Smart Meet

Nath, Anindita

Bagavathyraj, Harshavardhini

Godinez, Karla

Cao, Zejing

Final Report

Human Computer Interaction

04/28/2017

Contents

1.Introduction

Title : SmartMeet

1.1 Objectives

SmartMeet is a smartphone application to coordinate the availability of multiple parties to schedule a meeting/event. The system will automatically determine the common available date and time to hold an upcoming meeting/event without much effort on the part of the user.

1.2 Features

- The creator of a group can modify it by adding or removing group participants.
- On each of the given date(s), input by the group creator, *SmartMeet* asks the group participants to select their corresponding available times from the application interface.
- Based on the inputs of all group participants , SmartMeet automatically picks the common available time.
- *SmartMeet* sends notification of the scheduled event date and time to each group participant.
- If no common time is available, SmartMeet will notify the same as well.

1.3 Distinct feature

- Use of mobile contacts instead of email-ids(as used by almost all scheduler applications available in the market) is the distinguishing feature.
- The primary utility of the feature is to enable the group-creator auto-search while adding the group-participants from among the contacts stored in his/her cell-phone.
- Mobile contacts are also used to send notifications to the users regarding the status of the event at any time, no matter wherever they are.This ensures that one does not miss out on any important information regarding the event owing to non-synchronization of their email-accounts in their phones or low speed data connectivity.

2.Usability

2.1 Target User-population

The main user of the application are college/university students. There is a huge market for this user base.Only in 2016, 17,490,000 undergraduate students : 7,528,000 male students, 9,962,000 female students and 3,025,000 graduate students : 1,279,000 male students, 1,746,000 female students enrolled to college or universities in the United States.
(Data from https://nces.ed.gov/programs/digest/d15/tables/dt15_105.20.asp?current=yes)

Key user-population characteristics

Users are familiar with smartphones, they know how to download, install and use any general application on a smartphone(preferably an Android phone). Age group-18 to 35.

2.2 Target Device

The application *Smart Meet* is designed exclusively for smartphones, preferably Android devices, because:

Relevance to the task

The tasks are essentially based designed for a smartphone application.

- Each group participant of a particular event is identified using his/her mobile number, saved in the contact list of the group creator.
- The application sends text notification(s) instead of emails to the participants as well as the group creator at various stages informing them about the status of the created event. And for completing this task,cell phone numbers of the group creator as well as of each of the event's participants are required.

Relevance to User-population characteristic

- The main users of the application have been designated as college/university students who are of the age group 18-35 and are expected to be familiar with the use of smartphones and has the minimum knowledge of downloading, installing and using any general application on a smartphone. The reason for choosing this category of population is the ever growing base of student population and its frequent need for updated/innovative applications assisting them in their daily/repetitive tasks. As is evident from recent statistics, 9 out of 10 Americans own a cell phone and about 58% of American Adults are smartphone users.(Source: Few Internet Project "Mobile Technology Fact Sheet", 2016).

- In particular, Android users are preferred because they more in number than IOS users.Statistical data reveal that IOS user base is around 40% while Android User base is around 50% of the total number of smartphone users in the United States.(Source: Kantar WorldPanel WorldTech, Q2 2016). Again, in general, Android phones are comparatively much cheaper than Iphones.

Relevance to Usability goals

- Smartphones, in general, are light-weighted, portable and can be accessed anywhere and at anytime unlike desktops,laptops or even tablets.
- Android OS is easier to use compared to IOS platforms. Even average smartphone users could easily become familiar with its interface with minimum frequency of use.

Other Aspects :

- From the developers point of view, we have more knowledge of programming for Android than for iOS, hence, our choice.
- It is easier to download applications without permission from Google Play Store than in iOS platform.
- Less investment of money is required while developing or launching any smart phone application than launching a website.

2.3 Usability Goals

- Speed of performance
  Group meeting is easy to schedule without much hassle on the part of the user since the system automatically picks the available date and time common to all group-participants.
- Time to learn
  As the intended user has knowledge with smartphone's latest technology, there's no need of user training or pre-learning.The application interface design is consistent with most of the basic smartphone applications available in the market.
- Rate of error by user
  Human error is considerably reduced since data entry is minimized as user input is basically through selection from the application interface (direct manipulation).
- Retention over time
  Because of its simple and familiar interface, even with less frequency of use, the users generally remember how to use the application.

2.4 Metrics for the goals

- Satisfaction
  The user satisfaction should be higher than 90%.
- Efficiency
  The system should be able to schedule the group event after the users have input their data and send instant notifications to all participants, taking approximately less than a minute.

3.Interface

3.1 Tasks supported

Sl.	Interface includes	User Tasks it supports
1.	Screen 1: Fields to input creator name and contact number.	Creation of a group.
2.	Screen 1: Field to search for participants names and contact with autocomplete feature and also option to add/edit/delete participants.	Creator adds participants to the group
3.	Screen 2: Fields to input event name, duration, and controls to select date and time; Option to add and view event details (time, name of event, etc).	Creation of an event.
4.	Screen 2: Option to send notification about event to the participants.	Creator invites participants to join the event.
5.	Screen 3: Controls to select time for the corresponding to the event date by the participants; option to view event details; option to notify the creator about participants availability.	Participants join the event.
6.	Screen 4: Option to confirm or cancel the meeting and sends notifications.	Creator schedules/cancels the meeting and informs the participants.

