Developer Documentation for GroupAlarm (ASIJGE)

1 Project Plan

1.1 Vision

In every group project there is always that one person who arrives late to every meeting. After a while it becomes disturbing for the rest of the group when every meeting starts with trying to contact the sleepyhead of the group. This also usually turns out to be a waste of time since most people turn off the alarm signal on their phone during the night. The other group members have to sit and wait for their last member or start the meeting without him/her and then having to explain everything that was on the meeting when the sleepyhead arrives.

The vision behind GroupAlarm is therefore to create an application which allows a group of users to intervene with the other members habits for getting up in the morning (or afternoon for that sake…). By doing this it increases the chance for all of the members to be on time and to get the work started properly. There shall no longer be a way to turn off the clock and pretend to not have heard it or to "forget" to set an alarm - the group will know. The group of users will be able to control every alarm set by this app due to notifications whenever a group member snoozes, stops or ignores his/hers alarm.

1.2 Features

The user stories with feature tags 'group' and 'event' is the far most important to be able to fulfill the vision, otherwise the app is just an alarm clock (which already exists on every smartphone). Yet the existence of an alarm clock is crucial in order to create the other functions.

Feature tag	User story
alarmclock	As a user I want to create a personal alarm.
alarmclock	As a user I want to be able to edit the alarms that I've created.
alarmclock	As a user I want to be able to activate and deactivate alarms in the alarm list view.
alarmclock	As a user I want to be able to view my alarms by time and day.
alarmclock	As a user I want to be able to stop my alarm so that it won't ring again.
alarmclock	As a user I want to be able to snooze my alarm so that I may continue sleeping for a few more minutes before it rings again.
alarmclock	As a user I want to be able to choose a snooze interval when I create or edit an alarm.

alarmclock	As a user I want to be able to choose time and weekday when I create or edit an alarm.
alarmclock	As a user I want to be able to set a message when I create or edit an alarm.
app	As a user I have to create a user account to use the application.
alarmclock	As a user I want to be able to remove alarms that are no longer relevant.
group alarm	As a group member I want to create and submit a group alarm so that it´s visible to my group members.
group alarm	As a user I want to be able to view all the alarms for a certain group I am in.
group alarm	As a group member I want to be able to change the alarm signal for a group member who is snoozing.
group alarm	As a group member I want to be able to create an alarm for my group.
group alarm	As a group member I want to be able to edit an alarm for my group.
group alarm	As a group member I want to be able to delete an alarm for my group.
group alarm	As a user I want to be able to view my alarms filtered by my groups.
group	As a group member I want to get notified whenever a group member snoozes, stops or ignores his/hers alarm.
group	As a user I want to be able to create a group.
group	As a user I want to be able to invite other users to a group that I'm a member of.
group	As a user I want to be able to accept group invitations from other users so I can join groups.
group	As a group member I want to be able to leave a group that I'm a member of.
group	As a group member I want to be able to set different weekdays for the different alarms for my group.
group	As a user I want to be able to view the groups that I'm a member of.
group	As a user I want to be able to leave a group as the last remaining member so that the group is resolved.

1.3 Roles

Product owner: Håkan Burden

Scrum master: Alexander Hederstaf

Developer team: Alexander Hederstaf, Sebastian Odbjer, Isabel Azcárate, Joakim Sehlstedt, Gabriella Hallams and Emma Wibel

2 Software Development Model

Two different software development models are the sequential model and the iterative model. When working according to the sequential model, the tasks of the development is handled in a sequential order and consists of requirements, design, implementations and quality assurance.

The development process of this project has been iterative. The iterative model constantly alter between design, implementation, requirements and quality assurance. Because the different phases has been handled several times during this development process, no specific order of the phases have existed.

This application has been an innovative project with a clear goal. How to reach the goal was discovered along the way which required a process that could handle constant change. The upcoming steps of the development process was not always known long before which made the planning process harder. The choice to work less with defined process control was based on our less static environment. Instead the choice of a way to work was empirical process control with short iterations.

3 Design Decisions

3.1 API level

API level is a value that identifies the framework that can be used with a certain Android platform. It consists of a number of classes, packages, XML elements, Intents and permissions for the application to request.

The design of the application is based on Android SDK 16 and can be used with the version 4.1 Android platform. The choice of API level was motivated by the amount tools and functionality available for development of the application combined with the number of users able to reach the application. A higher API level requires a newer phone and result in less available users.

The API level 16 uses a version code known as Jelly Bean and the original version was released in June 2012. The earlier version code was called KitKat. An update to Jelly Bean offers the users and developers new features and include big improvements regarding design as well as smoothness and user feeling. The goal with the choice of version code was to create an application with a good user experience

3.2 Pure Android/User interface

Pure Android was the goal while building the application. The design of the application is based on the Google Android guidelines and the user interface and icons creates a recognizable environment for an Android user. The GroupAlarm application uses common Android interface options such as dialogs, actionbars and context menus. Regarding design decisions in Android it is important to not forget that the interface will look different depending on the brand version of the phone. To accommodate this we used mostly the built in graphical resources of the Android system. This makes sure that no detail will look out of place on any phone.

3.3 Database

We chose to use a local database for personal alarms and a server database for group alarms. This allows the user to use the application as a normal alarm while not having Internet connection. For the group alarms we chose to use a server database to make the application open and reachable for everyone connected to the cloud. The server chosen was Parse.

