



Projet 4 :

-Créez une application de gestion de réunions.



Lamzone

Contexte :

Cadrage du projet

- Liste des réunions comprenant :
 - l'heure de la réunion ;
 - le lieu de la réunion ;
 - le sujet de la réunion ;
 - la liste des participants (adresses mail).
- Ajout d'une réunion.
- Suppression d'une réunion.
- Filtre des réunions par date ou par lieu.
- Gestion de l'affichage responsive sur toutes les tailles de téléphone et de tablette Android, en modes portrait et paysage.
- Application supportant Android 5.0 (API21) et ses versions supérieures.

Phasage

- Phase 1 : Création de l'application comprenant toutes les fonctionnalités demandées.
- Phase 2 : Réalisation d'un test unitaire et d'un test instrumentalisé par fonctionnalité.

Spécificités

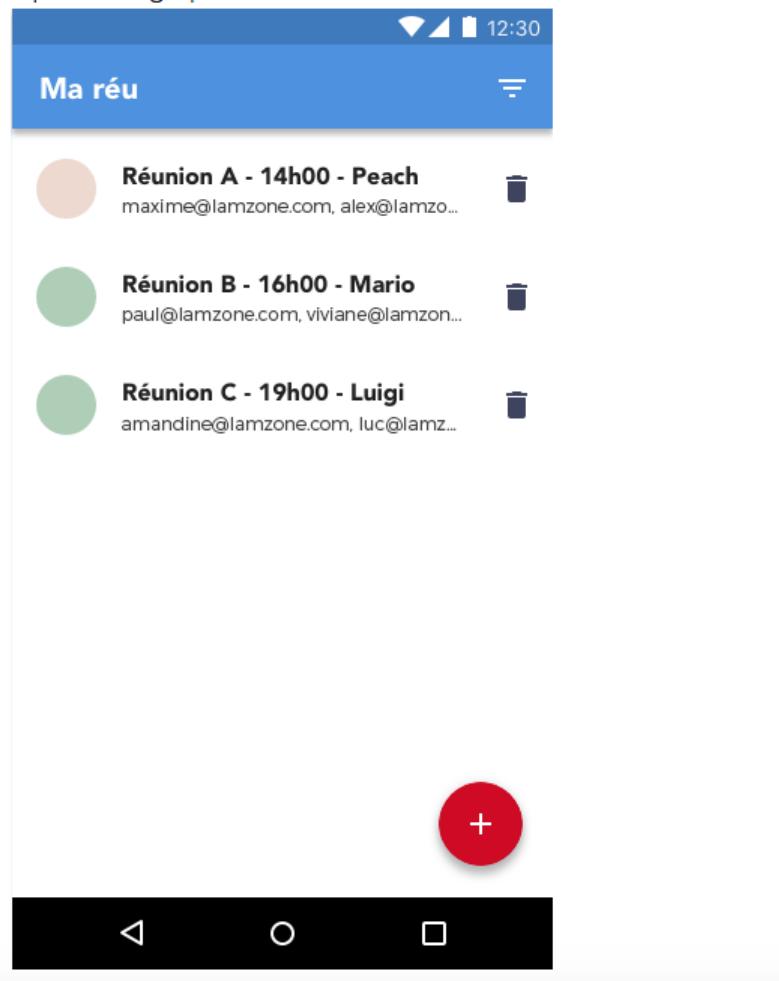
- L'application sera un POC, la persistance des données ne sera donc pas gérée.

Parcours Développeur Android



Contexte :

De son côté, Maxime, le graphiste de l'entreprise, vous fournit les premiers écrans de l'application. Il les a réalisés avec Sketch, un logiciel qui permet de faire des maquettes très utilisée parmi les graphistes.



Cossu Denis -Projet 4-

Parcours Développeur Android



Bonjour Francis,

Comme vous nous l'avez demandé voici le paragraphe qui justifie notre choix :

Le langage Java a des avantages par rapport à Kotlin: Il existe depuis plus de 20 ans donc il possède une large communauté pour aider en cas de problème et il y a plus d'experts en java qu'en Kotlin notamment dans notre entreprise. C'est un langage qui n'est pas difficile à apprendre, à comprendre, à exécuter, un nombre très important d'application Android ont été codé en Java. Il est open-source et il est régulièrement mis à jour par Oracle Corporation qui continue de l'améliorer. De plus Java est également flexible pour le développement d'application multiplate-formes (sa gestion pratique de la mémoire permet de rendre son utilisation portable sur une très grande gamme d'appareil) même si dans ce cas ce n'est pas ce qui nous intéresse particulièrement.

Cossu Denis -Projet 4-

Parcours Développeur Android



Kotlin quant à lui, même si il a ses avantages (code potentiellement plus court donc plus rapide à écrire, ressemble à Apple Swift...) a quand même ses inconvénients dont notamment le soucis de trouver des personnes qualifiées dans ce langage (récent donc petite communauté, nos équipes sont plus spécialisées en Java). Il me semble même que la compilation est parfois plus lente qu'en Java.

Nous aurions choisi Kotlin si nous avions un personnel qualifié dans ce langage, dans les frameworks et outils qui lui sont associés et si nous avions voulu accélérer la mise sur le marché de l'application. Cependant puisque ce n'est pas le cas nous préférions rester sur du Java pour coder cette application.

Cordialement,
Cossu Denis.

Cossu Denis -Projet 4-

Parcours Développeur Android

(Model) Reunion :

```
public class Reunion implements Serializable {  
  
    /** Identifier */  
    private Integer id;  
  
    private String subject;  
  
    private String imageUrl;  
  
    private String lieu;  
  
    private String hour;  
  
    private String dateAnnee;  
  
    private String dateMois;  
  
    private String dateJour;  
  
    private List<String> adressesMail;  
  
    public Reunion(Integer id, String subject,  
                  String imageUrl, String lieu, String hour  
                  , String dateAnnee, String dateMois,  
                  String dateJour, List<String> adressesMail) {  
  
        this.id = id;  
        this.subject = subject;  
        this.imageUrl = imageUrl;  
        this.lieu = lieu;  
        this.hour = hour;  
        this.dateAnnee = dateAnnee;  
        this.dateMois = dateMois;  
        this.dateJour = dateJour;  
        this.adressesMail = adressesMail;  
    }  
}
```

```
public Integer getId() { return id; }  
  
public void setId(Integer id) { this.id = id; }  
  
public String getSubject() { return subject; }  
  
public void setSubject(String subject) { this.subject = subject; }  
  
public String getImageUrl() { return imageUrl; }  
  
public String getLieu() { return lieu; }  
  
public String getHour() { return hour; }  
  
public String getDateAnnee() { return dateAnnee; }  
  
public String getDateMois() { return dateMois; }  
  
public String getDateJour() { return dateJour; }  
  
public List<String> getAdressesMail() { return adressesMail; }  
  
@Override  
public boolean equals(Object o) {  
    if (this == o) return true;  
    if (o == null || getClass() != o.getClass()) return false;  
    Reunion reunion = (Reunion) o;  
    return Objects.equals(id, reunion.id);  
}  
  
@Override  
public int hashCode() { return Objects.hash(id); }
```

