Universidade de Aveiro
Mestrado em Cibersegurança
Código: 41794 - Engenharia Reversa
Responsible: João Paulo Silva Barraca

Android Reversing of OB Aplication

Monday 18th March, 2024
Ricardo Covelo (102668) - Telmo Sauce (104428)

Contents

1 Introduction

In this project, we were tasked to apply our classroom-acquired knowledge to analyze and identify the inner workings and potential vulnerabilities of an Android application called ”Oliveira do Bairro Municipality sports facilities”.

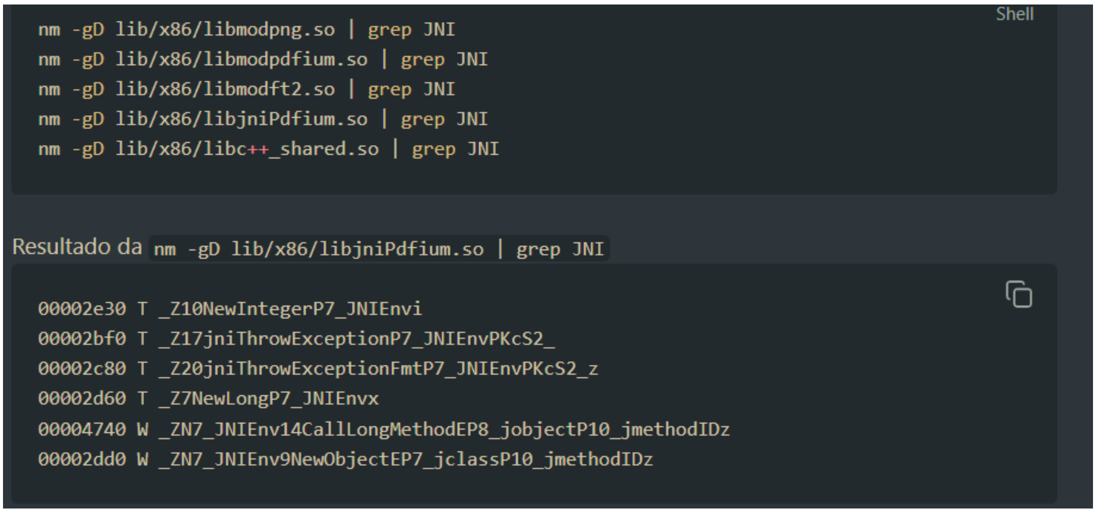
2 Static Analisys

2.1 Libraries

In this application, 5 libraries were found:

- libmodpng.so
- libmodpdfium.so
- libmodft2.so
- libjniPdfium.so
- libc++_shared.so

After conducting a short analysis, we determined that the only library utilized in the project was libjniPdfium, an open-source library designed for PDF file manipulation and rendering, and the other libraries were only for supporting this one. We came to this conclusion after analyzing all the files and checking for signs of dynamic and static linking processes.

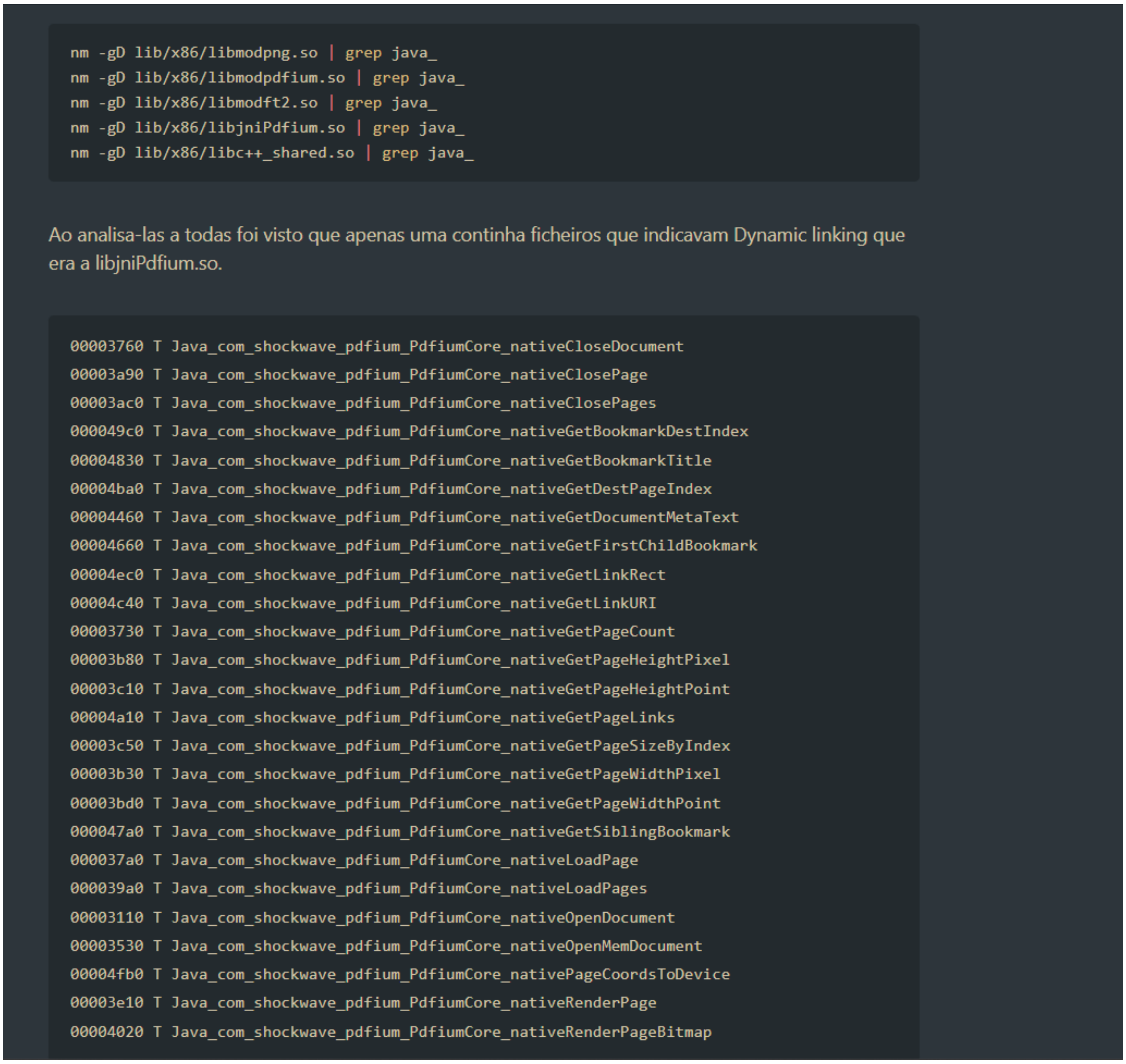


```
nm -gD lib/x86/libmodpng.so | grep JNI
nm -gD lib/x86/libmodpdfium.so | grep JNI
nm -gD lib/x86/libmodft2.so | grep JNI
nm -gD lib/x86/libjniPdfium.so | grep JNI
nm -gD lib/x86/libc++_shared.so | grep JNI
```