3.4 Parse.com

To enable the communication between the users and also for the storage of common alarms we choose to use Parse Core for the network. Parse.com is a backend for developers providing

scalable and secure solutions for software projects. We decided to use their products because they are well known and widely used by many developers. We also wanted to simplify the expansion of GroupAlarm to other operating systems and to enable easy connection with other apps in a possible future.
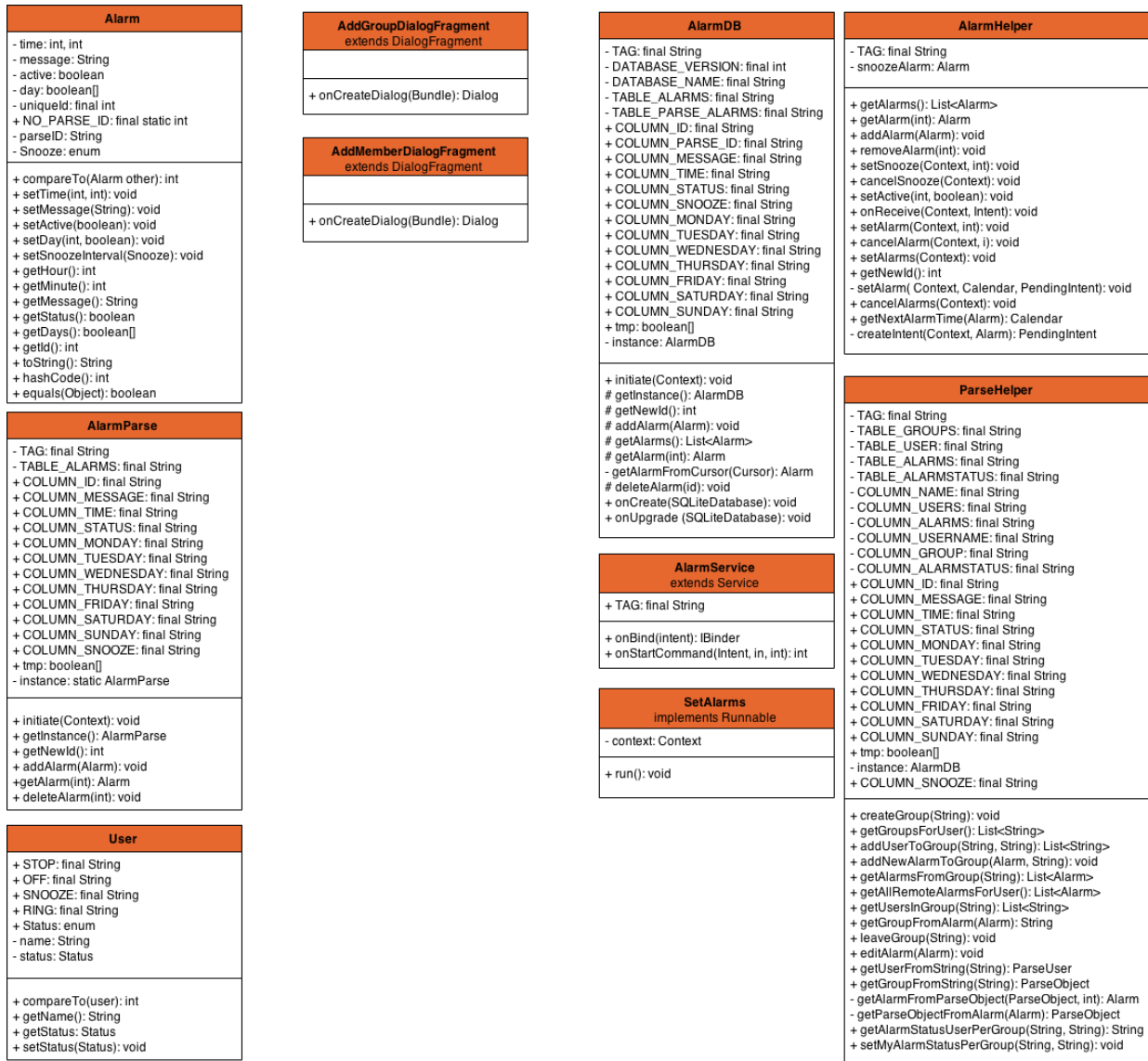
While making the desicion of using Parse.com products, we indirectly choose that the application will be depending on Internet connection. This is a fact that could have a bad impact on the applications business value. A further disadvantage of using a server is that the phone has to be connected to the internet for the alarm to go off. But we believe that since the data ratio GroupAlarm takes in consideration is very small and with the extension of the 3G and 4G, that the GroupAlarm users will be able to use the application without problems due to this fact.

The Parse products we use are Parse Core and Parse Push. Parse Core takes care of all the cloud data in our application and allows scalability and communication between the users. The group alarms are stored in the Parse Core server and are reachable for all group members. To notify the group of the activities by the other group members we choose Parse Push. It takes care of the applications different notifications and is crucial for the GroupAlarm's image and business value. The snooze notifications are the core element in the GroupAlarm-idea.

3.5 Would we make the same decisions again?

Due to lack of knowledge and further investigations this question is hard to answer. With the same amount of time we probably would have done it the same way just because the limited time forced us to chose solutions we had heard of beforehand. The fact that we knew about the chosen solutions since previous courses or project gives reason to believe that they are usable and suitable. Given the chance to do the project again with a larger amount of time the design decisions would hopefully be the same because we all hope that we made the best possible decisions, despite of no further analysis.

