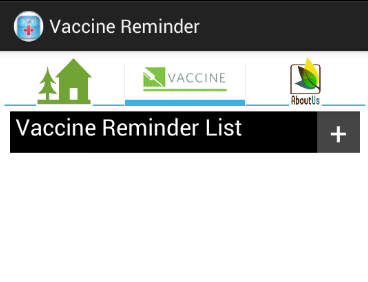
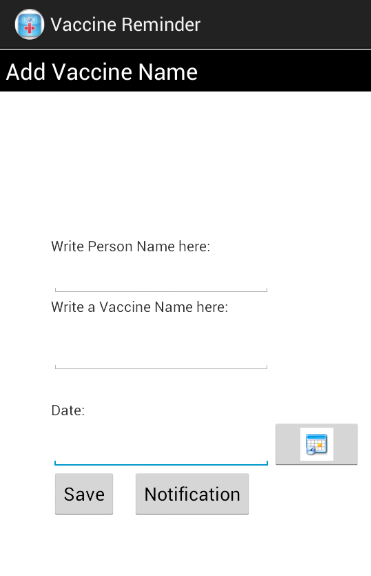
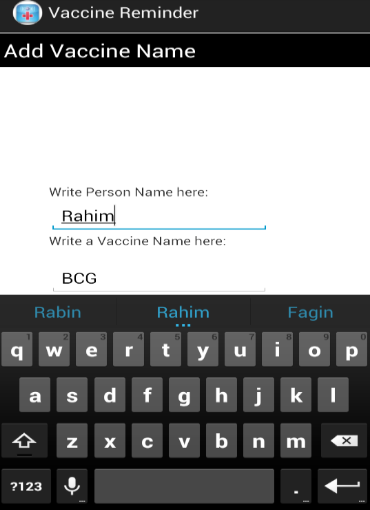
 

Fig: Vaccine Details

Fig: Home Screen Showing all the vaccine list

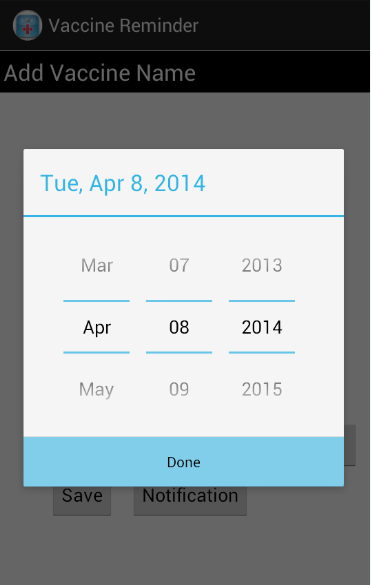
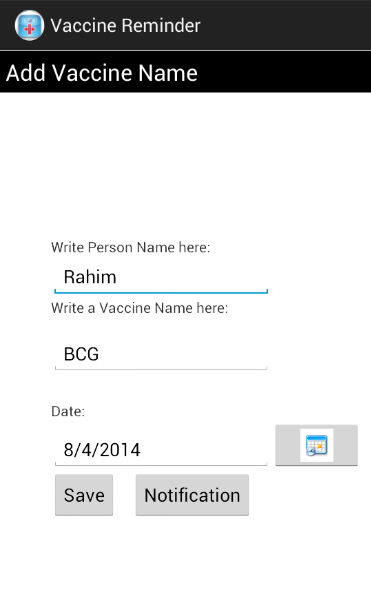
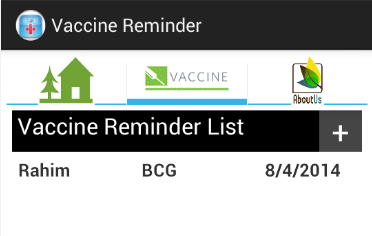
* The Second Tab is for vaccine reminder record entry. For add a new reminder user needs to click the (+) button. If do so, the user needs to enter the person’s name who has to take vaccine, vaccine name and the due date and click the “Save” button against which a vaccine reminder notification will fire at due date.

c) Entering Person name and Vaccine Name

b) The screen after clicking the (+) button

a) Before adding any reminder

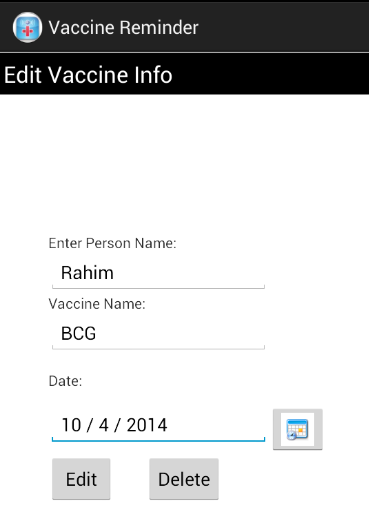
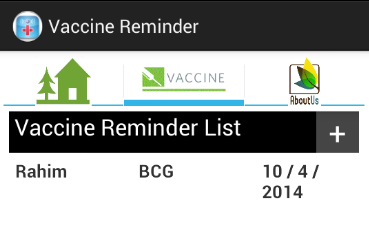
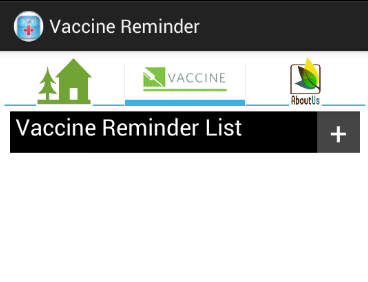
d) Picking the Due Date

f) After clicking the “Save” button

e) After entering all the Information

Fig: Adding a new Vaccine reminder.