Parcours Développeur Android

Reunion ApiService et DummyReunionGenerator :

```
/**  
 * Reunion API client  
 */  
public interface Reunion ApiService {  
  
    List<Reunion> getReunions();  
  
    void deleteReunion(Reunion reunion);  
  
    void addReunion(Reunion reunion);  
  
    void reunionTri(boolean salle);  
}
```

```
public abstract class DummyReunionGenerator {  
  
    public static List<Reunion> DUMMY_REUNION = Arrays.asList(  
        new Reunion( id: 1, subject: "Peach", imageUrl: "http://www.arthesis-diffusion.fr/fichiers_s  
            , lieu: "Réunion A", hour: "14h00", dateAnnee: "2019", dateMois: "01",  
            dateJour: "01",Arrays.asList("maxime@lamzone.com","alex@lamzone.com")),  
        new Reunion( id: 2, subject: "Mario", imageUrl: "http://www.materic.fr/images/realisations/s  
        new Reunion( id: 3, subject: "Daisy", imageUrl: "http://www.arthesis-diffusion.fr/fichiers_s  
        new Reunion( id: 4, subject: "Luigi", imageUrl: "http://affaire.terrabotanica.fr/wp-content/  
    );  
  
    static List<Reunion> generateReunions() { return new ArrayList<>(DUMMY_REUNION); }  
}
```

Cossu Denis -Projet 4-

Parcours Développeur Android



DummyReunion ApiService et DI :

```
public class DummyReunion ApiService implements Reunion ApiService {  
  
    private List<Reunion> reunions = DummyReunionGenerator.generateReunions();  
  
    @Override  
    public List<Reunion> getReunions() { return reunions; }  
  
    @Override  
    public void deleteReunion(Reunion reunion) { reunions.remove(reunion); }  
  
    @Override  
    public void addReunion(Reunion reunion) { reunions.add(reunion); }  
  
    @Override  
    public void reunionTri(boolean salle) {  
        Comparator<Reunion> comparator;  
  
        if (salle) {  
            comparator = Comparator.comparing(Reunion::getLieu);  
        } else {  
            comparator = Comparator.comparing(Reunion::getDateAnnee).  
                thenComparing(Reunion::getDateMois).thenComparing(Reunion::getDateJour).  
                thenComparing(Reunion::getHour);  
        }  
        Collections.sort(reunions, comparator);  
    }  
}
```

```
/**  
 * Dependency injector to get instance of services  
 */  
public class DI {  
  
    private static Reunion ApiService service = new DummyReunion ApiService();  
  
    public static Reunion ApiService getReunion ApiService() { return service; }  
  
    public static Reunion ApiService getNewInstance ApiService() {  
        return new DummyReunion ApiService();  
    }  
}
```

Parcours Développeur Android



Les events :

- AddReunionEvent
- DeleteReunionEvent
- TriReunionEvent

```
public class AddReunionEvent {  
  
    public Reunion reunion;  
  
    /**  
     * Constructor.  
     * @param reunion  
     */  
    public AddReunionEvent(Reunion reunion) {  
  
        this.reunion = reunion;  
    }  
}
```

```
public class DeleteReunionEvent {  
  
    /**  
     * Reunion to delete  
     */  
    public Reunion reunion;  
  
    /**  
     * Constructor.  
     * @param reunion  
     */  
    public DeleteReunionEvent(Reunion reunion) {  
  
        this.reunion = reunion;  
    }  
}
```

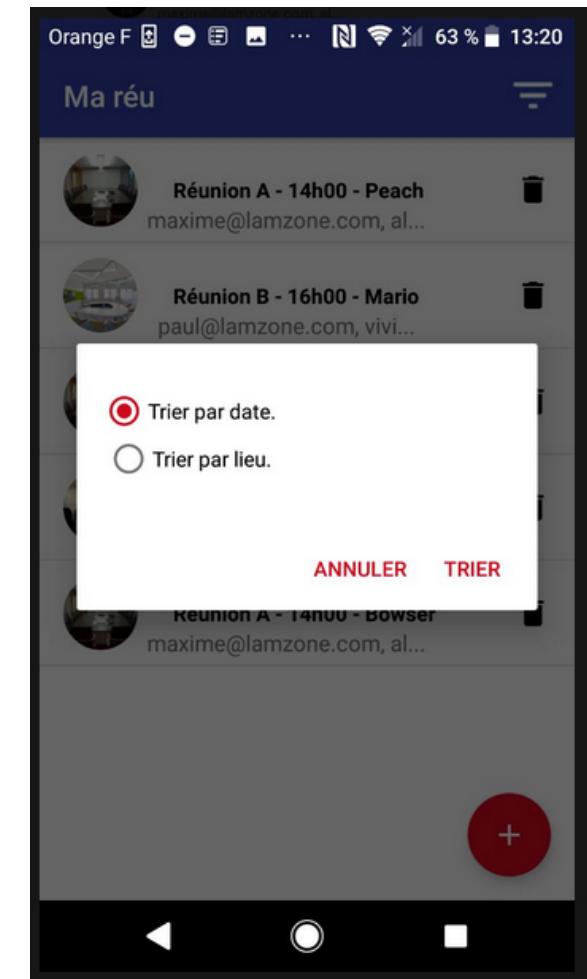
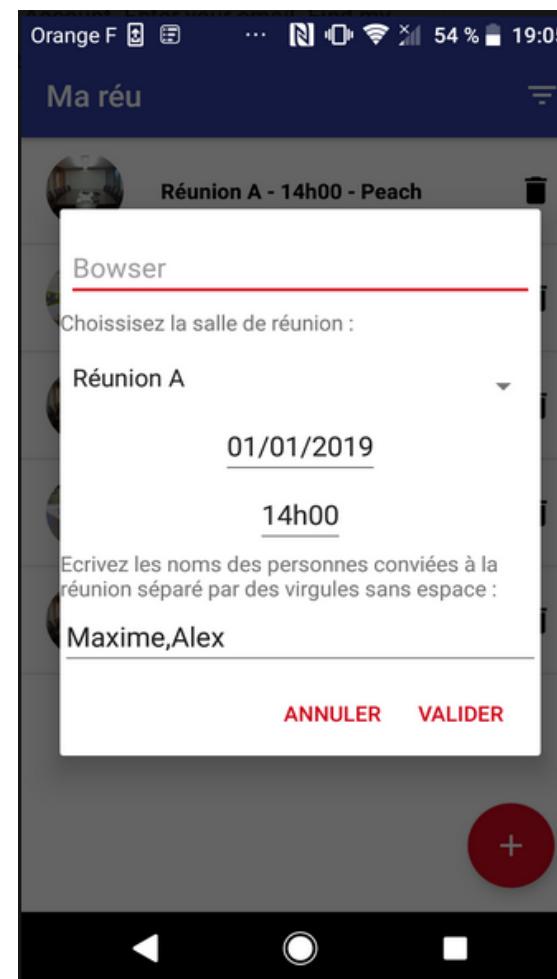
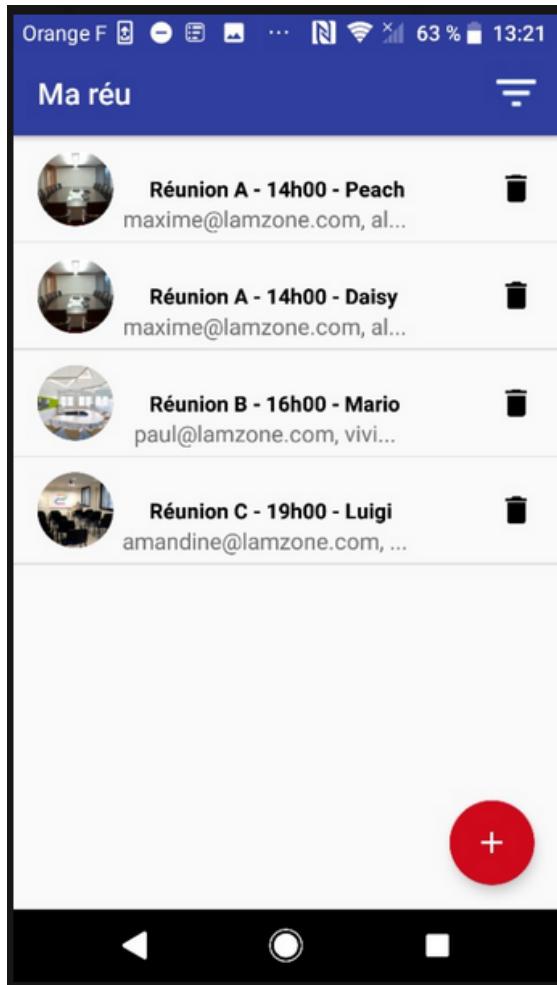
```
public class TriReunionEvent {  
  
    public boolean salle ;  
  
    public TriReunionEvent(boolean salle) {  
        this.salle = salle;  
    }  
}
```