Resultado da `nm -gD lib/x86/libjniPdfium.so | grep JNI`

```
00002e30 T _Z10NewIntegerP7_JNIEnvi
00002bf0 T _Z17jniThrowExceptionP7_JNIEnvPKcS2_
00002c80 T _Z20jniThrowExceptionFmtP7_JNIEnvPKcS2_z
00002d60 T _Z7NewLongP7_JNIEnvx
00004740 W _ZN7_JNIEnv14CallLongMethodEP8_jobjectP10_jmethodIDz
00002dd0 W _ZN7_JNIEnv9NewObjectEP7_jclassP10_jmethodIDz
```

Figure 1: No JNI_OnLoad file on any file



```
nm -gD lib/x86/libmodpng.so | grep java_
nm -gD lib/x86/libmodpdfium.so | grep java_
nm -gD lib/x86/libmodft2.so | grep java_
nm -gD lib/x86/libjniPdfium.so | grep java_
nm -gD lib/x86/libc++_shared.so | grep java_
```

Ao analisa-las a todas foi visto que apenas uma continha ficheiros que indicavam Dynamic linking que era a libjniPdfium.so.

```
00003760 T Java_com_shockwave_pdfium_PdfiumCore_nativeCloseDocument
00003a90 T Java_com_shockwave_pdfium_PdfiumCore_nativeClosePage
00003ac0 T Java_com_shockwave_pdfium_PdfiumCore_nativeClosePages
000049c0 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetBookmarkDestIndex
00004830 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetBookmarkTitle
00004ba0 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetDestPageIndex
00004460 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetDocumentMetaText
00004660 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetFirstChildBookmark
00004ec0 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetLinkRect
00004c40 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetLinkURI
00003730 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageCount
00003b80 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageHeightPixel
00003c10 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageHeightPoint
00004a10 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageLinks
00003c50 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageSizeByIndex
00003b30 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageWidthPixel
00003bd0 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetPageWidthPoint
000047a0 T Java_com_shockwave_pdfium_PdfiumCore_nativeGetSiblingBookmark
000037a0 T Java_com_shockwave_pdfium_PdfiumCore_nativeLoadPage
000039a0 T Java_com_shockwave_pdfium_PdfiumCore_nativeLoadPages
00003110 T Java_com_shockwave_pdfium_PdfiumCore_nativeOpenDocument
00003530 T Java_com_shockwave_pdfium_PdfiumCore_nativeOpenMemDocument
00004fb0 T Java_com_shockwave_pdfium_PdfiumCore_nativePageCoordsToDevice
00003e10 T Java_com_shockwave_pdfium_PdfiumCore_nativeRenderPage
00004020 T Java_com_shockwave_pdfium_PdfiumCore_nativeRenderPageBitmap
```

Figure 2: Java_ file format just appears on libjniPdfium

2.2 Versions and Updates

After using the command ”apktool d app.apk” and navigating to the folder ”<appfolder>/unknown” we can find all third-party services used and their respective versions. This project is using Firebase, Google Play, and Transport.

Firebase

Service	Current Version	New Version	CVE
firebase-analytics_client	21.0.0	21.5.1	
firebase-annotations_client	16.0.0	16.2.0	
firebase-auth-interop_client	20.0.0		
firebase-auth_client	21.0.6	22.3.1	CVE-2022-2390
firebase-common_client	20.1.1	20.4.2	
firebase-components_client	17.0.0	17.1.5	
firebase-core_client	21.0.0	21.1.1	
firebase-crashlytics_client	18.2.11	18.6.2	
firebase-datatransport_client	18.1.5	18.2.1	
firebase-encoders-json_client	18.0.0	18.0.1	
firebase-encoders-proto_client	16.0.0		
firebase-encoders_client	17.0.0		
firebase-iid-interop_client	17.1.0		
firebase-installations-interop_client	17.0.1		
firebase-installations_client	17.0.1	17.2.0	
firebase-measurement-connector_client	19.0.0	20.0.1	CVE-2022-2390
firebase-messaging_client	23.0.6	23.4.1	CVE-2022-2390