3.2 Interaction Style

The interaction styles suitable to the application are listed below:

3.2.1 Relevance to the task

- Menu-Based: In menu-based system applied here, users have three choices to select from namely,*Group(used for creation of a group), Event(used for creation of an event) and Meet(used to schedule the event)* and select the one appropriate to the required task. Also, users may toggle between two views namely *Creator view* and *Participants View*.However, *Group* and *Meet* menu choices, though present on the interface, are disabled in the *Participant's* view.
- Form fill-in: Data such as group-creator's *name* and *contact*, event's *title* and *duration* has to be filled in(in the corresponding text boxes) through the touch/soft keyboard by the creator. While adding members to the group/participants for the created event,the creator may search either by their names or contact numbers. If saved in the creator's contact list, auto-suggestions appear for completion.
- Direct manipulation: To add members to the group, direct manipulation style(clicking on the plus sign) will be suitable.Selection of available date and time is done by choosing from the calendar/digital clock control provided on the screen.

3.2.2 Relevance to User-population characteristic

The selected user base -college/university students who are smartphone users- is expected to be familiar with each of these interaction styles implemented in this application since they are common to one or the other general purpose smartphone applications available in the market.

3.2.3 Relevance to Usability goals

- Familiarity with the styles makes it easier for the user to retain knowledge as to their use over a long period of time.
- No need of user training or pre-learning with regards to navigating from one screen to another or handling of the controls since the layout simple and properly labelled.

4.Design

4.1 Low-fidelity prototype

Three design concepts were considered for the application, *Smart Meet* (sketches/paper mock-ups were prepared for the same) :

- Design-1 : the *multiple-screens design* with loads of form fill-in and data-entry through

keyboard/touchpad.

- Design-2 : *minimal screen design* concept with minimum form fill-in and data entry and minimum number of windows(three).Input is mainly through selection from the touchscreen interface.
- Design-3 : the design which is radically different from the above two includes implementation of the natural language interface. Inputs for every task are through voice commands from the user with no use of keyboard or touch-pad whatsoever.

4.1.1 Key aspects of the best design

The second design, Design-2,concept is what we like best because of the following reasons:

Relevance to the task

- Auto-completion of the name/contact number (from the creator's contact list) while searching for members to be added to the group by the creator adds to the efficiency of the application's performance.
- Selection of inputs from the interface reduces number of forms to be filled in.
- Though the menu-views have two variations : *Creator View* and *Participant View*, the same number of menu items are retained in both disabling only a few of them in the *Participant View*. This is done to maintain consistency of the interface design across all screens.
- This design is preferred over natural language interface because it is easier to implement. Also, effective and entire implementation of the natural language interface is not yet known and failure to understand and/or misinterpretation of voice commands may lead to disastrous results.

Relevance to User-population characteristic

- Fast Performance, easy manipulation of controls, fast and effective response from the system with no requirement of training or pre-learning are the basic requisites of this generation's smart phone users, especially students (our intended user base). The second design ensures just that.

Relevance to Usability goals

- Use of widgets and/or system interface input
  Minimal keyboard input or form fill-in - which only requires filling in two/three text boxes- restricting user input mainly to selection from the application interface(e.g. date/time selection through calendar/clock control) and thus, guaranteeing speed of performance.
- Minimum data entry also ensures reduced rate of errors.
- The application interface design, being consistent with most of the basic smartphone applications available in the market reduces time to learn to a considerable extent.

- Even with less frequency of use, simplicity of the application's design ensures retention over time.
- Also the use of text notifications instead of emails implies less chances of missing out on important information regarding the event due to non-synchronization of the users emails or poor internet connectivity/speed. This aspect adds to the subjective satisfaction of the users.

<table>
<tr><td colspan="2">Use Case: Selecting available time</td></tr>
<tr><td colspan="2">Description: The system selects the common available time
Actors: Creator and Group-Participants
Precondition: Group is created, Participants added as well as Event is created on a particular date by the Creator</td></tr>
<tr><th>Creator/Group participant</th><th>System</th></tr>
<tr><td>1. Each participant inserts his/her available time on the given date of the event</td><td>1. The system validates all participants have inserted time
2. The system automatically searches for the common available time on a g
3. The system schedules the event
4. The system sends notification to the participants with the date and time of the event</td></tr>
<tr><td colspan="2">Alternative:
Adding/deleting group participants.
Exceptional:
No common available time: the creator is asked to repeat the process for another tentative date.</td></tr>
</table>

4.1.2 Use-Case Diagram

4.1.3 Screen-shots of Interface design

- Task- Creation of a group
- View : Creator View

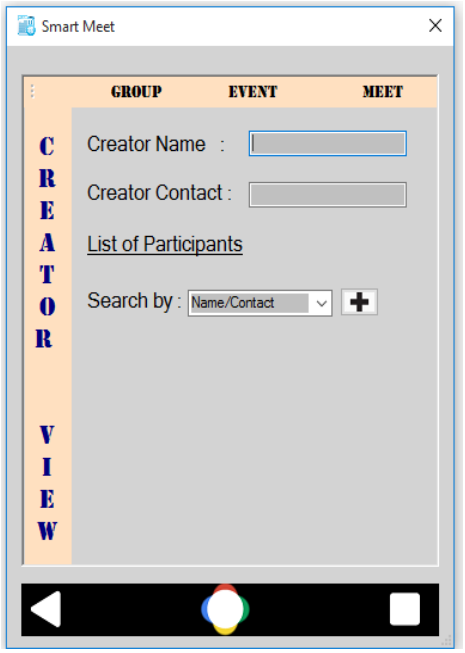


Fig 1: Input creator info (name,contact)