Cossu Denis -Projet 4-

Parcours Développeur Android



MaReuActivity :



Cossu Denís -Projet 4-

Parcours Développeur Android



MaReuActivity :

```
public class MaReuActivity extends AppCompatActivity {

    // UI Components
    @BindView(R.id.container)
    ViewPager mViewPager;

    ListReunionPagerAdapter mPagerAdapter;
    FloatingActionButton fab;
    int id ;
    String subject;
    String image;
    String reunion;
    String heure;
    String dateAnnee;
    String dateMois;
    String dateJour;
    List<String> adressesMail;
    EditText dateText = null ;
    EditText hourText = null ;
    private Spinner dialogSpinner = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_ma_reu);
        ButterKnife.bind( target: this);

        mPagerAdapter = new ListReunionPagerAdapter(getSupportFragmentManager());
        mViewPager.setAdapter(mPagerAdapter);

        id = 4;
        configureFab();
    }
}
```

```
private void configureFab() {
    fab = findViewById(R.id.fab);

    fab.setOnClickListener(view -> {
        subject = "Bowser";
        id += 1;
        reunion = "Réunion A";
        image = "http://www.arthesis-diffusion.fr/fichiers_site/a1346art/contenu_pages/CIMG3247.JPG";
        heure = "14h00";
        dateAnnee = "2019";
        dateMois = "01";
        dateJour = "01";
        adressesMail = Arrays.asList("maxime@lamzone.com","alex@lamzone.com");

        new AlertDialog.Builder(view.getContext())
            .setView(R.layout.dialog_choix_sujet)
            .setPositiveButton( text: "Valider", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    configureSujet(dialog);
                    configureLieu(dialog);
                    configureDateHeure(dialog);
                    configureAdresse(dialog);

                    EventBus.getDefault().post(new AddReunionEvent(new Reunion(id,subject,
                        image,reunion,heure,dateAnnee,dateMois,dateJour, adressesMail)));
                }
            })
            .setNegativeButton( text: "Annuler", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                }
            })
            .show();
    });
}

private void configureSujet(DialogInterface dialog){
    EditText etSujet = (EditText) ((AlertDialog) dialog).findViewById(R.id.sujet);

    if (etSujet.getText().toString().length()>0) {
        subject = etSujet.getText().toString();
    }
}
```

Cossu Denis -Projet 4-

Parcours Développeur Android



MaReuActivity (suite) :

```
private void configureLieu(DialogInterface dialog){
    dialogSpinner = ((AlertDialog) dialog).findViewById(R.id.project_spinner);

    if (dialogSpinner.getSelectedItem() instanceof String) {
        reunion = (String) dialogSpinner.getSelectedItem();
    }

    switch (reunion) {
        case "Réunion A":
            image = "http://www.arthesis-diffusion.fr/fichiers_site/a1346art/co
            break;
        case "Réunion B":
            image = "http://www.materic.fr/images/realisations/salle-reunion-NS
            break;
        case "Réunion C":
            image = "http://affaire.terrabitica.fr/wp-content/uploads/2016/01
            break;
        case "Réunion D":
            image = "https://cdn.beewake.com/default/5a05/ce/thumb_5a05ce7372a2
            break;
        case "Réunion E":
            image = "http://www.lelavoir-ateliersreunis.fr/wp-content/uploads/2
            break;
        case "Réunion F":
            image = "https://blogbizmeeting.files.wordpress.com/2016/12/bagnole
            break;
        case "Réunion G":
            image = "http://www.sonovision.com/images/galleries/_2018_GALERIE/2
            break;
        case "Réunion H":
            image = "http://www.videlio.com/upload/images/videlio-iec/cas-clien
            break;
        case "Réunion I":
            image = "https://www.ahstatic.com/photos/0906_sm_00_p_2048x1536.jpg
            break;
        case "Réunion J":
            image = "https://www.comet-meetings.com/wp-content/uploads/2019/02/
            break;
    }
}

private void configureDateHeure(DialogInterface dialog){
    dateText = (EditText) ((AlertDialog) dialog).findViewById(R.id.dateText);

    if (dateText.getText().toString().length()>0) {
        dateAnnee = dateText.getText().toString().substring(6,10);
        dateMois = dateText.getText().toString().substring(3,5);
        dateJour = dateText.getText().toString().substring(0,2);
    }
}
```

```
hourText = (EditText) ((AlertDialog) dialog).findViewById(R.id.hourText);

if (hourText.getText().toString().length()>0) {
    heure = hourText.getText().toString();
}
}

private void configureAdresse(DialogInterface dialog){
    EditText textAdress = (EditText) ((AlertDialog) dialog).findViewById(R.id.textAdress);

    if (textAdress.getText().toString().length()>0) {
        List<String> strings = Arrays.asList(textAdress.getText().toString().split( regex: ","));
        List <String> Adresse = new ArrayList<String>();
        for(int i = 0 ; i < strings.size() ; i++) {
            Adresse.add( strings.get(i).toLowerCase() + "@lamzone.com");
        }
        adressesMail = Adresse;
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    switch (id) {
        case R.id.Tri:
            new AlertDialog.Builder( context: this)
                .setView(R.layout.dialog_tri)
                .setPositiveButton( text: "Trier", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        RadioButton triSalle = (RadioButton) ((AlertDialog) dialog)
                            .findViewById(R.id.triLieu);
                        boolean salle = false;

                        if (triSalle.isChecked()){
                            salle = true;
                        }
                        EventBus.getDefault().post(new TriReunionEvent(salle));
                    }
                })
                .setNegativeButton( text: "Annuler", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                    }
                })
                .show();
            return true;
    }
    return super.onOptionsItemSelected(item);
}
```