Google Play

Service	Current Version	New Version	CVE
play-services-ads-identifier_client	18.0.0	18.0.1	CVE-2022-2390
play-services-auth-api-phone_client	17.4.0	18.0.2	CVE-2022-2390
play-services-base_client	18.0.1	18.3.0	CVE-2022-2390
play-services-basement_client	18.0.0	18.3.0	CVE-2022-2390
play-services-cloud-messaging_client	17.0.1	17.1.0	CVE-2022-2390
play-services-measurement-api_client	21.0.0	21.5.1	CVE-2022-2390
play-services-measurement-base_client	21.0.0		
play-services-measurement-impl_client	21.0.0	21.5.1	CVE-2022-2390
play-services-measurement-sdk-api_client	21.0.0		
play-services-measurement-sdk_client	21.0.0		
play-services-measurement_client	21.0.0		
play-services-safetynet_client	17.0.0		
play-services-stats_client	17.0.2	17.0.3	CVE-2022-2390
play-services-tasks_client	18.0.1	18.1.0	CVE-2022-2390

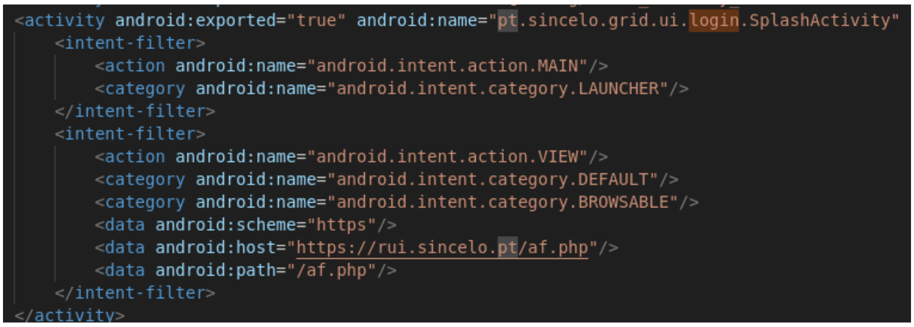
Transport

Service	Current Version	New Version
transport-api_client	3.0.0	
transport-backend-cct_client	3.1.6	3.2.0
transport-runtime_client	3.1.6	3.2.0

2.3 Android Manifest

Looking into the **Android Manifest** file we can see the permissions and the structure of the app.

Analyzing the Manifest reveals that the application starts with an authentication process. The layout for this initial screen is defined in ”include_splash_login.xml” and is handled by the ”SplashActivity” class.



```
<activity android:exported="true" android:name="pt.sincelo.grid.ui.login.SplashActivity"
    <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
    </intent-filter>
    <intent-filter>
        <action android:name="android.intent.action.VIEW"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <category android:name="android.intent.category.BROWSABLE"/>
        <data android:scheme="https"/>
        <data android:host="https://rui.sincelo.pt/af.php"/>
        <data android:path="/af.php"/>
    </intent-filter>
</activity>
```

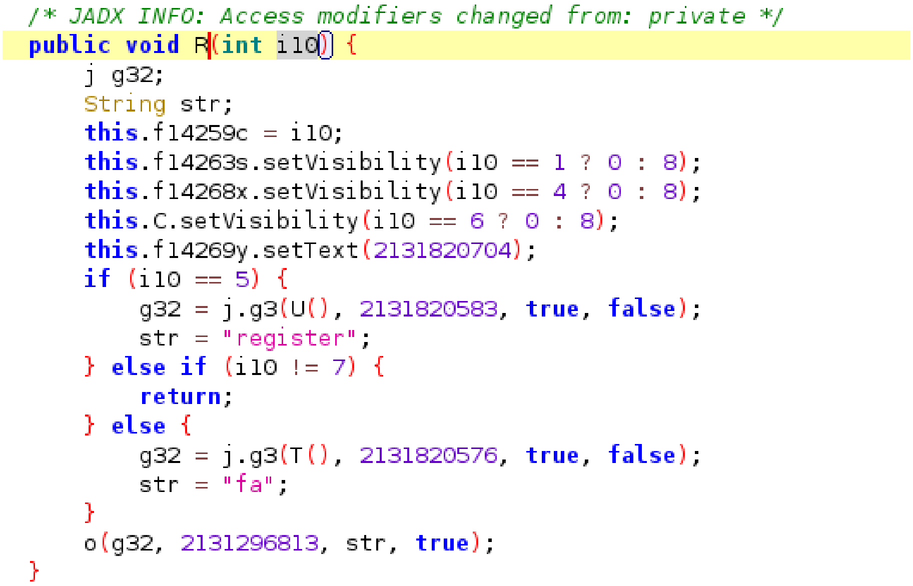
Figure 3: Main Activity

2.4 Authentication

During the static analisys we found functions on the file SplashActivity which make the authentication possible.

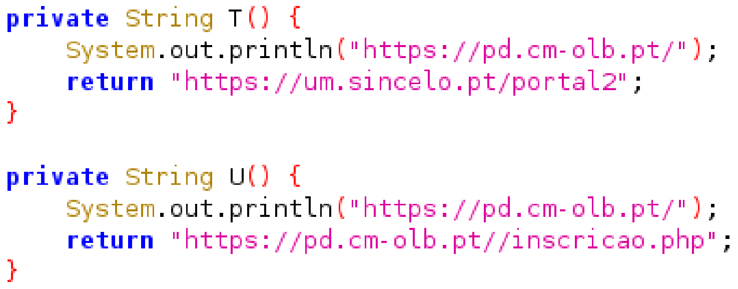
The function R filters register.php and loads it into a fragment, calling another function U within this process. Interestingly, R also attempts to load ”/portal2” which redirects to a different login page which we didn’t found being used on the application.