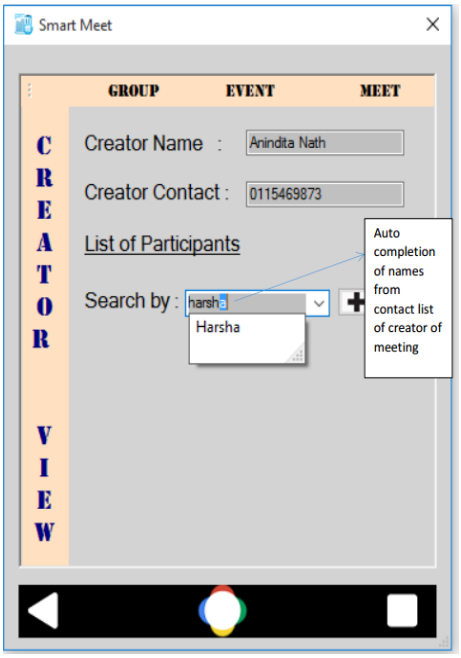


Fig 2: Search for participants

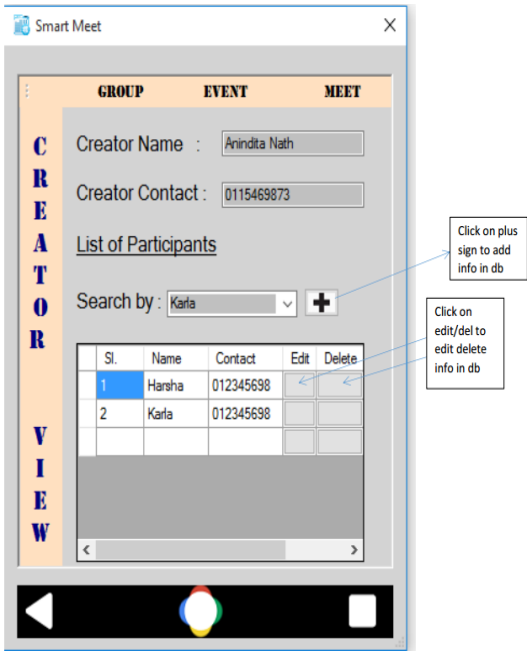


Fig 3: Add/Edit/Delete participants

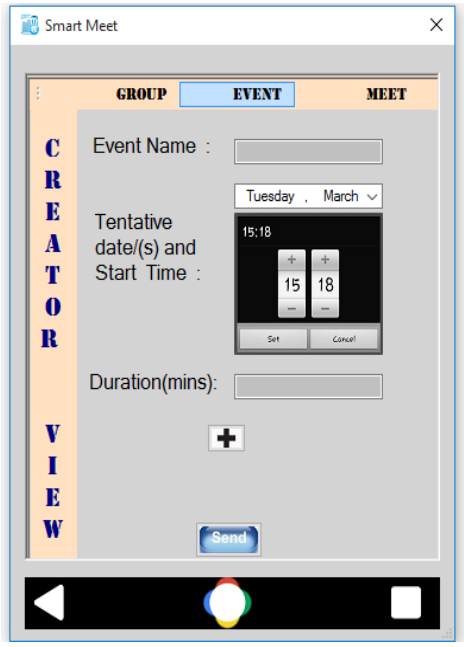


Fig 4: Event screen

- Task- Creation of an event
- View : Creator View

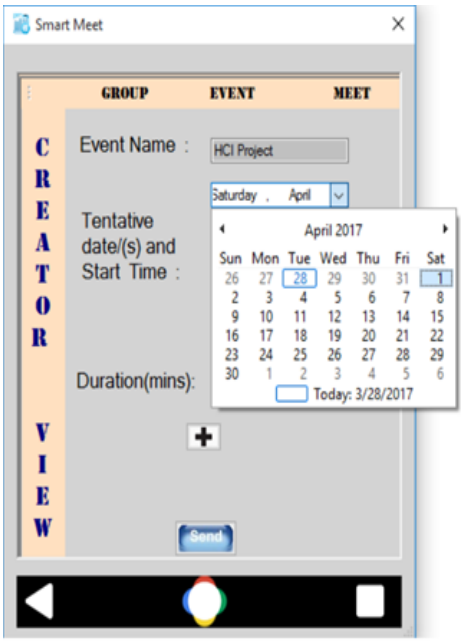


Fig 5: Input event date and time

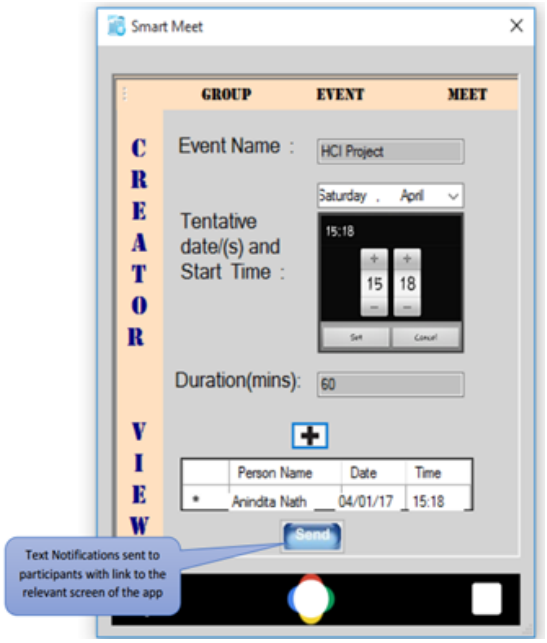


Fig 6: Notify participants