Parcours Développeur Android



MaReuActivity (suite) :

```
private void configureLieu(DialogInterface dialog){
    dialogSpinner = ((AlertDialog) dialog).findViewById(R.id.project_spinner);

    if (dialogSpinner.getSelectedItem() instanceof String) {
        reunion = (String) dialogSpinner.getSelectedItem();
    }

    switch (reunion) {
        case "Réunion A":
            image = "http://www.arthesis-diffusion.fr/fichiers_site/a1346art/co
            break;
        case "Réunion B":
            image = "http://www.materic.fr/images/realisations/salle-reunion-NS
            break;
        case "Réunion C":
            image = "http://affaire.terrabitica.fr/wp-content/uploads/2016/01
            break;
        case "Réunion D":
            image = "https://cdn.beewake.com/default/5a05/ce/thumb_5a05ce7372a2
            break;
        case "Réunion E":
            image = "http://www.lelavoir-ateliersreunis.fr/wp-content/uploads/2
            break;
        case "Réunion F":
            image = "https://blogbizmeeting.files.wordpress.com/2016/12/bagnole
            break;
        case "Réunion G":
            image = "http://www.sonovision.com/images/galleries/_2018_GALERIE/2
            break;
        case "Réunion H":
            image = "http://www.videlio.com/upload/images/videlio-iec/cas-clien
            break;
        case "Réunion I":
            image = "https://www.ahstatic.com/photos/0906_sm_00_p_2048x1536.jpg
            break;
        case "Réunion J":
            image = "https://www.comet-meetings.com/wp-content/uploads/2019/02/
            break;
    }
}

private void configureDateHeure(DialogInterface dialog){
    dateText = (EditText) ((AlertDialog) dialog).findViewById(R.id.dateText);

    if (dateText.getText().toString().length()>0) {
        dateAnnee = dateText.getText().toString().substring(6,10);
        dateMois = dateText.getText().toString().substring(3,5);
        dateJour = dateText.getText().toString().substring(0,2);
    }
}
```

```
hourText = (EditText) ((AlertDialog) dialog).findViewById(R.id.hourText);

if (hourText.getText().toString().length()>0) {
    heure = hourText.getText().toString();
}
}

private void configureAdresse(DialogInterface dialog){
    EditText textAdress = (EditText) ((AlertDialog) dialog).findViewById(R.id.textAdress);

    if (textAdress.getText().toString().length()>0) {
        List<String> strings = Arrays.asList(textAdress.getText().toString().split( regex: ","));
        List <String> Adresse = new ArrayList<String>();
        for(int i = 0 ; i < strings.size() ; i++) {
            Adresse.add( strings.get(i).toLowerCase() + "@lamzone.com");
        }
        adressesMail = Adresse;
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    switch (id) {
        case R.id.Tri:
            new AlertDialog.Builder( context: this)
                .setView(R.layout.dialog_tri)
                .setPositiveButton( text: "Trier", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        RadioButton triSalle = (RadioButton) ((AlertDialog) dialog)
                            .findViewById(R.id.triLieu);
                        boolean salle = false;

                        if (triSalle.isChecked()){
                            salle = true;
                        }
                        EventBus.getDefault().post(new TriReunionEvent(salle));
                    }
                })
                .setNegativeButton( text: "Annuler", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                    }
                })
                .show();
            return true;
    }
    return super.onOptionsItemSelected(item);
}
```

Parcours Développeur Android

ListReunionPagerAdapter (suite) :

```
public class ListReunionPagerAdapter extends FragmentPagerAdapter {  
  
    public ListReunionPagerAdapter(FragmentManager fm) { super(fm); }  
  
    /**  
     * getItem is called to instantiate the fragment for the given page.  
     * @param position  
     * @return  
     */  
    @Override  
    public Fragment getItem(int position) {  
  
        return ReunionFragment.newInstance();  
    }  
  
    /**  
     * get the number of pages  
     * @return  
     */  
    @Override  
    public int getCount() { return 1; }  
}
```

Cossu Denis -Projet 4-

Parcours Développeur Android

ReunionFragment :

```
public class ReunionFragment extends Fragment {

    private Reunion ApiService m ApiService;
    private List<Reunion> m Reunions;
    private RecyclerView m RecyclerView;

    /**
     * Create and return a new instance
     * @return {@link ReunionFragment}
     */
    public static ReunionFragment newInstance() {
        ReunionFragment fragment = new ReunionFragment();
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        m ApiService = DI.getReunion ApiService();
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_reunion_list, container,
                                     attachToRoot: false);
        Context context = view.getContext();
        m RecyclerView = (RecyclerView) view;
        m RecyclerView.setLayoutManager(new LinearLayoutManager(context));
        m RecyclerView.addItemDecoration(new DividerItemDecoration(getContext(),
                DividerItemDecoration.VERTICAL));
        initList();
        return view;
    }
}
```

```
* Init the List of Reunions
*/
private void initList() {
    m Reunions = m ApiService.getReunions();
    m RecyclerView.setAdapter(new MyReunionRecycler ViewAdapter(m Reunions));
}

@Override
public void onStart() {
    super.onStart();
    EventBus.getDefault().register( subscriber: this);
}

@Override
public void onStop() {
    super.onStop();
    EventBus.getDefault().unregister( subscriber: this);
}

/**
 * Fired if the user clicks on a delete button
 * @param event
 */

@Subscribe
public void onDeleteReunion(DeleteReunionEvent event) {
    m ApiService.deleteReunion(event.reunion);
    initList();
}

@Subscribe
public void onAddReunionEvent(AddReunionEvent event) {
    m ApiService.addReunion(event.reunion);
    initList();
}

@Subscribe
public void onTriReunionEvent(TriReunionEvent event) {
    m ApiService.reunionTri(event.salle);
    initList();
}
```

Parcours Développeur Android

MyReunionRecyclerViewAdapter :

```
public class MyReunionRecyclerViewAdapter extends RecyclerView.Adapter<MyReunionRecyclerViewAdapter.ViewHolder> {

    private final List<Reunion> mReunions;

    public MyReunionRecyclerViewAdapter(List<Reunion> items) {
        mReunions = items;
    }

    @Override
    public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext())
            .inflate(R.layout.fragment_reunion, parent, attachToRoot: false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(final ViewHolder holder, int position) {
        final Reunion reunion = mReunions.get(position);

        holder.mReunionName.setText(reunion.getLieu() + " - " + reunion.getHour() + " - " + reunion.getSubject());

        String adresses = reunion.getAdressesMail().get(0) + ", " + reunion.getAdressesMail().get(1);

        switch (holder.mReunionAvatar.getResources().getConfiguration().orientation) {
            case Configuration.ORIENTATION_PORTRAIT: adresses = adresses.substring(0, 22) + "..."; break;
            case Configuration.ORIENTATION_LANDSCAPE: adresses = adresses + "..."; break;
        }

        holder.mReunionAdresse.setText(adresses);
        Glide.with(holder.mReunionAvatar.getContext())
            .load(reunion.getImageUrl())
            .apply(RequestOptions.circleCropTransform())
            .into(holder.mReunionAvatar);

        holder.mDeleteButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                EventBus.getDefault().post(new DeleteReunionEvent(reunion));
            }
        });
    }

    @Override
    public int getItemCount() { return mReunions.size(); }

    public class ViewHolder extends RecyclerView.ViewHolder {
        @BindView(R.id.item_list_avatar)
        public ImageView mReunionAvatar;
        @BindView(R.id.item_list_name)
        public TextView mReunionName;
        @BindView(R.id.item_list_adresse)
        public TextView mReunionAdresse;
        @BindView(R.id.item_list_delete_button)
        public ImageButton mDeleteButton;
    }
}
```

```
holder.mReunionName.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent nextPage = new Intent(holder.mReunionAvatar.getContext(),
                                      DetailReunionActivity.class);
        nextPage.putExtra(REUNION, reunion );
        holder.mReunionAvatar.getContext().startActivity(nextPage);
    }
});

@Override
public int getItemCount() { return mReunions.size(); }

public class ViewHolder extends RecyclerView.ViewHolder {
    @BindView(R.id.item_list_avatar)
    public ImageView mReunionAvatar;
    @BindView(R.id.item_list_name)
    public TextView mReunionName;
    @BindView(R.id.item_list_adresse)
    public TextView mReunionAdresse;
    @BindView(R.id.item_list_delete_button)
    public ImageButton mDeleteButton;

    public ViewHolder(View view) {
        super(view);
        ButterKnife.bind(target: this, view);
    }
}
```