The login functionality seems to be handled separately using the include_splash_login.xml layout file. However, both user sign-up and login validation take place outside the application.



```java
/* JADX INFO: Access modifiers changed from: private */
public void R(int i10) {
    j g32;
    String str;
    this.f14259c = i10;
    this.f14263s.setVisibility(i10 == 1 ? 0 : 8);
    this.f14268x.setVisibility(i10 == 4 ? 0 : 8);
    this.C.setVisibility(i10 == 6 ? 0 : 8);
    this.f14269y.setText(2131820704);
    if (i10 == 5) {
        g32 = j.g3(U(), 2131820583, true, false);
        str = "register";
    } else if (i10 != 7) {
        return;
    } else {
        g32 = j.g3(T(), 2131820576, true, false);
        str = "fa";
    }
    o(g32, 2131296813, str, true);
}
```

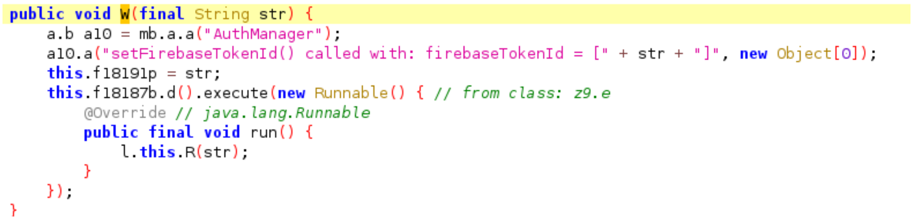
Figure 4: function R



```java
private String T() {
    System.out.println("https://pd.cm-olb.pt/");
    return "https://um.sincelo.pt/portal2";
}

private String U() {
    System.out.println("https://pd.cm-olb.pt/");
    return "https://pd.cm-olb.pt//inscricao.php";
}
```

Figure 5: Functions U and T

It was also found a function to make the connection with Firebase, which was later discovered using dynamic analysis that is called at the end of the authentication process.



```java
public void W(final String str) {
    a.b a10 = mb.a.a("AuthManager");
    a10.a("setFirebaseTokenId() called with: firebaseTokenId = [" + str + "]", new Object[0]);
    this.f18191p = str;
    this.f18187b.d().execute(new Runnable() { // from class: z9.e
        @Override // java.lang.Runnable
        public final void run() {
            l.this.R(str);
        }
    });
}
```

Figure 6: Firebase token function

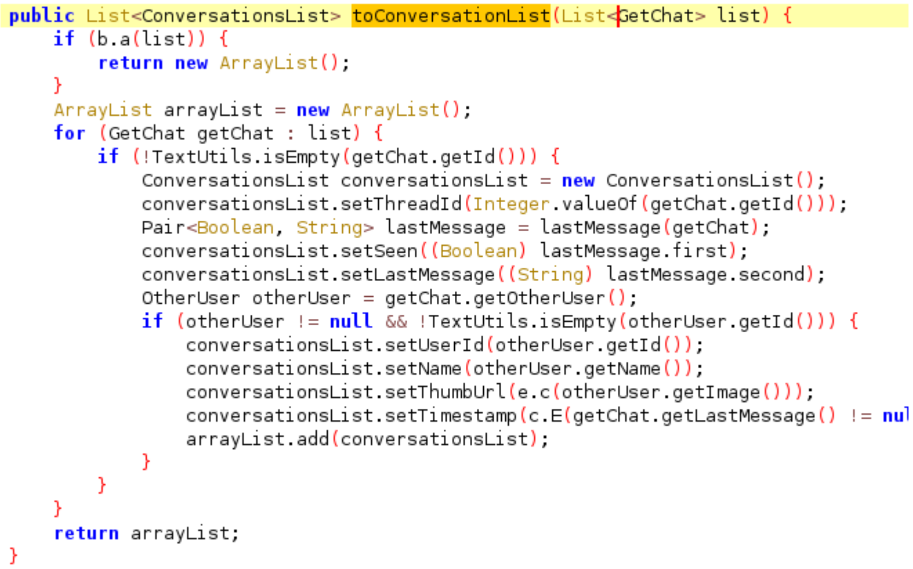
2.5 Communication

During our investigation, we identified a folder located at **”pt/sincelo/grid/data/model/messages”**. This folder contains core classes essential for enabling communication within the app. These classes play a crucial role in organizing chat data and messages.

- **OtherUser** - This class has information about the user to which the user is talking.
- **NewMessagesCount** - This is used as a counter for the number of messages the user hasn't seen.
- **Message** - Here is stored the information of the message such as the date, the user who sent the message, and its id. This doesn't include the last message of the Chat.
- **LastMessage** - This contains the information of the last message of the Chat.
- **Chat** - This class shows the chat that a user is having with another-
- **getChat**: This class is used to get the general information of a chat to show the user.