4 UML

Alarm

- time: int, int
- message: String
- active: boolean
- day: boolean[]
- uniqueId: final int
+ NO_PARSE_ID: final static int
- parseID: String
- Snooze: enum

+ compareTo(Alarm other): int
+ setTime(int, int): void
+ setMessage(String): void
+ setActive(boolean): void
+ setDay(int, boolean): void
+ setSnoozeInterval(Snooze): void
+ getHour(): int
+ getMinute(): int
+ getMessage(): String
+ getStatus(): boolean
+ getDays(): boolean[]
+ getId(): int
+ toString(): String
+ hashCode(): int
+ equals(Object): boolean

AlarmParse

- TAG: final String
- TABLE_ALARMS: final String
+ COLUMN_ID: final String
+ COLUMN_MESSAGE: final String
+ COLUMN_TIME: final String
+ COLUMN_STATUS: final String
+ COLUMN_MONDAY: final String
+ COLUMN_TUESDAY: final String
+ COLUMN_WEDNESDAY: final String
+ COLUMN_THURSDAY: final String
+ COLUMN_FRIDAY: final String
+ COLUMN_SATURDAY: final String
+ COLUMN_SUNDAY: final String
+ COLUMN_SNOOZE: final String
+ tmp: boolean[]
- instance: static AlarmParse

+ initiate(Context): void
+ getInstance(): AlarmParse
+ getNewId(): int
+ addAlarm(Alarm): void
+getAlarm(int): Alarm
+ deleteAlarm(int): void

User

+ STOP: final String
+ OFF: final String
+ SNOOZE: final String
+ RING: final String
+ Status: enum
- name: String
- status: Status

+ compareTo(user): int
+ getName(): String
+ getStatus: Status
+ setStatus(Status): void

AddGroupDialogFragment
extends DialogFragment

+ onCreateDialog(Bundle): Dialog

AddMemberDialogFragment
extends DialogFragment

+ onCreateDialog(Bundle): Dialog

AlarmDB

- TAG: final String
- DATABASE_VERSION: final int
- DATABASE_NAME: final String
- TABLE_ALARMS: final String
- TABLE_PARSE_ALARMS: final String
+ COLUMN_ID: final String
+ COLUMN_PARSE_ID: final String
+ COLUMN_MESSAGE: final String
+ COLUMN_TIME: final String
+ COLUMN_STATUS: final String
+ COLUMN_SNOOZE: final String
+ COLUMN_MONDAY: final String
+ COLUMN_TUESDAY: final String
+ COLUMN_WEDNESDAY: final String
+ COLUMN_THURSDAY: final String
+ COLUMN_FRIDAY: final String
+ COLUMN_SATURDAY: final String
+ COLUMN_SUNDAY: final String
+ tmp: boolean[]
- instance: AlarmDB

+ initiate(Context): void
# getInstance(): AlarmDB
# getNewId(): int
# addAlarm(Alarm): void
# getAlarms(): List<Alarm>
# getAlarm(int): Alarm
- getAlarmFromCursor(Cursor): Alarm
deleteAlarm(id): void
+ onCreate(SQLiteDatabase): void
+ onUpgrade (SQLiteDatabase): void

AlarmService
extends Service

+ TAG: final String

+ onBind(intent): IBinder
+ onStartCommand(Intent, in, int): int

SetAlarms
implements Runnable

- context: Context

+ run(): void

AlarmHelper

- TAG: final String
- snoozeAlarm: Alarm

+ getAlarms(): List<Alarm>
+ getAlarm(int): Alarm
+ addAlarm(Alarm): void
+ removeAlarm(int): void
+ setSnooze(Context, int): void
+ cancelSnooze(Context): void
+ setActive(int, boolean): void
+ onReceive(Context, Intent): void
+ setAlarm(Context, int): void
+ cancelAlarm(Context, i): void
+ setAlarms(Context): void
+ getNewId(): int
- setAlarm(Context, Calendar, PendingIntent): void
+ cancelAlarms(Context): void
+ getNextAlarmTime(Alarm): Calendar
- createIntent(Context, Alarm): PendingIntent

ParseHelper

- TAG: final String
- TABLE_GROUPS: final String
- TABLE_USER: final String
- TABLE_ALARMS: final String
- TABLE_ALARMSTATUS: final String
- COLUMN_NAME: final String
- COLUMN_USERS: final String
- COLUMN_ALARMS: final String
- COLUMN_USERNAME: final String
- COLUMN_GROUP: final String
- COLUMN_ALARMSTATUS: final String
+ COLUMN_ID: final String
+ COLUMN_MESSAGE: final String
+ COLUMN_TIME: final String
+ COLUMN_STATUS: final String
+ COLUMN_MONDAY: final String
+ COLUMN_TUESDAY: final String
+ COLUMN_WEDNESDAY: final String
+ COLUMN_THURSDAY: final String
+ COLUMN_FRIDAY: final String
+ COLUMN_SATURDAY: final String
+ COLUMN_SUNDAY: final String
+ tmp: boolean[]
- instance: AlarmDB
+ COLUMN_SNOOZE: final String

+ createGroup(String): void
+ getGroupsForUser(): List<String>
+ addUserToGroup(String, String): List<String>
+ addNewAlarmToGroup(Alarm, String): void
+ getAlarmsFromGroup(String): List<Alarm>
+ getAllRemoteAlarmsForUser(): List<Alarm>
+ getUsersInGroup(String): List<String>
+ getGroupFromAlarm(Alarm): String
+ leaveGroup(String): void
+ editAlarm(Alarm): void
+ getUserFromString(String): ParseUser
+ getGroupFromString(String): ParseObject
- getAlarmFromParseObject(ParseObject, int): Alarm
- getParseObjectFromAlarm(Alarm): ParseObject
+ getAlarmStatusUserPerGroup(String, String): String
+ setMyAlarmStatusPerGroup(String, String): void

MainActivity
extends ActionBarActivity

- NEW_ALARM_CODE: final int
- EDIT_ALARM_CODE: final int
- REMOVE_ALARM_CODE: final int
- listView: ListView
- rowItems: List<Alarm>
- adapter: AlarmListViewAdapter
- runListUpdate: Runnable
- TAG: final String
- refreshNetAlarm: Runnable

onCreate(Bundle): void
+ onResume(): void
+ onCreateOptionsMenu(Menu): boolean
+ onOptionsItemSelected(MenuItem): boolean
# onActivityResult(int, int, Intent): void
+ onCreateContextMenu(ContextMenu, View,
ContextMenu.ContextMenuInfo): void
+ onContextItemSelected(MenuItem): boolean
- editAlarm(int): void