Parcours Développeur Android

DetailReunionActivity :

```
public class DetailReunionActivity extends AppCompatActivity {

    private TextView text = null;
    private TextView salle = null;
    private TextView presentation = null;
    private TextView dateEtHeure = null;
    private TextView adresseMailTextView = null;
    private ImageView imageImageView = null;
    private Button back;
    private String adresseMail = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail_reunion);

        text = (TextView) findViewById(R.id.text);
        salle = (TextView) findViewById(R.id.reunion);
        presentation = (TextView) findViewById(R.id.description);
        dateEtHeure = (TextView) findViewById(R.id.dateEtHeure);
        adresseMailTextView = (TextView) findViewById(R.id.adresseMail);
        imageImageView = (ImageView) findViewById(R.id.imageAvatar);

        Intent i = getIntent();
        Reunion reunion = (Reunion) i.getSerializableExtra(REUNION);

        for(int j = 0 ; j < reunion.getAdressesMail().size() ; j++) {
            adresseMail = adresseMail + reunion.getAdressesMail().get(j) + ", ";
        }

        configureBack();
    }
}
```

```
Glide.with( activity: this)
    .load(reunion.getImageUrl())
    .centerCrop()
    .into(imageImageView);

text.setText(reunion.getSubject());
salle.setText("Salle : " + reunion.getLieu());
presentation.setText("Nous allons aborder plus en détail les différentes étapes du projet sur "
    + reunion.getSubject());
dateEtHeure.setText("le " + reunion.getDateJour() + "/"
    + reunion.getDateMois() + "/" + reunion.getDateAnnee() + " à " + reunion.getHour());
adresseMailTextView.setText(adresseMail);
presentation.setMovementMethod(new ScrollingMovementMethod());

}

private void configureBack() {
    back = findViewById(R.id.back);

    back.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { finish(); }
    });
}
```

Parcours Développeur Android

Les tests unitaires :

```
/*
 * Unit test on Reunion service
 */
@RunWith(JUnit4.class)
public class ReunionUnitTest {

    private Reunion ApiService service;

    @Before
    public void setup() { service = DI.getNewInstance ApiService(); }

    @Test
    public void getReunionsWithSuccess() {
        List<Reunion> reunions = service.getReunions();
        List<Reunion> expectedReunions = DummyReunionGenerator.DUMMY_REUNION;
        assertThat(reunions, IsIterableContainingInAnyOrder
                .containsInAnyOrder(expectedReunions.toArray()));
    }

    @Test
    public void deleteReunionWithSuccess() {
        Reunion reunionToDelete = service.getReunions().get(0);
        service.deleteReunion(reunionToDelete);
        assertFalse(service.getReunions().contains(reunionToDelete));
    }

    @Test
    public void addReunion() {

        Reunion reunionToAdd = service.getReunions().get(0);
        service.addReunion(reunionToAdd);
        assertTrue(service.getReunions().contains(reunionToAdd));
    }
}
```

```
@Test
public void getReunionsTriSalleWithSuccess() {
    List<Reunion> reunions = service.getReunions();

    service.reunionTri( salle: true);

    List<Reunion> expectedReunions = Arrays.asList(
        new Reunion( id: 1, subject: "Peach", imageUrl: "http://www.ar
        new Reunion( id: 3, subject: "Daisy", imageUrl: "http://www.ar
        new Reunion( id: 2, subject: "Mario", imageUrl: "http://www.ma
        new Reunion( id: 4, subject: "Luigi", imageUrl: "http://affair
    );

    assertEquals(reunions, expectedReunions);
}

@Test
public void getReunionsTriDateWithSuccess() {
    List<Reunion> reunions = service.getReunions();

    service.reunionTri( salle: false);

    List<Reunion> expectedReunions = Arrays.asList(
        new Reunion( id: 1, subject: "Peach", imageUrl: "http://www.ar
        new Reunion( id: 2, subject: "Mario", imageUrl: "http://www.ma
        new Reunion( id: 4, subject: "Luigi", imageUrl: "http://affair
        new Reunion( id: 3, subject: "Daisy", imageUrl: "http://www.ar
    );

    assertEquals(reunions, expectedReunions);
}
```

Parcours Développeur Android

Les tests instrumentalisés de l'application : ViewActions et ViewAssertion

```
public class DeleteViewAction implements ViewAction {  
    @Override  
    public Matcher<View> getConstraints() { return null; }  
  
    @Override  
    public String getDescription() { return "Click on specific button"; }  
  
    @Override  
    public void perform(UiController uiController, View view) {  
        View button = view.findViewById(R.id.item_list_delete_button);  
        button.performClick();  
    }  
}
```

```
public class LaunchActivityAction implements ViewAction {  
    @Override  
    public Matcher<View> getConstraints() { return null; }  
  
    @Override  
    public String getDescription() { return "Click on specific button"; }  
  
    @Override  
    public void perform(UiController uiController, View view) {  
        View button = view.findViewById(R.id.item_list_name);  
        button.performClick();  
    }  
}
```

```
public class RecyclerViewItemCountAssertion implements ViewAssertion {  
    private final Matcher<Integer> matcher;  
  
    public static RecyclerViewItemCountAssertion withItemCount(int expectedCount) {  
        return withItemCountMatchers.is(expectedCount));  
    }  
  
    public static RecyclerViewItemCountAssertion withItemCount(Matcher<Integer> matcher) {  
        return new RecyclerViewItemCountAssertion(matcher);  
    }  
  
    private RecyclerViewItemCountAssertion(Matcher<Integer> matcher) { this.matcher = matcher; }  
  
    @Override  
    public void check(View view, NoMatchingViewException noViewFoundException) {  
        if (noViewFoundException != null) {  
            throw noViewFoundException;  
        }  
  
        RecyclerView recyclerView = (RecyclerView) view;  
        RecyclerView.Adapter adapter = recyclerView.getAdapter();  
        Assert.assertThat(adapter.getItemCount(), matcher);  
    }  
}
```