* User can also edit the entered record by clicking on them and changing any of the portions (person’s name/vaccine name/date). By clicking the Edit button the vaccine reminder list will be updated. If one needs to delete the reminder, he/she can go the edit screen and by clicking the Delete button the vaccine reminder will be deleted.

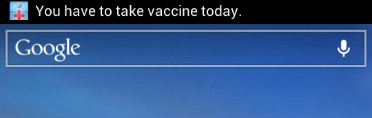
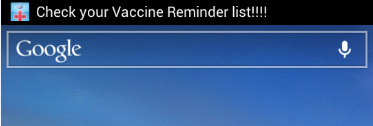
c) After clicking the “Delete” button

b) After clicking “Edit” Button

a) Changing the due date

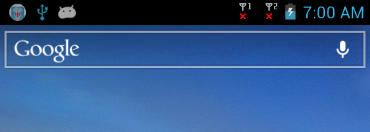
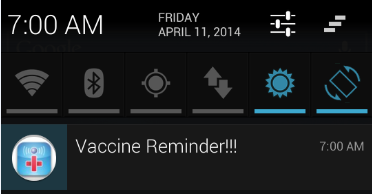
Fig: Editing or deleting an existing Vaccine reminder.

* At the due date, the notification will be fired at a specific time (7 A.M.). Until the notification is checked, there will be a reminder icon present on the Screen. When one checks the notification and clicks it, the home screen of the app will be appeared and he/she can check for which the notification has been fired.

b) Notification is firing.

a) Notification is firing.

e) After clicking the Notification on the notification bar.

d) Reminder Notification on the notification bar.

c) Reminder Icon is present until checking notification bar.

Fig: Notification generation for Vaccine reminder.

* The third tab is about app information. Here a small description and features of the app are given. Also the reference of the information of the vaccines is given here.

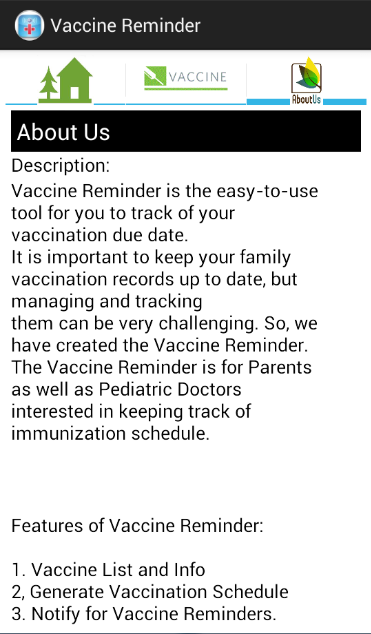


Fig: About Us Tab.

***Benefits:***

* The Application is not a server based one. It is an offline based Application.
* Separate tone has not to chosen for alarm. Whichever tone is selected for notification in the smart phone, this app will use the tone default.
* If vibration mode is on in the phone, the phone will be vibrated default for notification.
* There is no way that multiple user can use this one very app. If we use multiple user record, the phone memory will need more space which makes the app inconvenient to the user.

***Limitation:***

* Since this is not online based, to run the app, internal memory is used.
* To make the app user friendly the interface has to be designed in a more clarified way. But there is no exit button in every interface which makes the app clumsy for a new user.
* The notification will fire only in a fixed time.
* The apps must be explicitly started by the user for tracking.
* User authentication or any high security system is not used.

***Future Plan:***

* Our Future plan is to add features that will generate an automated vaccination and reminder schedule, customized to the users’ birth date to ensure timely vaccination.
* It will also keep track of vaccination history against user data.
* In our app there is no option such as mandatory or optional vaccines. This feature can be added to make the app more realistic.

***Conclusion:***

So far the Government of Bangladesh has expanded program on immunization (EPI), it is recognized as mTikka system. The mission of the EPI program in Bangladesh is to eradicate morbidity and mortality from vaccine-preventable diseases to levels where they are no longer a public health concern by providing high-quality EPI services to all children in the country. Vaccine reminder app will help to ensure timely vaccination.

***References:***

* <http://drfahimahmad.wordpress.com/2012/12/23/expanded-program-on-immunization-epi-a-bangladeshi-success-story/>
* <http://medinfobd.com/index.php?menuid=102>
* <http://www.apollodhaka.com/vaccination_centre.php>
* <http://www.iedcr.org/pdf/files/NPHC%20WEB/UPLOAD-3/Thematic%20Session/6_Dr.%20Md.%20Tajul%20Islam%20A.%20Bari.pdf>
* <http://ban.searo.who.int/LinkFiles/Publication_September-2012_EPI_Surveillance_Bulletin.pdf>
* <http://www.path.org/vaccineresources/files/Abt-PNACH278.pdf>
* <http://www.ibnsinatrust.com/ibn-hospital/children-vaccine-program>
* <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3702342/>
* <http://www.historyofvaccines.org/content/blog/vaccination-reminder-systems>