- Task- Schedule/Cancelevent
- View : Creator View

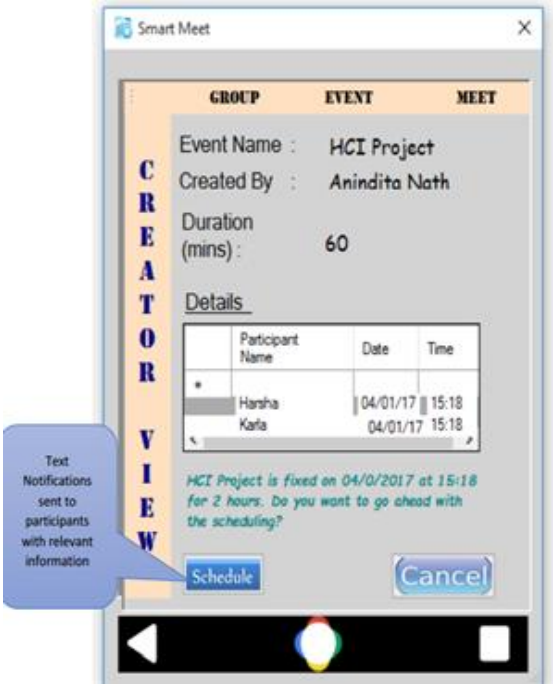


Fig 7: Confirm/schedule event, notify all

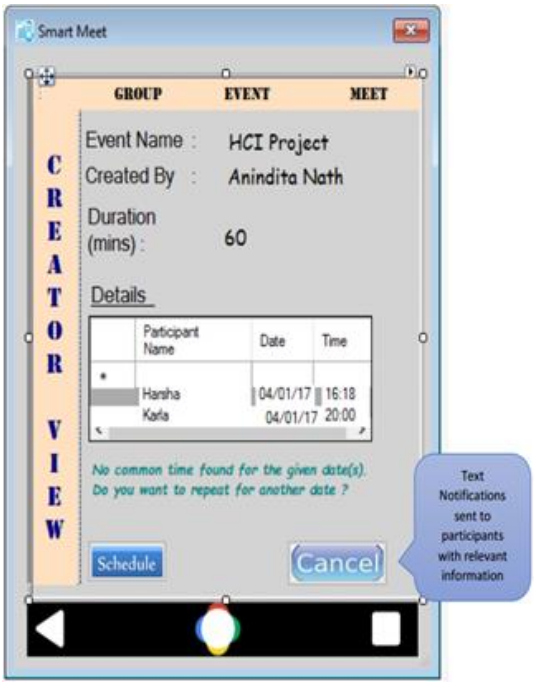


Fig 8: Cancel event, notify

- Task- Participants availability
- View : Participant View

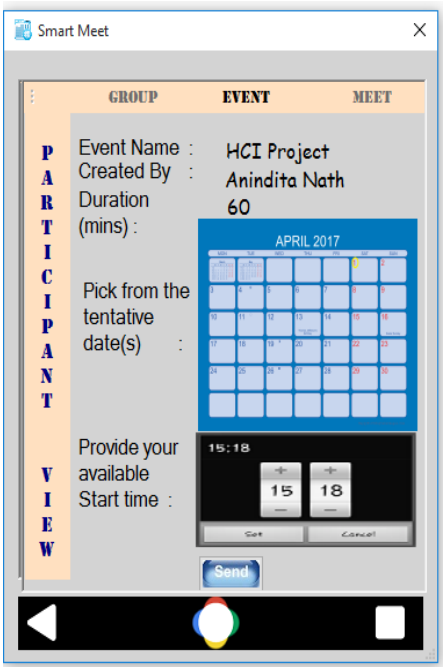


Fig 9: Participants view of event screen

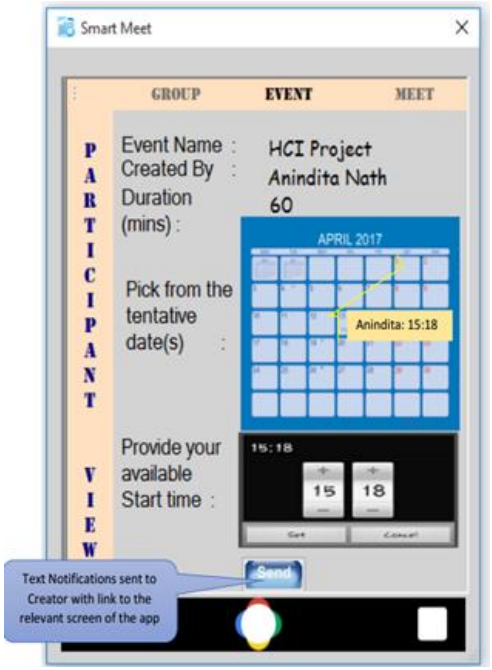
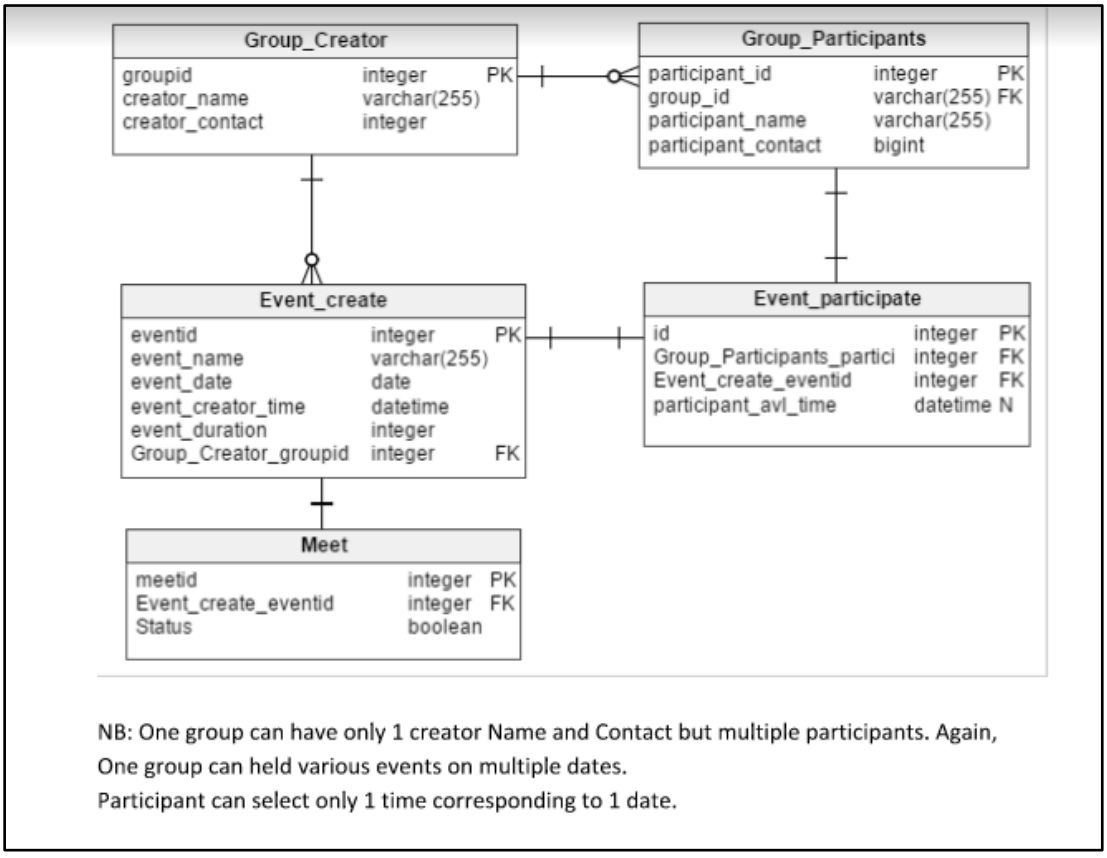