Cossu Denis -Projet 4-

Parcours Développeur Android



Les tests instrumentalisés :

```
@FixMethodOrder(MethodSorters.NAME_ASCENDING)
@RunWith(AndroidJUnit4.class)
public class ReunionListTest {

    // This is fixed
    private static int ITEMS_COUNT = 4;
    private MaReuActivity mActivity;

    @Rule
    public ActivityTestRule<MaReuActivity> mActivityRule =
        new ActivityTestRule(MaReuActivity.class);

    @Before
    public void setUp() {
        mActivity = mActivityRule.getActivity();
        assertThat(mActivity, notNullValue());
    }

    /**
     * We ensure that our recyclerview is displaying at least one item
     */
    @Test
    public void A_myReunionsList_shouldNotBeEmpty() {
        // First scroll to the position that needs to be matched and click on it
        onView(withId(R.id.list_reunions))
            .check(matches(hasMinimumChildCount( minChildCount: 1)));
    }

    @Test
    public void B_myReunionsList_LaunchDetailActivity() {
        Intents.init();

        onView(withId(R.id.list_reunions))
            .perform(RecyclerViewActions.actionOnItemAtPosition
                [ position: 1, new LaunchActivityAction()]);
        intended(hasComponent(DetailReunionActivity.class.getName()));
    }
}
```

```
@Test
public void C_myReunionList_DetailActivity_withGoodName() {

    UiDevice mDevice = UiDevice.getInstance(InstrumentationRegistry.getInstrumentation());

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 0, new LaunchActivityAction()));

    ViewInteraction textConfirmPeach = onView(withId(R.id.text));
    textConfirmPeach.check(matches(withText("Peach")));

    mDevice.pressBack();

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 1, new LaunchActivityAction()));

    ViewInteraction textConfirmMario = onView(withId(R.id.text));
    textConfirmMario.check(matches(withText("Mario")));

}

@Test
public void D_myReunionsList_triSalleItem() {

    onView(withContentDescription( text: "Trier"))
        .perform(click());
    onView(withText("Trier")).perform(click());

    UiDevice mDevice = UiDevice.getInstance(InstrumentationRegistry.getInstrumentation());

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 0, new LaunchActivityAction()));

    ViewInteraction textConfirmPeach = onView(withId(R.id.text));
    textConfirmPeach.check(matches(withText("Peach")));

    mDevice.pressBack();

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 1, new LaunchActivityAction()));

    ViewInteraction textConfirmMario = onView(withId(R.id.text));
    textConfirmMario.check(matches(withText("Mario")));

    mDevice.pressBack();
}
```

Parcours Développeur Android



Les tests instrumentalisés (suite) :

```
    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 3, new LaunchActivityAction()));

    ViewInteraction textConfirmDaisy = onView(withId(R.id.text));
    textConfirmDaisy.check(matches(withText("Daisy")));
}

@Test
public void E_myReunionsList_triDateItem() {

    onView(withContentDescription( text: "Trier"))
        .perform(click());

    onView(withId(R.id.triLieu))
        .perform(click());

    onView(withText("Trier")).perform(click());

    UiDevice mDevice = UiDevice.getInstance(InstrumentationRegistry.getInstrumentation());

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 0, new LaunchActivityAction()));

    ViewInteraction textConfirmPeach = onView(withId(R.id.text));
    textConfirmPeach.check(matches(withText("Peach")));

    mDevice.pressBack();

    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 1, new LaunchActivityAction()));

    ViewInteraction textConfirmDaisy = onView(withId(R.id.text));
    textConfirmDaisy.check(matches(withText("Daisy")));

    mDevice.pressBack();
}
```

Parcours Développeur Android



Les tests instrumentalisés (suite) :

```
onView(withId(R.id.list_reunions))
    .perform(RecyclerViewActions.actionOnItemAtPosition( position: 2, new LaunchActivityAction()));

ViewInteraction textConfirmMario = onView(withId(R.id.text));
textConfirmMario.check(matches(withText("Mario")));

mDevice.pressBack();

onView(withId(R.id.list_reunions))
    .perform(RecyclerViewActions.actionOnItemAtPosition( position: 3, new LaunchActivityAction()));

ViewInteraction textConfirmLuigi = onView(withId(R.id.text));
textConfirmLuigi.check(matches(withText("Luigi")));

}

/**
 * When we delete an item, the item is no more shown
 */
@Test
public void F_myReunionsList_deleteAction_shouldRemoveItem() {
    // Given : We remove the element at position 2
    onView(withId(R.id.list_reunions)).check(withItemCount(ITEMS_COUNT));
    // When perform a click on a delete icon
    onView(withId(R.id.list_reunions))
        .perform(RecyclerViewActions.actionOnItemAtPosition( position: 1, new DeleteViewAction()));
    // Then : the number of element is 3
    ITEMS_COUNT = ITEMS_COUNT - 1;
    onView(withId(R.id.list_reunions)).check(withItemCount(ITEMS_COUNT));
}

@Test
public void G_myReunionsList_addItem() {

    onView(withId(R.id.list_reunions)).check(withItemCount(ITEMS_COUNT));
    onView(withId(R.id.fab)).perform(click());
    onView(withText("Valider")).perform(click());
    ITEMS_COUNT = ITEMS_COUNT + 1;
    onView(withId(R.id.list_reunions)).check(withItemCount(ITEMS_COUNT));
}
```

Parcours Développeur Android



les 2 rapports d'exécution (unitaire et instrumentalisé) des tests finaux qui réussissent :

ReunionUnitTest: 5 total, 5 passed

```
&quot;/private/var/folders/mb/87l0xvrn54z1558tw56_6m500000gn/T/A  
Studio.app/Contents/jre/jdk/Contents/Home/bin/java&quot; -ea -Didea.  
Didea.launcher.bin.path=/private/var/folders/mb/87l0xvrn54z1558tw56.  
Studio.app/Contents/bin&quot; -Dfile.encoding=UTF-8 -classpath &qu  
/F88C4349-7E2B-488B-AA20-0B29E08A3532/d/Android Studio.app/C  
/T/AppTranslocation/F88C4349-7E2B-488B-AA20-0B29E08A3532/d/A  
/mb/87l0xvrn54z1558tw56_6m500000gn/T/AppTranslocation/F88C43  
rt.jar:/Users/deniscossu/Library/Android/sdk/platforms/android-28/data.
```

ReunionListTest: 7 total, 7 passed

com.alphakiwi.mareu.ReunionListTest

- A_myReunionsList_shouldNotBeEmpty
- B_myReunionsList_LaunchDetailActivity
- C_myReunionList_DetailActivity_withGoodName
- D_myReunionsList_triSalleItem
- E_myReunionsList_triDateItem
- F_myReunionsList_deleteAction_shouldRemoveItem
- G_myReunionsList_addItem