GroupActivity
extends ActionBarActivity

- progress: View
- listView: ListView
- rowItems: List<String>
- adapter: GroupListViewAdapter
- TAG: final String

onCreate(Bundle): void
+ onResume(): void
+ onCreateOptionsMenu(Menu): boolean
+ onOptionsItemSelected(MenuItem): boolean
+ addGroup(String): void
+ showProgress(boolean): void

ParseUpdate
implements Runnable

- groups: List<String>

+ run(): void

NewGroup
implements ParseUpdate

- groupName: String

+ setGroupName(String): void
+ run(): void

ViewHelper

+ showProgress(final boolean, final View,
Context): void

Application

+ onCreate(): void

AlarmScreenActivity
extends Activity

- TAG: final String
- STATUS_STOPPED: final String
- STATUS_SNOOZED: final String
- STATUS_RINGING: final String
- STATUS_OFF: final String
- WAKELOCK_TIMEOUT: final int
- lock: PowerManager.WakeLock
- mediaPlayer: MediaPlayer
- group: String
- isGroup: boolean

onCreate(Bundle): void
+ onResume(): void
# onPause(): void
+ onCreateOptionsMenu(Menu): boolean
+ onOptionsItemSelected(MenuItem): boolean

EditGroupActivity
extends ActionBarActivity

- TAG: final String
- NEW_ALARM_CODE: final int
- EDIT_ALARM_CODE: final int
- userListView: ListView
- alarmListView: ListView
- userProgress: View
- alarmProgress: View
- userItems: List<User>
- alarmItems: List<Alarm>
- alarmAdapter: AlarmListViewAdapter
- userAdapter: UserListViewAdapter
- alarms: List<Alarm>
- users: List<String>
- groupName: String
- setAlarms: SetAlarms

onCreate(Bundle): void
+ onResume(): void
+ onPause(): void
+ onCreateOptionsMenu(Menu): boolean
+ onOptionsItemSelected(MenuItem): boolean
+ addUser(String): void
+ onCreateContextMenu(menu, View,
ContextMenu.ContextMenuInfo)
+ onContextItemSelected(MenuItem): boolean
# onActivityResult(int, int, Intent): void
- showProgress(boolean): void
- sendNotificationToGroup(String): void

ParseUpdate
implements Runnable

+ run(): void

AlarmParseUpdate
extends ParseUpdate

alarm: Alarm

+ setAlarm(Alarm): void

NewAlarm
implements AlarmParseUpdate

+ run(): void

DeleteAlarm
implements AlarmParseUpdate

+ run(): void

EditAlarm
implements AlarmParseUpdate

+ run(): void

NewUser
implements ParseUpdate

- user: String

+ setUser(String): void
+run(): void

RemoveActivity
extends ActionBarActivity

- listView: ListView
- rowItems: List<Alarm>
- adapter: RemoveListAdapter
- TAG: final String

onCreate(Bundle): void
+ onCreateOptionsMenu(Menu): boolean
+ onOptionsItemSelected(MenuItem): boolean

LoginActivity
extends Activity
implements LoaderCallbacks<Cursor>

- TAG: final String
- mAuthTask: UserLoginTask
- mRegTask: UserRegistrationTask
- mEmailView: AutoCompleteTextView
- mProgressView: View
- mLoginFormView: View
- mMessageBox: TextView
- mEmailSignInButton: Button
- mEmailRegisterButton: Button

onCreate(Bundle): void
- populateAutoComplete(): void
+ attemptLogin(): void
- isEmailValid(String): boolean
- isPasswordValid(String): boolean
+ onCreateLoader(int,
Bundle): Loader<Cursor>
+ onLoadFinished(Loader<Cursor>,Cursor):
void
+ onLoaderReset(Loader<Cursor>,Cursor):
void
- addEmailsToAutoComplete(List<String>):
void

+ UserLoginTask
extends AsyncTask<Void, Void, Boolean>

- mEmail: final String
- mPassword: final String

doInBackground(Void... params): Boolean
onPostExecute(final Boolean): void
onCancelled(): void

+ UserRegistrationTask
extends AsyncTask<Void, Void, Boolean>

- mEmail: final String
- mPassword: final String
- newUser: ParseUser

doInBackground(Void... params): Boolean
onPostExecute(final Boolean): void
onCancelled(): void

EditAlarmActivity

- TAG: final String
- dayNames: CharSequence[]
- snoozeTimes: CharSequence[]
- DISABLE_SNOOZE: final String
- parentIntent: Intent
- newAlarm: Alarm
- resultBool: boolean[]
- resultInt: int
- alarmMessage: EditText
- alarmTime: TextView
- alarmDays: TextView
- snoozeInterval: TextView