Fig 10: Input available time,notify creator

4.2 Schema DataBase



5.Implementation

5.1 Techno

logy Used

- Android Studio(2.3.1), for designing the interface
- Java as the programming language
- SQLite for the back-end (as the RDBMS)

5.2 Medium-fidelity Prototype

SmartMeet application is only in its first stage of implementation. Due to time constraint and lack of professional experience with the design/implementation of an Android Application , some of the features could not be implemented, and some back-end functions had to be restricted to a modified simple version.

Below are listed the features of the medium fidelity prototype :

- Coding for designing the GUI is complete following the basic design principles/guidelines. (for code, refer Appendix I)

5.2.1 Minimum Back-end implemented :

- Store the names and contacts of the participants in the database to facilitate auto-completion while creator searches for them to add to the group.
- As the creator/participant of the group clicks on the "+" sign after filling in all the required fields on the screen, relevant information (user details : name and contacts, available times, event details : name, date and time , duration,etc.) gets updated automatically in the corresponding fields of the database.
- The details corresponding to each participant are fetched and presented in a list-view on screen.
- The application is able find the input available time common to all group-members and schedules the event accordingly.

5.3 Challenges

1. It is difficult/time consuming to implement the feature, namely, sending text notification to participants/ creator informing the status of the event/meeting.
2. Due to time constraint, it is difficult to implement the auto retrieval of participant's contact number from the creator's contact list as he/she inputs the participant name while creating a group.
3. Combining the proposed features/functionalities with the interface design to create the high-fidelity prototype because of time constraint.

5.4 Future work

Listed below are the features that need to be implemented to fully deploy as well as enhance the utility of the application :

- SmartMeet needs to be fully implemented with complete back-end functionalities in order to be deployed in the Google Play-Store. Also, to create a good and effective interface, each and every flaw observed in the evaluation processes needs to be corrected.
- Import contacts of participants from creator's contact list
- Sending notifications to every participant/creator with appropriate links to relevant pages of the application.
- Letting the participants modify the event's date as well.
- Adding the option to include the option to input "meeting place/ event venue" by every participant so that the application will auto-filter the common venue for the event.
- Having an alternate option, instead of just sending notification to the user, if SmartMeet fails to determine a common available time for the meeting/ event.
- Develop the application with cross-platform functionalities so that it works with iOS as well.

6.Evaluation

6.1 Heuristic Evaluation

After evaluating the preliminary design of the initial prototype, how users interacted with the interface to perform the given tasks, each heuristic rule was rated on a scale of 0-4:

0 → I don't agree that this is a usability problem at all
1 → Cosmetic problem
2 → Minor usability problem
3 → Major usability problem
4 → Usability catastrophe

Screen 1: Fields to input creator name and contact number. Field to search for participants names and contact with autocomplete feature and also option to view/add/edit/delete participants.

Heuristic	Rating	Comments
Visibility of system status	0	The system immediately gives user feedback as the name/contact is input, it is visible on screen.
Match between system and the real world	0	The system uses language familiar to the user, no confusion for the user.
User control and freedom	1	The user can undo or redo the action by touching the left arrow button.However, a “Back” or “ Reset” button would have been effective to reset all fields at the same time.
Consistency and standards	0	The system maintains consistency of layout and language throughout all screens.
Error prevention	3	No validation checking appears for contact field. No information as to whether a mobile number has to be input or land-line. No error/alert message if the format of the mobile number is wrong.
Recognition rather than recall	0	The user does not need to remember any information, relevant information are displayed on screen.
Flexibility and efficiency of use	1	The system does not support keyboard shortcuts.
Aesthetic and minimalist design	1	No extra or irrelevant information in the system. However, the labels are too wordy and font -size is quite large, consume a lot of space and not visually appealing either.
Help users recognize, diagnose and recover from errors	2	No confirmation/ alert message.
Help and documentation	2	The application does not come with any help/ documentation guide/tutorial.

Screen 2: Fields to input event name, duration, and controls to select date and time; Option to add and view event details. Option to send notification about event to the participants.