Parcours Développeur Android



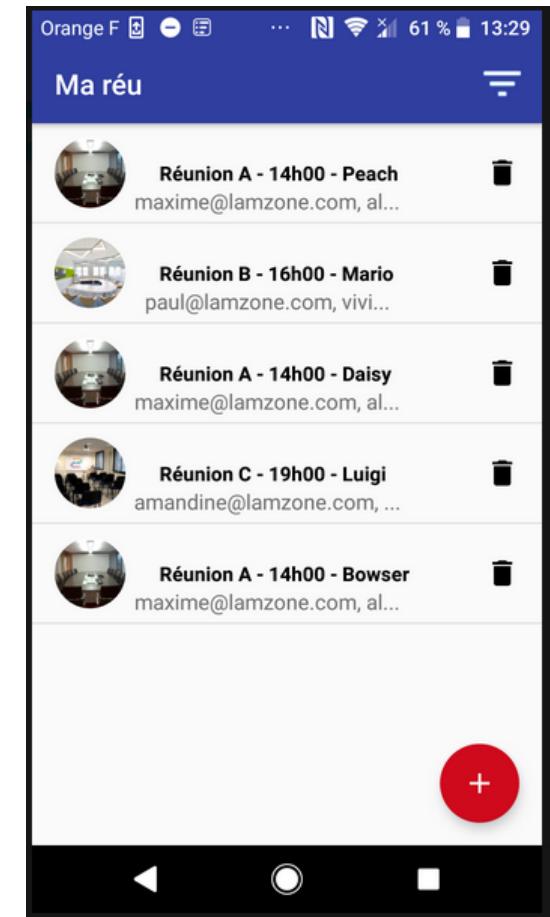
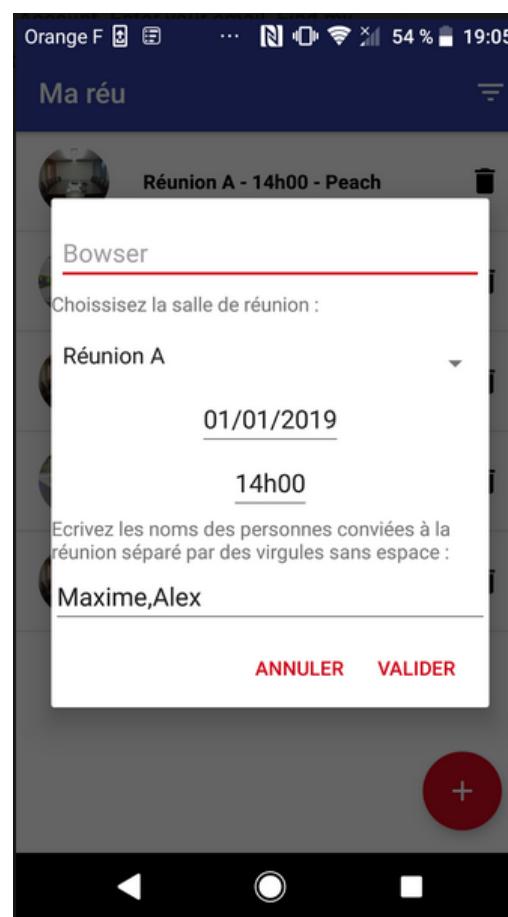
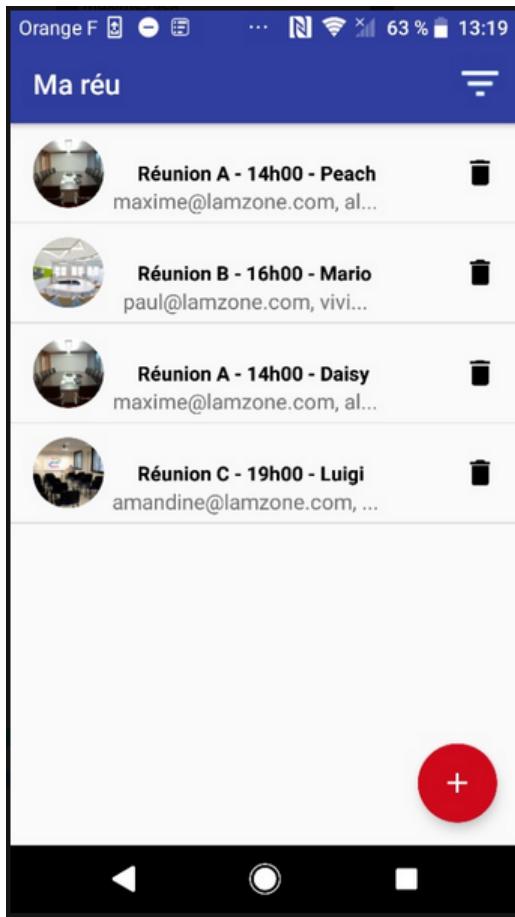
Un grand nombre de layouts différents pour que l'écran s'adapte aux différentes tailles de téléphones aussi bien mode portrait que paysage

```
▼  layout
    ▼  activity_detail_reunion (12)
        <> activity_detail_reunion.xml
        <> activity_detail_reunion.xml (land)
        <> activity_detail_reunion.xml (sw240dp)
        <> activity_detail_reunion.xml (sw240dp-land)
        <> activity_detail_reunion.xml (sw320dp)
        <> activity_detail_reunion.xml (sw320dp-land)
        <> activity_detail_reunion.xml (sw480dp)
        <> activity_detail_reunion.xml (sw480dp-land)
        <> activity_detail_reunion.xml (sw600dp)
        <> activity_detail_reunion.xml (sw600dp-land)
        <> activity_detail_reunion.xml (sw720dp)
        <> activity_detail_reunion.xml (sw720dp-land)
    ►  activity_ma_reu (12)
    ►  dialog_choix_sujet (12)
    ►  dialog_tri (12)
    ►  fragment_list_reunion (12)
    ►  fragment_reunion (12)
    ►  fragment_reunion_list (12)
```

Cossu Denis -Projet 4-

Parcours Développeur Android

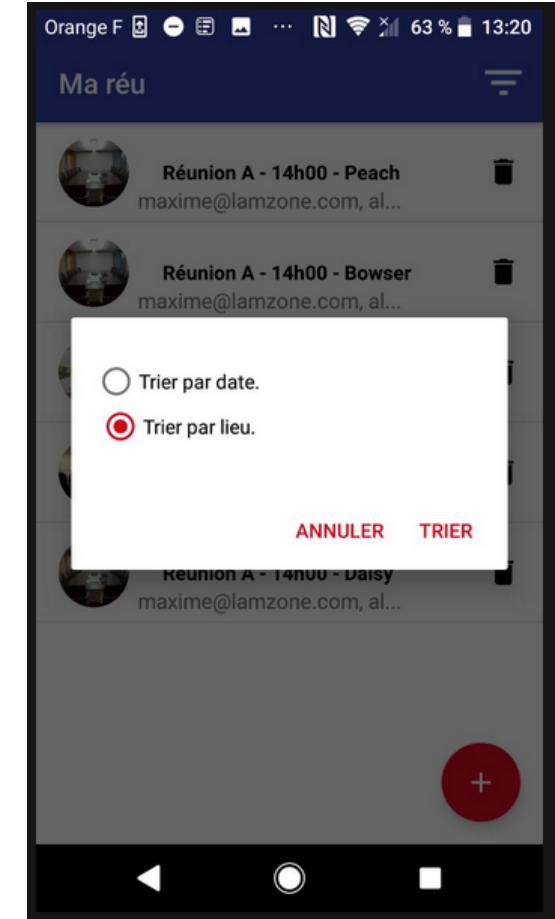
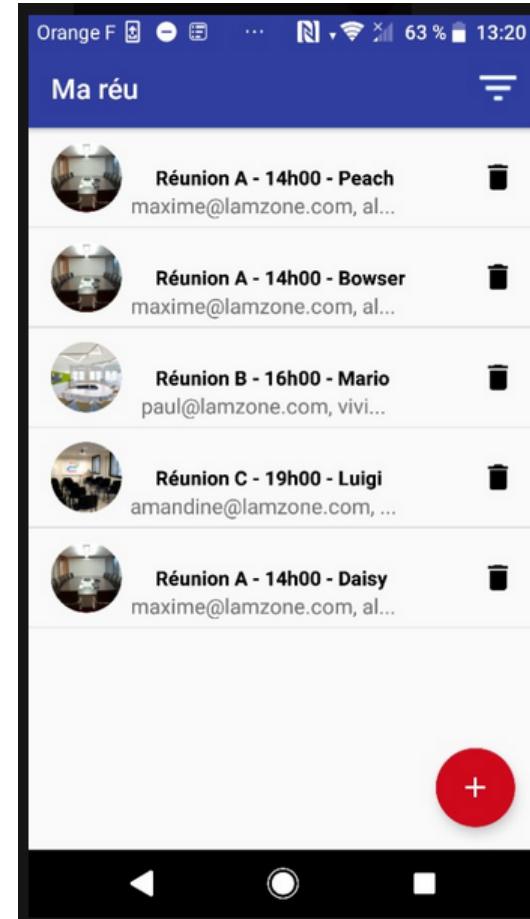
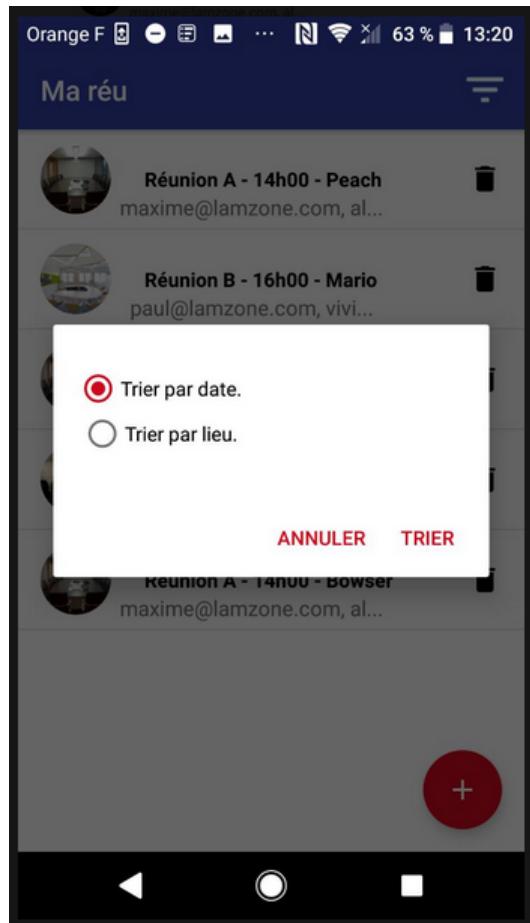
Conclusion :