onCreate(Bundle): void
+ onClick(View): void
- showTimePickerDialog(): void
- updateAlarmTextView(): void
- showSnoozeIntervalDialog(): void
- updateSnoozeIntervalTextView(): void
- showAlarmDaysDialog(): void
- updateAlarmDayTextView(): void
+ onCreateOptionsMenu(Menu): boolean
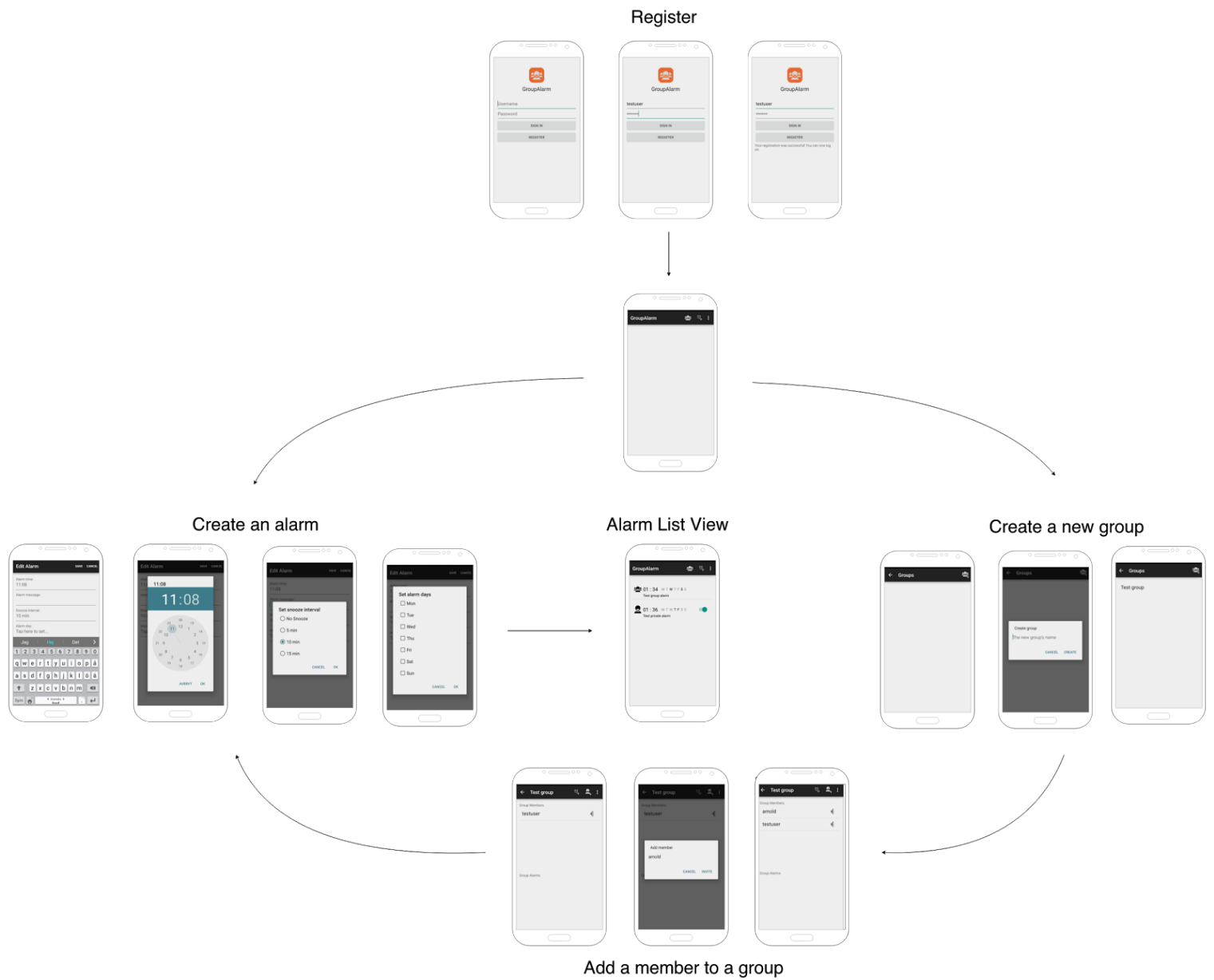+ onOptionsItemSelected(MenuItem): boolean

---

AlarmListViewAdapter

+ getView(int, View, ViewGroup): View
- alarmCheckedListener(final
Alarm): CompoundButton.OnCheckedChangeListener
- resolveToastMessage(Calendar): String

- TAG: final String
- snoozeAlarm: Alarm

ViewHolder

imageView: ImageView
time: TextView
eventDesc: TextView
monday: TextView
tuesday: TextView
wednesday: TextView
thursday: TextView
friday: TextView
saturday: TextView
sunday: TextView

GroupListViewAdapter

- context: Context

+ getView(int, View, ViewGroup): View

RemoveListViewAdapter

- TAG: final String
- context: Context
+ itemsToRemove: boolean []

+ getView(int, View, ViewGroup): View
- removeAlarmCheckedListener(final
int): CompoundButton.OnCheckedChangeListener
- reDrawUi(): void

ViewHolder

time: TextView
eventDesc: TextView
checkBox: CheckBox

UserListViewAdapter

- context: Context

+ getView(int, View, ViewGroup): View

5 Flowchart

Register

Create an alarm

Alarm List View

Create a new group

Add a member to a group

6 External Dependencies

GroupAlarm has some external dependencies which narrows the usage of the application.

- Android

  GroupAlarm can only function on android phones supporting API level 16 or more. This narrows the possible number of users.

- Internet

  The choice of using the Parse server for the group network makes the application dependent of Internet connection for group alarms. This narrows the geographical position of the users.

7 Sprint Planning

Feature tag	User story	Business value [1-10]	Development effort [1-50]
alarmclock	As a user I want to create a personal alarm.	1	50
alarmclock	As a user I want to be able to edit the alarms that I've created.	3	10
alarmclock	As a user I want to be able to activate and deactivate alarms in the alarm list view.	5	12
alarmclock	As a user I want to be able to view my alarms by time and day.	5	28
alarmclock	As a user I want to be able to stop my alarm so that it won't ring again.	5	4
alarmclock	As a user I want to be able to snooze my alarm so that I may continue sleeping for a few more minutes before it rings again.	7	6
alarmclock	As a user I want to be able to choose a snooze interval when I create or edit an alarm.	4	12
alarmclock	As a user I want to be able to choose time and weekday when I create or edit an alarm.	4	4
alarmclock	As a user I want to be able to set a message when I create or edit an alarm.	5	6
app	As a user I have to create a user account to use the application.	9	12
alarmclock	As a user I want to be able to remove alarms that are no longer relevant.	2	16
group alarm	As a group member I want to create and submit a group alarm so that it´s visible to my group members.	10	50
group alarm	As a user I want to be able to view all the alarms for a certain group I am in.	10	10
group alarm	As a group member I want to be able to change the alarm signal for a group member who is snoozing.	10	42
group alarm	As a group member I want to be able to edit an alarm for my group.	10	46
group alarm	As a group member I want to be able to delete an alarm for my group.	8	8
group	As a group member I want to get notified whenever a group member snoozes, stops or ignores his/hers alarm.	10	50
group	As a user I want to be able to create a group.	8	36
group	As a user I want to be able to invite other users to a group that I'm a member of.	9	36
group	As a user I want to be able to accept group invitations from other users so I can join groups.	10	32
group	As a group member I want to be able to leave a group that I'm a member of.	6	8
group	As a group member I want to be able to set different weekdays for the different alarms for my group.	10	16
group	As a user I want to be able to view the groups that I'm a member of.	8	20
group	As a user I want to be able to leave a group as the last remaining member so that the group is resolved.	5	12