Heuristic	Rating	Comments
Visibility of system status	0	Blinking of cursor in the respective fields indicative of input required to be inserted. Also, presence of digital calendar and clock indicates direct manipulation required to input event date and creator's available time. Plus sign to add the information and Send button to send notifications are obvious.
Match between system and the real world	0	The system uses language familiar to the user, no confusion on the part of the user to understand the instructions.
User control and freedom	1	The user can undo or redo the action by touching the left arrow button.However, a "Back" or " Reset" button would have been effective to reset all fields at the same time
Consistency and standards	0	The system maintains consistency of layout and language.
Error prevention	2	No pop-up/message appears to confirm whether the event details has been entered correctly.
Recognition rather than recall	0	No need of information recall in this screen.
Flexibility and efficiency of use	0	The system does not support keyboard shortcuts but the direct manipulation features provides ease and speed of use.
Aesthetic and minimalist design	2	No extra or irrelevant information in the system. However, the calendar interface almost overlaps the digital clock. Also, the presence of a large digital clock, almost always, even when the user is not using it goes against the concept of minimalist design.
Help users recognize, diagnose and recover from errors	2	No any error/ alert message if the time or date is entered in wrong format.
Help and documentation	2	The application does not come with any help/ documentation guide/tutorial.

Screen 3: Controls to select participants available time for the corresponding event date; option to view event details; option to notify the creator about participants availability.

Heuristic	Rating	Comments
Visibility of system status	0	All the relevant event details, namely, name, creator's name , event date and creator's available time appear on the screen.The clock control to input time is apparent.
Match between system and the real world	0	The system uses language familiar to the user, no confusion regarding the instructions.
User control and freedom	1	The user can undo or redo the action by touching the left arrow button.However, a "Back" or " Reset" button would have been effective to reset all fields at the same time
Consistency and standards	0	The system maintains consistency of layout and language throughout.
Error prevention	3	No error/alert message or validation checking. No information/instruction as to what the user should do if the time remains same as that of the creator. Does they still need to set time and notify creator?
Recognition rather than recall	0	The user should not have to remember/recall any information in this screen.
Flexibility and efficiency of use	0	The system does not support keyboard shortcuts but the direct manipulation features provides ease and speed of use.
Aesthetic and minimalist design	1	No extra or irrelevant information in the system but the fonts colour or size could be changed to make the instructions legible.
Help users recognize, diagnose and recover from errors	2	No confirmation message appears on setting the time. No proper feedback.
Help and documentation	2	The application does not come with any

		help/ documentation guide/tutorial.

Screen 4: Option to confirm or cancel the meeting and send notifications.

Heuristic	Rating	Comments
Visibility of system status	0	All the relevant event details, namely, name, participant and event details appear on the screen. Schedule and Cancel buttons have their proper meanings/usability.
Match between system and the real world	0	The system uses language familiar to the user, no confusion regarding the instructions.
User control and freedom	1	The user can undo or redo the action by touching the left arrow button.However, a "Back" or " Reset" button would have been effective to reset all fields at the same time
Consistency and standards	0	The system maintains consistency of layout and language throughout.
Error prevention	0	No error/alert message or validation checking required on this screen.
Recognition rather than recall	0	The user does not have to remember/recall any information on this screen.
Flexibility and efficiency of use	0	The system does not support keyboard shortcuts but the direct manipulation features provides ease and speed of use.
Aesthetic and minimalist design	1	No extra or irrelevant information in the system but the fonts colour or size could be changed to make the instructions legible.
Help users recognize, diagnose and recover from errors	2	No confirmation message appears after sending notification. No proper feedback.
Help and documentation	2	The application does not come with any help/ documentation guide/tutorial.

6.2 Cognitive Walkthrough

We conducted cognitive walkthrough of two of the important user-tasks using the best proposed interface design (Design-2)

Assumption: For our cognitive walkthroughs we are assuming that the user is familiar with the use of smartphone applications, in general.

Cognitive Walkthrough-1

The task chosen for the cognitive walkthrough is the first and one of the vital tasks in the application :

Create a group adding members to it.

The following is the action sequence for this task specified in terms of the user's actions (Action) and the system's response (Response):

Action	**A**:	Select the "*Group*" menu
Response	**A**:	The screen changes to a form fill-in lay out with the cursor blinking at the beginning of the first control(text box).
Action	**B**:	Input Creator name
Response	**B**:	Cursor position changes to the beginning of next text box.
Action	**C**:	Input Creator contact
Response	**C**:	Validation check appears if input is in wrong format else cursor shifts to the next control.
Action	**D**:	Type name or contact number to add members.
Response	**D**:	Auto suggestions of the name appears for completion.
Action	**E**:	Click on +.
Response	**E**:	The details (name and contact number) of the added member appear in a tabulated form with options to edit/delete information.
Action	**F**:	Repeat from A to E.
Response	**F**:	The screen refreshes resetting all the controls to default and cursor to the beginning of first control.

Proceeding with the walkthrough :

Having determined our action list, we proceed with the walkthrough answering three questions for each action and telling a story about the usability of the system in each one of them.

Action A: Select the "*Group*" menu

Question 1: Will users be able to notice that the correct action is available?

Yes, on the top of the screen in the coloured rectangular bar, "Group" menu item is distinctly visible.

Question 2: Once users find the correct action at the interface , will they know that

it is the right one for the effect they are trying to produce?
Since the task says “Create a group adding members to it”, the name of the menu-item itself indicates that selecting it is, in fact,the right action.
Question 3: After the action is taken, will users understand the feedback they get?
Yes, as soon as the screen with the form fill-in layout appears with cursor blinking at the beginning of the first control, the user is assured of his right choice.