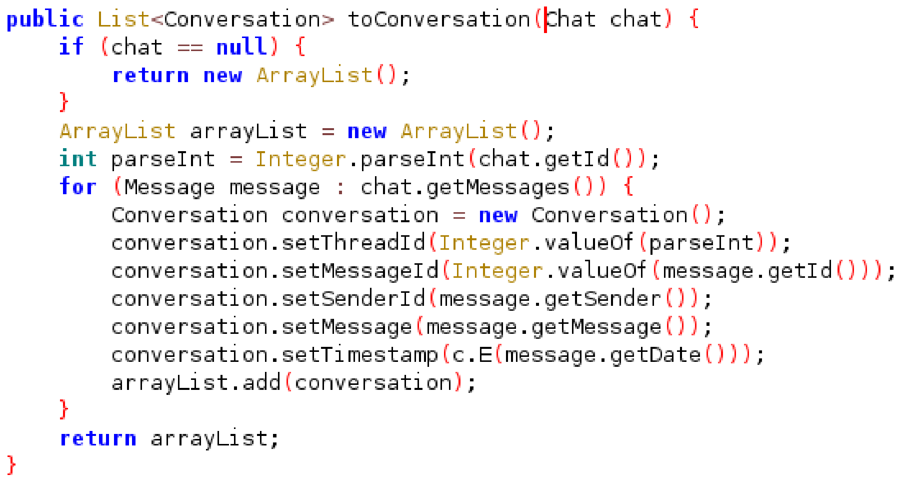
We identified two classes, **Conversation** and **ConversationList**, whose purposes are unclear.

However, some functions convert Chat to Conversation and getChat to ConversationList. This suggests these classes might serve a specific role in processing chat data, which at the moment is unclear to us.



```
public List<ConversationsList> toConversationList(List<GetChat> list) {
    if (b.a(list)) {
        return new ArrayList();
    }
    ArrayList arrayList = new ArrayList();
    for (GetChat getChat : list) {
        if (!TextUtils.isEmpty(getChat.getId())) {
            ConversationsList conversationsList = new ConversationsList();
            conversationsList.setThreadId(Integer.valueOf(getChat.getId()));
            Pair<Boolean, String> lastMessage = lastMessage(getChat);
            conversationsList.setSeen((Boolean) lastMessage.first);
            conversationsList.setLastMessage((String) lastMessage.second);
            OtherUser otherUser = getChat.getOtherUser();
            if (otherUser != null && !TextUtils.isEmpty(otherUser.getId())) {
                conversationsList.setUserId(otherUser.getId());
                conversationsList.setName(otherUser.getName());
                conversationsList.setThumbUrl(e.c(otherUser.getImage()));
                conversationsList.setTimestamp(c.E(getChat.getLastMessage() != nul
                arrayList.add(conversationsList);
            }
        }
    }
    return arrayList;
}
```

Figure 7: ToConversationList



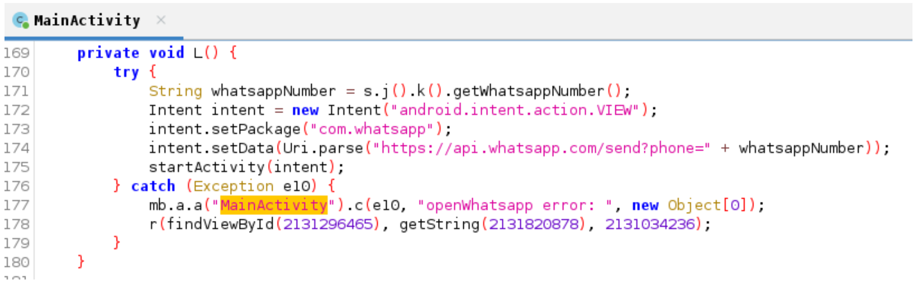
```
public List<Conversation> toConversation(Chat chat) {
    if (chat == null) {
        return new ArrayList();
    }
    ArrayList arrayList = new ArrayList();
    int parseInt = Integer.parseInt(chat.getId());
    for (Message message : chat.getMessages()) {
        Conversation conversation = new Conversation();
        conversation.setThreadId(Integer.valueOf(parseInt));
        conversation.setMessageId(Integer.valueOf(message.getId()));
        conversation.setSenderId(message.getSender());
        conversation.setMessage(message.getMessage());
        conversation.setTimestamp(c.E(message.getDate()));
        arrayList.add(conversation);
    }
    return arrayList;
}
```

Figure 8: ToConversation

It was also discovered two communication services being used:

1. **WhatsApp-Api:** This service was found referenced within the MainActivity file (9).

2. **Firebase:** This service was identified in the ”GridFirebaseMessagingService” class.



```
    private void L() {
        try {
            String whatsappNumber = s.j().k().getWhatsappNumber();
            Intent intent = new Intent("android.intent.action.VIEW");
            intent.setPackage("com.whatsapp");
            intent.setData(Uri.parse("https://api.whatsapp.com/send?phone=" + whatsappNumber));
            startActivity(intent);
        } catch (Exception e10) {
            mb.a.a("MainActivity").c(e10, "openWhatsapp error: ", new Object[0]);
            r(findViewById(2131296465), getString(2131820878), 2131034236);
        }
    }
```