Before working with sprints
Making sure that all group members have git, Android studio and gets to know the programs. Having a brainstorm session where all members participated to plan the structure of the app and creating user stories and planning the different parts of the project. Since our group members would have few possibilities to work at the same time due to different schedules this meeting was very important for the groups future work.

Sprint #1
Deadline: 29-04-15

The user stories handled this week.

User story	Business value [1-10]	Development effort [1-50]
As a user I want to create a personal alarm.	1	50
As a user I want to be able to edit the alarms that I've created.	3	10
As a user I want to be able to activate and deactivate alarms in the view that displays all of my alarms.	5	12
As a user I want to be able to view my alarms by time and day.	5	28

What to do:
Make an app in which the user can set an alarm that goes of at the chosen moment and give the user the possibility to navigate between the different views in the app. The user should also have the possibility of changing the alarm and remove it while it's no longer wanted.

How to do it:
Making an app that opens and gives the user a possibility to navigate in the Main View. Therefor we create a MainActivity class. We create an AlarmData class where the information for the alarm can be set. A test-class for testing the AlarmData class. Create a class enabling changes and editing of the alarm, EditAlarmActivity class. Also making a class for removing the alarm from the view, RemoveActivity class. To enable the view of different alarms we create CustomListViewAdapter.

Sprint #2

Deadline: 06-05-15

The user stories handled this week.

User story	Business value [1-10]	Development effort [1-50]
As a user I want to be able to set a message when I create or edit an alarm.	5	6
As a user I want to be able to snooze my alarm so that I may continue sleeping for a few more minutes before it rings again.	7	4
As a user I want to be able to stop my alarm so that it won't ring again.	5	4
As a user I have to create a user account to use the application.	9	12
As a user I want to be able to remove alarms that are no longer relevant.	2	16

What to do:

Enable for the user to set a message to an alarm. Testing the alarm clock so far and make it possible to store alarms in a database. Also enable the user to have a login to the app.

How to do it:

Editing settings in the AlarmActivity class. Enhancing the usage of the ListView to enable switch on and off and making the alarm message visible in the alarm list. Making a dummy login to the app, expandable for future network. Fix the missing functionalities in the alarm function revealed during the testing of the alarm. Write a AlarmDB, a database class to store alarms and a test class for the database.

Sprint #3

Deadline: 13-05-15

The user stories handled this week.

User story	Business value [1-10]	Development effort [1-50]
As a user I want to be able to choose a snooze interval when I create or edit an alarm.	4	12

As a user I want to be able to choose time and weekday when I create or edit an alarm.	4	2

What to do:

Make sure that the alarm functions as wanted and making the UI easier to understand and handle. Enable for the user to choose from different snooze intervals.

How to do it:

Ensure the and broadening database functionality to make a correct the RemoveActivity after test results. Also looking through the EditAlarmActivity a cleaning up unnecessary methods and fixing default-values to match current time. Adding pop-ups to ease the usage and change icons for more easily understood user interface.

Sprint #4

Deadline: 2015-05-20

The user stories handled this week.

User story	Business value [1-10]	Development effort [1-50]
As a group member I want to create and submit a group alarm so that it´s visible to my group members.	10	50
As a user I want to be able to create a group.	8	36
As a user I want to be able to view all the alarms for a certain group I am in.	10	10
As a group member I want to be able to edit an alarm for my group.	10	46
As a group member I want to be able to delete an alarm for my group.	8	8
As a user I want to be able to invite other users to a group that I'm a member of.	9	36
As a user I want to be able to accept group invitations from other users so I can join groups.	10	32
As a group member I want to be able to leave a group that I'm a member of.	6	8

What to do:

Make the group functionality work. Then adding so that we can view and delete our group alarms. Test all the UI functions.

How to do it:

Implement the Parse server, extend the database to include groups with the Parse Core product. Create the visual differences for the group functionalities. Make a group view and create dialogs for adding members to groups and adding context menus where needed.

Sprint #5

Deadline: 2015-05-27

The user stories handled this week.

User story	Business value [1-10]	Development effort [1-50]
As a group member I want to be able to set different weekdays for the different alarms for my group.	10	16
As a user I want to be able to view the groups that I'm a member of.	8	20
As a user I want to be able to leave a group as the last remaining member so that the group is resolved.	5	12
As a group member I want to get notified whenever a group member snoozes, stops or ignores his/hers alarm.	10	50
As a group member I want to be able to change the alarm signal for a group member who is snoozing.	10	42

What to do:

Create the last features and design for GroupAlarm application in general. Adding the functionality of changing the snooze signal for group members that snooze and enable notifications while a group member ignor, snooze or delete an alarm.

How to do it:

Fixing bugs and tests for the group functionalities. Implementing Parse push for the notifications within the application. Decide for suitable icons for all the different views and modes. Enable changing of other group members snooze signal and adding more possible alarms signal to the application. Implementing ParsHelper methods. Add navigation "up" as Android standard.