Action B: Input Creator name
Question 1: Will users be able to notice that the correct action is available?
Yes, the blank text box beside the label Creator Name is distinctly visible.
Question 2: Once users find the correct action at the interface , will they know that it is the right one for the effect they are trying to produce?
Since the cursor blinks at the beginning of the text box, it indicates that user input is expected.
Question 3: After the action is taken, will users understand the feedback they get?
The change of the cursor position to the next control indicates that this action is successful.

Action C: Input Creator contact
Question 1: Will users be able to notice that the correct action is available?
Yes, the blank text box beside the label Creator Contact is distinctly visible.
Question 2: Once users find the correct action at the interface , will they know that it is the right one for the effect they are trying to produce?
Since the cursor blinks at the beginning of the text box, it indicates that user input is expected.
Question 3: After the action is taken, will users understand the feedback they get?
The change of the cursor position to the next control indicates that this action is successful.
Action D: Type name or contact number to add members.
Question 1: Will users be able to notice that the correct action is available?
Yes, to any typical smartphone user, it will be clear enough that he has to start typing from the point where the cursor is present.
Question 2: Once users find the correct action at the interface , will they know that it is the right one for the effect they are trying to produce?
Appearance of “Search by” and “Name/Contact” suggests clearly that one may start typing in either the name or the contact number of the member.
Question 3: After the action is taken, will users understand the feedback they get?
As the user types the name/number, auto-suggestion appears for completion indicating that the user is on the right track.

Action E: Click on +.
Question 1: Will users be able to notice that the correct action is available?
Yes, '+' sign is distinctly visible.
Question 2: Once users find the correct action at the interface , will they know that it is the right one for the effect they are trying to produce?
In accordance with the general notion, '+' sign is synonymous with ADD.
Question 3: After the action is taken, will users understand the feedback they get?
As the member details appear on screen in a tabulated form, the user is assured that the member has been added to the group.

Action F: Repeat from A to E.
Question 1: Will users be able to notice that the correct action is available?
No "Reset" button present on screen. It might be confusing to know how to proceed further.
Question 2: Once users find the correct action at the interface , will they know that it is the right one for the effect they are trying to produce?
It is quite obvious that the user will understand that repetition of the process is done to continue adding members.
Question 3: After the action is taken, will users understand the feedback they get?
Resetting of all text-boxes to blank with cursor blinking at the beginning of the first one indicates to the user that the process has, indeed,started again.

Cognitive Walkthrough-2

The task chosen for the cognitive walkthrough is the first and one of the vital tasks in the application.

Participants Select of Available time.
The following is the action sequence for this task specified in terms of the user's actions (Action) and the system's response (Response):

Action	**A**:	Tap on highlighted date in the calendar.
Response	**A**:	Pop-up appears showing the available time entered by the creator.
Action	**B**:	Tap on the screen.
Response	**B**:	Pop-up disappears.
Action	**C**:	Tap on time field to input his/her available time.
Response	**C**:	The time selected by the user is displayed.

Action	**D**:	Touches ‘set’ to confirm/change the time.
Response	**D**:	Time input by the user is displayed.
Action	**E**:	Touches “Send” button.
Response	**E**:	Pop-up appears with the message “Creator is notified about your available time on selected date”.
Action	**F**:	Repeat from A to E to input available time for another highlighted date.
Response	**F**:	The screen refreshes resetting all the controls to default.

Proceeding with the walkthrough :
Having determined our action list, we proceed with the walkthrough answering three questions for each action and telling a story about the usability of the system in each one of them.
Action A: Tap on highlighted date in the calendar.
Question 1: Will users be able to notice that the correct action is available?
Yes, the calendar besides the label “Pick from the tentative date” is distinctly visible.
Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?
Yes, because as the user taps on the highlighted date, pop-up appears proving that some action has taken place.
Question 3: After the action is taken, will users understand the feedback they get?
Yes, as soon as the user taps on highlighted date, the screen pops-up the available time entered by the creator whose name is already present on the screen. However, the user may not expect this feedback for the action.

Action B: Tap on the screen.
Question 1: Will users be able to notice that the correct action is available?
No, the user does not know where to tap on the screen and might also be unsure of the intended purpose.
Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?
Yes, since once the user taps on the screen, pop-ups disappears from the screen revealing the purpose of the action.
Question 3: After the action is taken, will users understand the feedback they get?
Yes, the disappearance of the pop-up from the screen is an informative feedback.

Action C: Tap on time field to input his/her available time.
Question 1: Will users be able to notice that the correct action is available?
Yes, the digital clock beside the label “Provide your available Start time” is distinctly Visible. However, one might not be sure of this choice of action if his/her available time

matches with that of the creator, already displayed on screen.

Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?

Yes, since the time gets changed in the time-field, it indicates that the user has input what is expected.

Question 3: After the action is taken, will users understand the feedback they get?

The display of chosen time indicates that this action is successful.

Action D: Touches 'set' to confirm/change the time.

Question 1: Will users be able to notice that the correct action is available?

Yes, the "Set" button is distinctly visible. Any average smartphone user would understand its functionality.

Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?

Yes, once the user decides the time, user knows to tap on "Set" button to confirm the time.

Question 3: After the action is taken, will users understand the feedback they get?

No, no change/display appears on screen, so the user does not get any feedback regarding the validity of his/her action.

Action E: Touches "Send" button.

Question 1: Will users be able to notice that the correct action is available?

Yes, 'Send' button is distinctly visible.

Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?

Well, the user(assumed to be familiar with smartphone applications) is expected to know the need to notify the creator of his/her available time.

Question 3: After the action is taken, will users understand the feedback they get?

Yes, pop-up appears with the message "Creator is notified about your available time on selected date".

Action F: Repeat from A to E to input available time for another highlighted date.

Question 1: Will users be able to notice that the correct action is available?