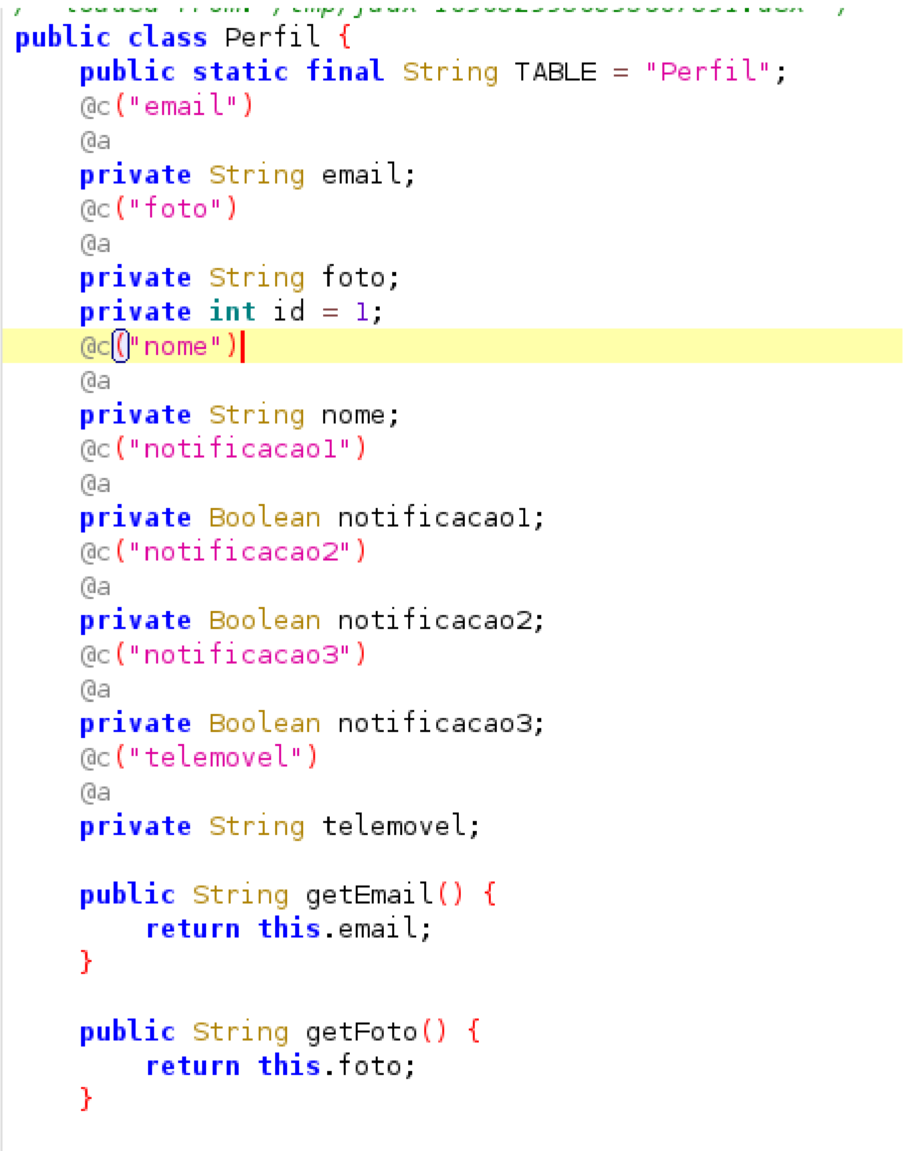
Figure 9: Whatsapp Api

While we found evidence of both services, it’s unclear which one is primarily used or if they operate concurrently. We can’t test it since our analysis suggests that the communication protocol requires initiation by a staff member. The only information observed during the analysis(dynamic) is that the getChat class is the 1st one to be used after opening the chat fragment.

2.6 Data Persistence

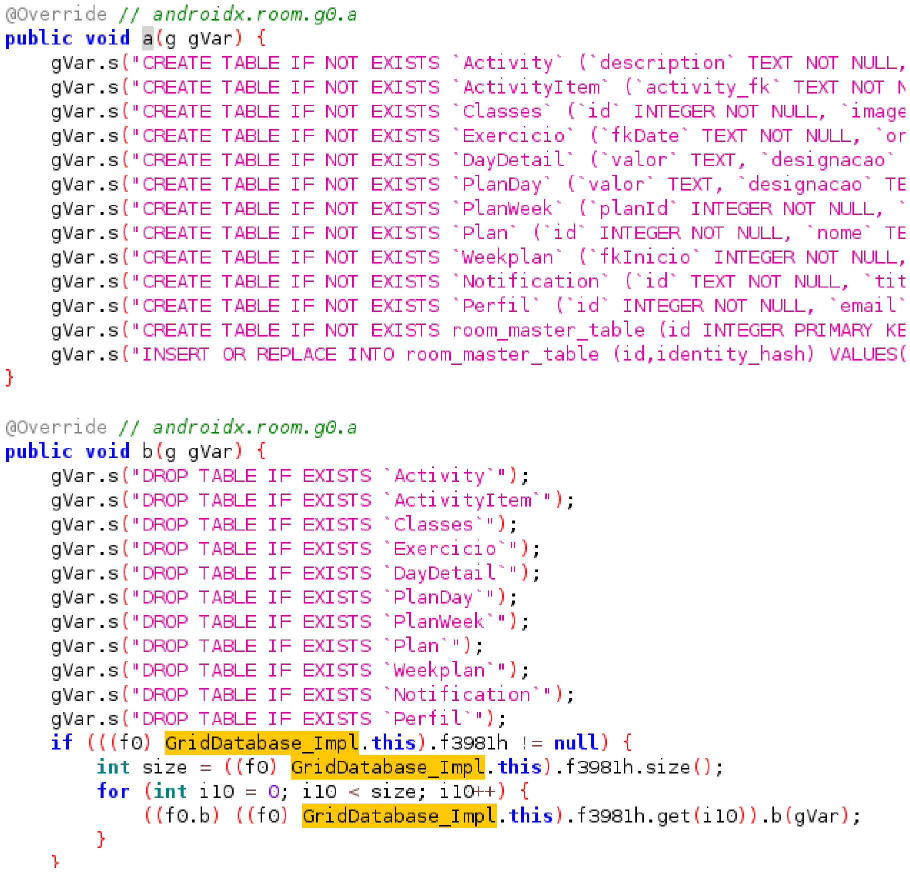
We found on the folder ”**/pt/sincelo/grid/data/model**” many classes used to organize the data available from the profile class (10) to other classes where events and schedules are stored.