7.1 Future plans

In the future for the GroupAlarm application features like uploading your own alarm signals and pictures for your user will be possible, as well as choosing alarm signal for your personal alarms. There are also some far future plans such as a connection to Facebook events where an automatic alarm will be set for an event you are going to participate in and GPS connection.

8 Acceptance tests

Create an account

Test ID: T1
Test Case description: Create a user account with a username and a password.
Precondition: The login view
Test Steps:
Variant 1
1. Write a username
2. Write a password
3. Press "Register"
4. The notification "You registration was successful" will show up

Variant 2
1. Write a username
2. Write a password
3. Press "Register"
4. The notification "Oops! Something went wrong with you registrationl" will show up

Variant 3
1. Write a username
2. Write a password
3. Press "Register"
4. The notification "The password is too short" will show up

Expected result
Variant 1
● The registration was successful and it is possible sign

Variant 2
● Someone else is using the same username, it is not possible to sign in

Variant 3
● The password is to short, it is not possible to sign in

Related user story
● As a user I have to create a user account to use the application.

Create a personal alarm

Test ID: T02
Test Case description: Create an alarm by choosing time, days, message and snooze interval.
Precondition: Viewing the alarm list.
Test Steps:

Variation 1
1. Press the "Alarm +" button
2. The "Add alarm" view is shown
3. Input the following:
    a. Alarm time: "10.30"
    b. Alarm message: "Private Alarm"
    c. Snooze interval: "5 min"
    d. Alarm day: Mon, Wed
4. Press the "Save" button
5. The "Alarm list" view is shown

Variation 2
1. Press the "Alarm +" button
2. The "Add alarm" view is shown
3. Press the "Save" button
4. The "Alarm list" view is shown

Expected result

Variation 1
● A private alarm is created and added to the alarm list. The alarm time and alarm message is shown in the alarm list. An on/off switch is also shown in the list to control if the alarm is active or not. When the alarm is created the switch will be off but if the switch is turned on, the alarm will go off at 10.30 the next upcoming Monday or Wednesday with the wanted 5 minutes snooze interval.

Variation 2
● A private alarm is created and added to the alarm list. The alarm time is set on the local time + 1 minute. The alarm message will be empty. An on/off switch is also shown in the list to control if the alarm is active or not and the alarm will be brighter when the alarm is on. The days chosen will be marked in the alarm list view. When the alarm is created the switch will be off but if the switch is turned on, the alarm will go off at the alarm time the next upcoming day with 10 minutes snooze interval.

Related user story
● As a user I want to create a personal alarm.
● As a user I want to be able to choose a snooze interval when I create or edit an alarm.
● As a user I want to be able to choose time and weekday when I create or edit an alarm.
● As a user I want to be able to set a message when I create or edit an alarm.

Edit an existing private alarm

Test ID: T03
Test Case description: Edit a private alarm in the alarm list
Precondition: Viewing the alarm list.
Test Steps:

1. Press and hold on an alarm
2. A dialog with the choices "Edit" and "Delete" is shown
3. Press "Edit"
4. Edit the following
    a. Alarm time: "11.30"
    b. Alarm message: "Private Alarm number two"
    c. Snooze interval: "10 min"
    d. Alarm day: Mon, Fri
5. Click the "Save" button
6. The "Alarm list" view is shown

Expected result

● The edited alarm will go off at 11.30 at Monday and Friday with 10 minute snooze. In the alarm list the new alarm time with be showed as well as the new days and the alarm message "Private Alarm number two".

Related user story
● As a user I want to be able to edit the alarms that I've created.
● As a user I want to be able to choose a snooze interval when I create or edit an alarm.
● As a user I want to be able to choose time and weekday when I create or edit an alarm.
● As a user I want to be able to set a message when I create or edit an alarm.

Remove an existing private alarm

Test ID: T04.1
Test Case description: Remove an alarm from the alarm list
Precondition: Viewing the alarm list.
Test Steps:

1. Press and hold on an alarm set at 11.30
2. A dialog with the choices "Edit" and "Delete" is shown
3. Press "Delete"
4. The "Alarm list" view is shown

Expected result

● When the "Delete" button is clicked, the alarm at 11.30 will be removed and the user will be returned to the alarm list. The removed alarm will no longer be in the list.

Related user story
● As a user I want to be able to remove alarms that are no longer relevant.

Remove many private alarms at the same time

Test ID: T04.2
Test Case description: Remove many alarms at the same time from the alarm list
Precondition: Viewing the alarm list.
Test Steps:

1. Press the action bar menu
2. A dialog with the choices "Delete Alarm" and "Logout user" is shown
3. Press "Delete Alarm"
4. The "Remove Alarm" view is shown with the alarm list and one checkbox for each alarm
5. Check in the checkbox for the alarm set at 11.30
6. Press the "Delete" button
7. The "Alarm list" view is shown

Expected result

● When the "Delete" button is clicked, the alarm set at 11.30 with the checked checkboxes will be removed and return to the alarm list. The removed alarms will no longer be a part of the alarm list.

Related user story
● As a user I want to be able to remove alarms that are no longer relevant.

View existing groups the user are a part of

Test ID: T05
Test Case description: To be able to overview all the groups the user is a part of
Precondition: Viewing the alarm list.
Test Steps:

1. Press the "Group" icon
2. The "Remove Alarm" view is shown

Expected result

● All the groups will be shown in as list

Related user story
● As a user I want to be able to view the groups that I'm a member of.

Create a new group

Test ID: T06
Test Case description: To be able to create a new group
Precondition: Viewing the group list
Test Steps:

1. Press the "Group + " icon
2. The dialog box "Create group" is shown
3. Input the following
    a. Group name: Test group
4. Press "Create"
5. The group view is shown

Expected result

● A new group with the name "Test group" is created and will be shown in the list of groups in the group view

Related user story
● As a user I want to be able to create a group.