Though refreshing or reloading of the screen is a basic task of all smartphone applications and the intended user base is expected to have the knowledge of it, no specific control is present on the screen to accomplish the task.

Question 2: Once users find the correct action at the interface, will they know that it is the right one for the effect they are trying to produce?

It is not quite clear to the user whether the repetition of the process is required to input another available time for same date or another date.

Question 3: After the action is taken, will users understand the feedback they get?

No controls/fields are reset, so the feedback does not reflect the correctness of the action.

6.3 Observation

We used the observation technique to observe users interacting with the SmartMeet application, an effective way to understand the usability of the system and also, to some extent the overall user experience. We recruited two users (user1-acting as the group as well as event creator/ and user2-acting as the group-participant, while using the application)- both were UTEP students from Bioinformatics department, who are acquainted with Smart Phone applications in general. We also explained the users what they will be doing (tasks) and what they will be observed for and how their observation data will help us. We made observation using medium-fidelity prototype

Sl. No.	Task	User-1
1.	Create group	● Fields like Creator Name, Creator Contact were informative enough. Cursor blinking in the respective controls indicates that the input is required. ● No validation checking for mobile number format. User left confused whether to add country code, etc. ● User faced no trouble in adding participants and did it fairly quickly. Auto-completion of names helped them to complete the task quickly. ● The details of the participants added could also be viewed in the screen. ● "Participant added successfully", message pops-up as an indicative feedback that the task is performed successfully. ● User was, however, not sure what to do with the Edit and delete buttons, whether they were relevant to the given task - create group. ● User completed the task in 2 minutes.

2.	Create Event and Notify	• Fields like Event Name, Duration, Event Date and Tentative time were informative enough. Cursor blinking in the respective controls indicates that the input is required. • Appearance of calendar corresponding to date field, and the clock interface corresponding to the time field helped the user to input date and time without any hassle and in a quick manner. • Plus sign indicates it needs to be clicked to add the event details. Message "Event added successfully" pops up in the screen to provide correct feedback about successful creation of the event. • It is obvious to the user that Send button is required to send notifications to the participants regarding the event created. • User completed the task in 1 minute.

Sl. No.	Task	User-2
3.	Input Participant's availability and notify	• Event details like name, duration, creator name, creator available date and time appears on screen. • However, user is not sure where to tap on the calendar to view the information about creator's input time of the event. • Once the user figures it out and the pop-up reveals the creator's input time, the user is confused what to do to make the pop-up disappear from the screen. • Clock interface indicates setting of time is required. But the user is not sure what is required of him/her to do if his/her available time matches with that of the creator and he/she is not required to change the time. User, however, after much thought leaves the time as it is and clicks on send to notify the creator of his decision.

		• User completed the task in 2 minutes.
SI.No	**Task**	**User-1**
4.	Schedule/ Cancel event and notify	• The creator receives all the participants details: name, contact, available time; it automatically appears on screen. • The application schedules the event by filtering the common available time of all users and shows it as a message on screen. If no common time is found, a message "No common time found for the given date(s). Do you want to repeat for another date?" appears on the screen. • It is pretty apparent to the user that he needs to click on schedule or cancel to confirm/cancel event respectively. However, he is unsure as to what to do to repeat the process. • He is not sure how to send the notifications. Though, an immediate pop-up reveals that the notification has been sent. • The user completed the task in less than a minute.

7.Roles and Responsibilities

Activities	***Target Completion date***	***Per Person-hours required***	***Person responsible***

Ideation			February 12	0.5-1	Karla Godinez
Design and Implementation of prototypes	Initial prototype (Sketches,Paper Mock-ups)	Design -1	March 28	1-2	Karla Godinez
		Design -2		1-2	Anindita Nath
		Design-3		1-2	Harsha B
	Medium Fidelity Prototype	GUI coding	April 25	12-15	Anindita Nath, Harsha B
		Minimum Back-end	April 26	12-15	Anindita Nath, Harsha B
Usability Testing and Evaluation	Heuristic Evaluation of Initial Prototype		April 1	2-3	Zejing Cao
	Cognitive Walkthrough-1 of Initial Prototype		April 2	1-2	Anindita Nath
	Cognitive Walkthrough-2 of Initial Prototype		April 2	1-2	Karla Godinez
	Observation of Medium Fidelity Prototype		April 24	2-3	Zejing Cao, Harsha B
Final Report			April 27	3-5	Anindita Nath,Harsha B,Karla Godinez,Zejing Cao
Presentation			April 30	1-2	Anindita Nath,Harsha B,Karla Godinez,Zejing Cao

8.Skills Acquired

While working on the project, we tried to apply the knowledge obtained from the course contents, namely, Usability Goals, Design Guidelines, Interaction Styles and Usability testing methods/ Evaluation Process as listed below :

- Learned to analyze the needs of the user in order to have a useful project idea.
- Analyze the challenges and risks to have a concrete, feasible idea.
- Create a low-fidelity prototype (use of sketches, paper mock-ups).
- Apply basic design guidelines and principles such as Contrast, Repetition, Alignment and Proximity to make an attractive design.
- Learned to implement the most effective interaction styles to complete SmartMeet tasks.
- Choose the appropriate evaluation method, namely, Cognitive Walkthrough, Heuristic Evaluation and Observation method, based on cost and time.
- Create a medium-fidelity prototype based on the analysis and redesign of the low-fidelity prototype.

8.1 Technical

- Learned the basics of Android Studio (with prior knowledge of coding in Java) and applied the knowledge to develop the demo-application.

8.2 Managerial

- Working in teams made the work more productive: Coordination among the team members with proper allocation/division of work.
- Time-management skills required to finish each project-deliverable within the given deadline.