We also found on the folder ”**/pt/sincelo/data/local.sources**” that the local database is called ”grid.db” and some queries (11) used to store information locally. This will enable the user to use most features of the app offline.



```
public class Perfil {
    public static final String TABLE = "Perfil";
    @c("email")
    @a
    private String email;
    @c("foto")
    @a
    private String foto;
    private int id = 1;
    @c("nome")
    @a
    private String nome;
    @c("notificacao1")
    @a
    private Boolean notificacao1;
    @c("notificacao2")
    @a
    private Boolean notificacao2;
    @c("notificacao3")
    @a
    private Boolean notificacao3;
    @c("telemovel")
    @a
    private String telemovel;

    public String getEmail() {
        return this.email;
    }

    public String getFoto() {
        return this.foto;
    }
```

Figure 10: Profile Class

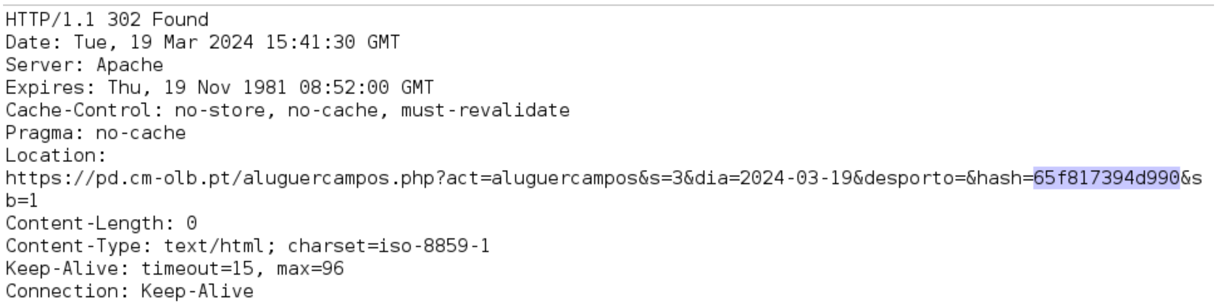


```
@Override // androidx.room.g0.a
public void a(g gVar) {
    gVar.s("CREATE TABLE IF NOT EXISTS `Activity` (`description` TEXT NOT NULL,
    gVar.s("CREATE TABLE IF NOT EXISTS `ActivityItem` (`activity_fk` TEXT NOT N
    gVar.s("CREATE TABLE IF NOT EXISTS `Classes` (`id` INTEGER NOT NULL, `image
    gVar.s("CREATE TABLE IF NOT EXISTS `Exercicio` (`fkDate` TEXT NOT NULL, `or
    gVar.s("CREATE TABLE IF NOT EXISTS `DayDetail` (`valor` TEXT, `designacao`
    gVar.s("CREATE TABLE IF NOT EXISTS `PlanDay` (`valor` TEXT, `designacao` TE
    gVar.s("CREATE TABLE IF NOT EXISTS `PlanWeek` (`planId` INTEGER NOT NULL, `
    gVar.s("CREATE TABLE IF NOT EXISTS `Plan` (`id` INTEGER NOT NULL, `nome` TE
    gVar.s("CREATE TABLE IF NOT EXISTS `Weekplan` (`fkInicio` INTEGER NOT NULL,
    gVar.s("CREATE TABLE IF NOT EXISTS `Notification` (`id` TEXT NOT NULL, `tit
    gVar.s("CREATE TABLE IF NOT EXISTS `Perfil` (`id` INTEGER NOT NULL, `email`
    gVar.s("CREATE TABLE IF NOT EXISTS room_master_table (id INTEGER PRIMARY KE
    gVar.s("INSERT OR REPLACE INTO room_master_table (id,identity_hash) VALUES(
}

@Override // androidx.room.g0.a
public void b(g gVar) {
    gVar.s("DROP TABLE IF EXISTS `Activity`");
    gVar.s("DROP TABLE IF EXISTS `ActivityItem`");
    gVar.s("DROP TABLE IF EXISTS `Classes`");
    gVar.s("DROP TABLE IF EXISTS `Exercicio`");
    gVar.s("DROP TABLE IF EXISTS `DayDetail`");
    gVar.s("DROP TABLE IF EXISTS `PlanDay`");
    gVar.s("DROP TABLE IF EXISTS `PlanWeek`");
    gVar.s("DROP TABLE IF EXISTS `Plan`");
    gVar.s("DROP TABLE IF EXISTS `Weekplan`");
    gVar.s("DROP TABLE IF EXISTS `Notification`");
    gVar.s("DROP TABLE IF EXISTS `Perfil`");
    if (((f0) GridDatabase_Impl.this).f3981h != null) {
        int size = ((f0) GridDatabase_Impl.this).f3981h.size();
        for (int i10 = 0; i10 < size; i10++) {
            ((f0.b) ((f0) GridDatabase_Impl.this).f3981h.get(i10)).b(gVar);
        }
    }
```