Invite a member to a group

Test ID: T07

Test Case description: Invite member to a group the user is a part of

Precondition: Viewing the group list

Test Steps:

Variant 1

1. Click on one of the groups in the group list
2. The view of the group clicked on is shown
3. Press the "Person +" icon
4. The dialog box "Add member" pops up
5. Input the following:
   a. Username: Testmember
6. Click "Invite"
7. The view of the group clicked on is shown

Variant 2

1. Click on the group "Testgroup" in the group list
2. The view of the "Testgroup" is shown
3. Press the "Person +" icon
4. The dialog box "Add member" pops up
5. Input the following:
   a. Username: NotRegisteredUser
6. Click "Invite"
7. The view of the group clicked on is shown

Expected result

Variant 1

● The member "Testmember" is added to the "Testgroup" and is shown in the "Testgroup" view in the list with the rest of the members of the group.

Variant 2

● The username "NotRegisteredUser" is not a registered user and therefore no user is added to the "Testgroup" and no user will be added in the group members list if the "Testgroup".

Related user story

● As a user I want to be able to invite other users to a group that I'm a member of.

Create a group alarm

Test ID: T08
Test Case description: Create a group alarm for a group you are a member of
Precondition: Viewing the group list
Test Steps:

Variant 1
1. Click on the group you would like to create a group alarm for
2. The view of the group clicked on is shown
3. Press the "Alarm +" button
4. The view for creating/editing alarms is shown
   a. Alarm time: "10.30"
   b. Alarm message: "GroupAlarm"
   c. Snooze interval: "5 min"
   d. Alarm day: Mon, Wed
5. Press "Save"
6. The "Group" view is shown

Variant 2
1. Click on the group you would like to create a group alarm for
2. The view of the group clicked on is shown
3. Press the "Alarm +" button
4. The view for creating/editing alarms is shown
   a. Alarm time: "10.30"
   b. Alarm message: "GroupAlarm"
   c. Snooze interval: "5 min"
   d. Alarm day: Mon, Wed
5. Press "Cancel"
6. The "Group" view is shown

Expected result
Variant 1
● An alarm with the alarm time "10.30", message "GroupAlarm" and days Monday and Wednesday is shown in the alarm list on the view of the specific group in the list of the alarms of the group. It will also be added in the alarm list with a group icon.

Variant 2
● No new alarm is added in the alarm list.

Related user story
● As a group member I want to create an alarm for my group.

Edit a group alarm

Test ID: T09
Test Case description: Edit a group alarm
Precondition: Viewing the specific alarm view
Test Steps:

1. Press and hold over the alarm that you would like to edit
2. A dialog with the choices "Edit" and "Delete" is shown
3. Press "Edit"
4. The "Edit Alarm" view is shown
5. Edit the following
   a. Alarm time: "12.30"
   b. Alarm message: "Group Alarm number two"
   c. Snooze interval: "10 min"
   d. Alarm day: Mon, Fri
6. Press "Save"
7. The specific group view is shown

Expected result

● The alarm will be edited. The time of the alarm will now be 12.30, message "Group Alarm number two", snooze interval 10 minutes and the alarm will go off the next Monday and Friday. The edited information will be shown in the "Testgroup" alarm list and in the "Testuser" alarm list.

Related user story
● As a group member I want to be able to edit an alarm for my group.

Delete a group alarm

Test ID: T10
Test Case description: Delete an alarm from the list in the specific group view
Precondition: Viewing the specific alarm view
Test Steps:

1. Press and hold over the alarm at 12.30
2. A dialog with the choices "Edit" and "Delete" is shown
3. Press "Delete"
4. The specific group view is shown

Expected result

● The alarm at 12.30 will no longer be in the list of alarms for the "Testgroup"

Related user story
● As a group member I want to be able to delete an alarm for my group.

Leave a group

Test ID: T11
Test Case description: A member leaves a group it is a member of
Precondition: Viewing the group list
Test Steps:

1. Click on the "Testgroup"
2. The view of the "Testgroup" is shown
3. Press the activity bar button
4. A dialog with the choice "Leave Group" is shown
5. Press "Leave Group"
6. The "Group" view is shown

Expected result

● The "Testgroup" will no longer be in the list if groups for the member. If you are the last member of the group, the "Testgroup" will be resolved.

Related user story
● As a group member I want to be able to leave a group that I'm a member of.
● As a user I want to be able to leave a group as the last remaining member so that the group is resolved.

Acceptance Test	User Story
T01 Create an account	As a user I have to create a user account to use the application.
T02 Create a personal alarm	As a user I want to create a personal alarm. As a user I want to be able to choose a snooze interval when I create or edit an alarm. As a user I want to be able to choose time and weekday when I create or edit an alarm. As a user I want to be able to set a message when I create or edit an alarm.
T03 Edit an existing alarm	As a user I want to be able to edit the alarms that I've created. As a user I want to be able to choose a snooze interval when I create or edit an alarm. As a user I want to be able to choose time and weekday when I create or edit an alarm. As a user I want to be able to set a message when I create or edit an alarm.
T04.1 Remove an existing private alarm T04.2 Remove many private alarms at the same time	As a user I want to be able to remove alarms that are no longer relevant.
T05 View existing groups the user is a part of	As a user I want to be able to view the groups that I'm a member of.
T06 Create a new group	As a user I want to be able to create a group.
T07 Invite a member to a group	As a user I want to be able to invite other users to a group that I'm a member of.
T08 Create a group alarm	As a group member I want to create an alarm for my group.
T09 Edit a group alarm	As a group member I want to be able to edit an alarm for my group.
T10 Delete a group alarm	As a group member I want to be able to delete an alarm for my group.
T11 Leave a group	As a group member I want to be able to leave a group that I'm a member of. As a user I want to be able to leave a group as the last remaining member so that the group is resolved.