Cossu Denís -Projet 4-

Parcours Développeur Android

Conclusion :

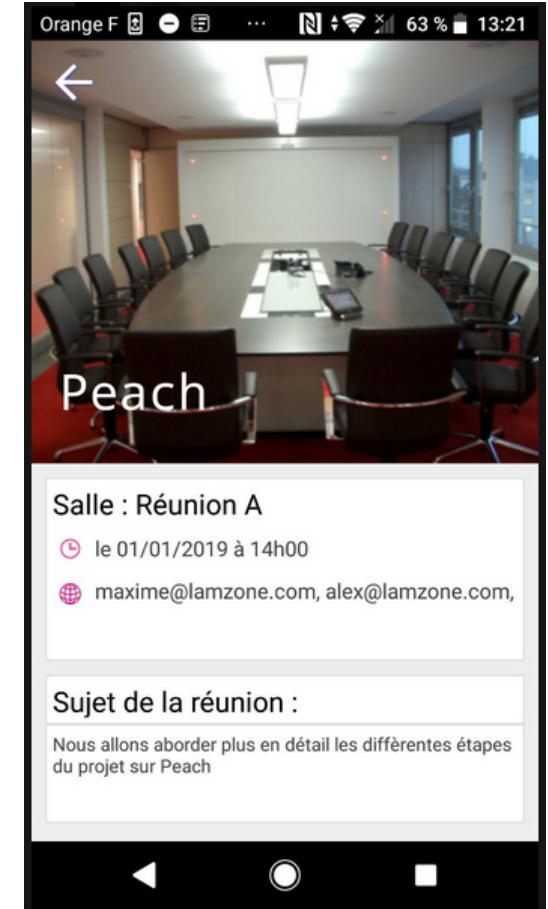
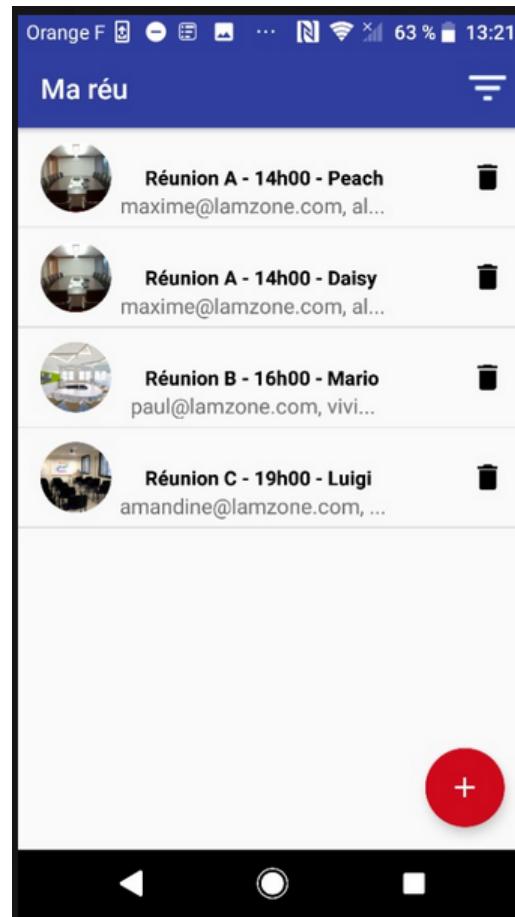
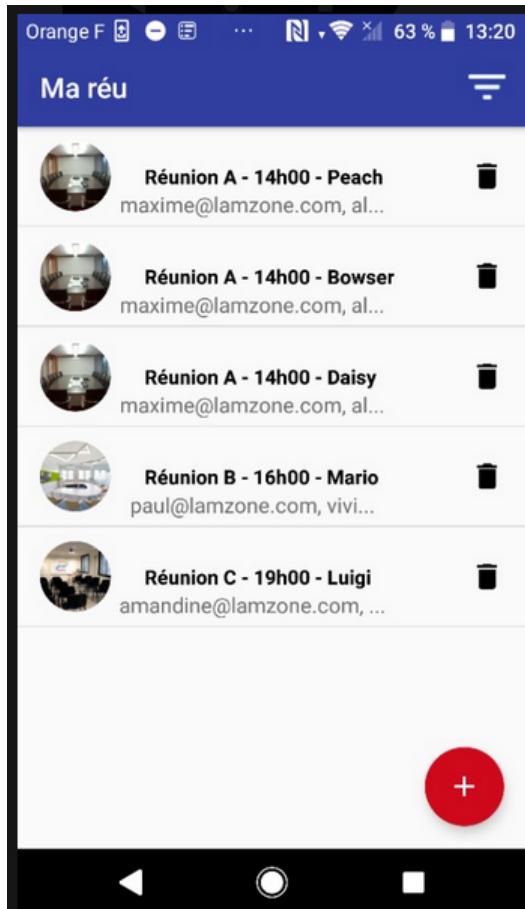


Cossu Denis -Projet 4-

Parcours Développeur Android



Conclusion :



Cossu Denis -Projet 4-

Parcours Développeur Android



Conclusion :

The image displays two screenshots of a mobile application interface. The top screenshot shows a list of meetings under the heading "Ma réu". It lists four meetings:

- Réunion A - 14h00 - Peach
maxime@lamzone.com, alex@lamzone.com...
- Réunion A - 14h00 - Daisy
maxime@lamzone.com, alex@lamzone.com...
- Réunion B - 16h00 - Mario
paul@lamzone.com, viviane@lamzone...
- Réunion C - 19h00 - Luigi
amandine@lamzone.com, alex@lamzon...

The bottom screenshot shows a detailed view of the first meeting, "Réunion A". It features a thumbnail image of a conference room with a large round table and several chairs. The name "Peach" is overlaid on the image. To the right, there is a sidebar with the following information:

Sujet de la réunion :
Nous allons aborder plus en détail les différentes étapes du projet sur Peach

Salle : Réunion A
🕒 le 01/01/2019 à 14h00
🌐 maxime@lamzone.com, alex@lamzone.com,

Cossu Denis -Projet 4-