Figure 11: Queries

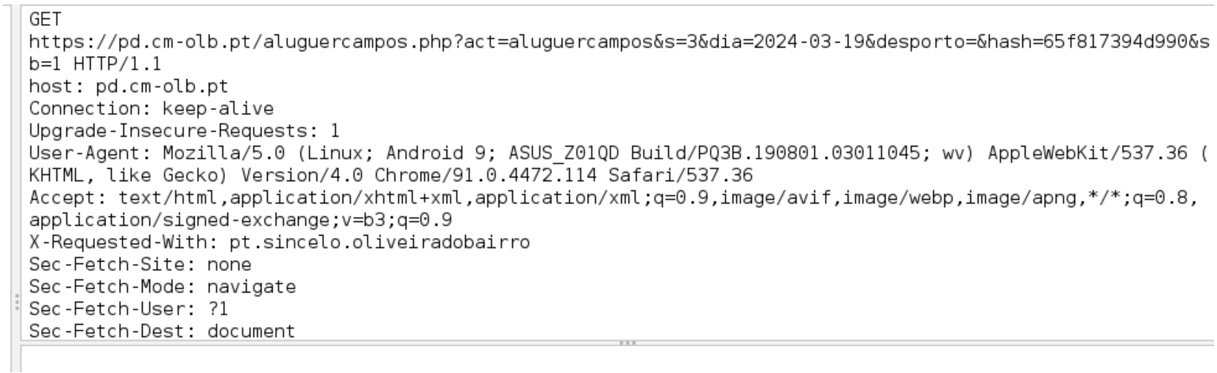
3 Dynamic Analisys

This application works in a very peculiar way, in most cases where it needs to load a new screen instead of loading it from the local XML files or directly accessing the URL with the PHP page, it does a GET request on an endpoint that redirects it to a second one that just then returns the PHP page like the following example.



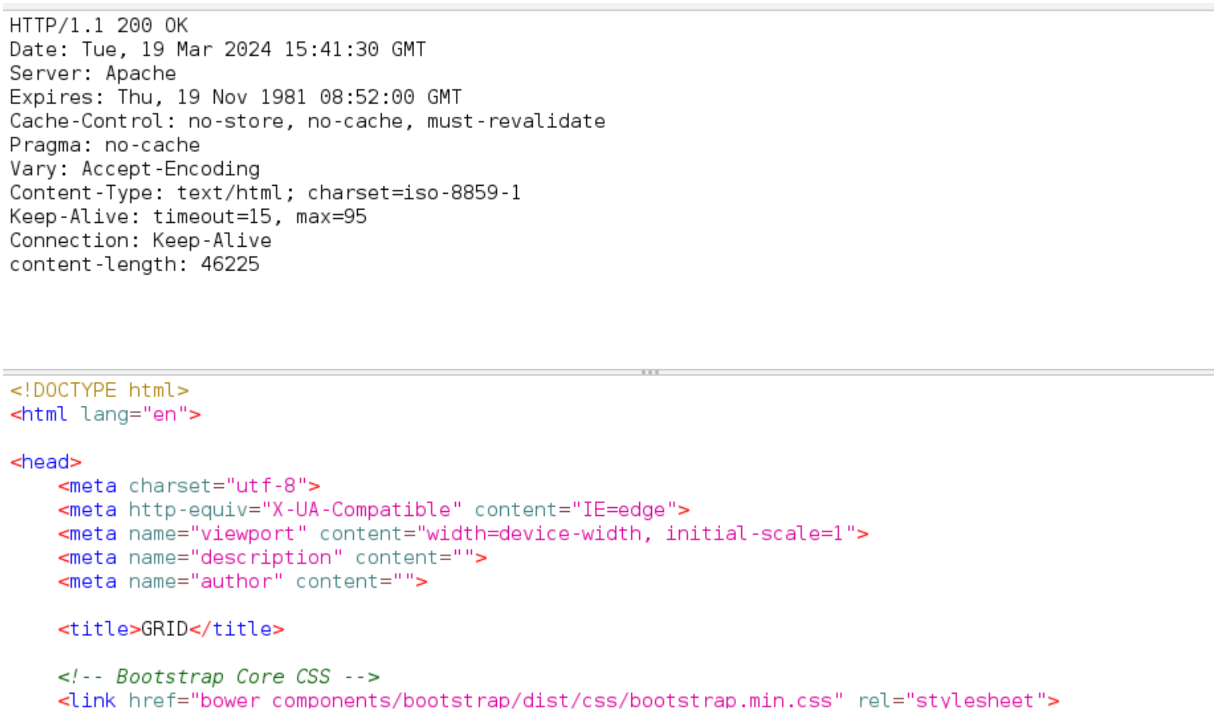
```
HTTP/1.1 302 Found
Date: Tue, 19 Mar 2024 15:41:30 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Location:
https://pd.cm-olb.pt/aluguercampos.php?act=aluguercampos&s=3&dia=2024-03-19&desporto=&hash=65f817394d990&s
b=1
Content-Length: 0
Content-Type: text/html; charset=iso-8859-1
Keep-Alive: timeout=15, max=96
Connection: Keep-Alive
```

Figure 12: Response from the first endpoint where ”location” is the actual URL



```
GET
https://pd.cm-olb.pt/aluguercampos.php?act=aluguercampos&s=3&dia=2024-03-19&desporto=&hash=65f817394d990&s
b=1 HTTP/1.1
host: pd.cm-olb.pt
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Linux; Android 9; ASUS_Z01QD Build/PQ3B.190801.03011045; wv) AppleWebKit/537.36 (
KHTML, like Gecko) Version/4.0 Chrome/91.0.4472.114 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,
application/signed-exchange;v=b3;q=0.9
X-Requested-With: pt.sincelo.oliveiradobairro
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
```

Figure 13: GET request for the endpoint previously mentioned

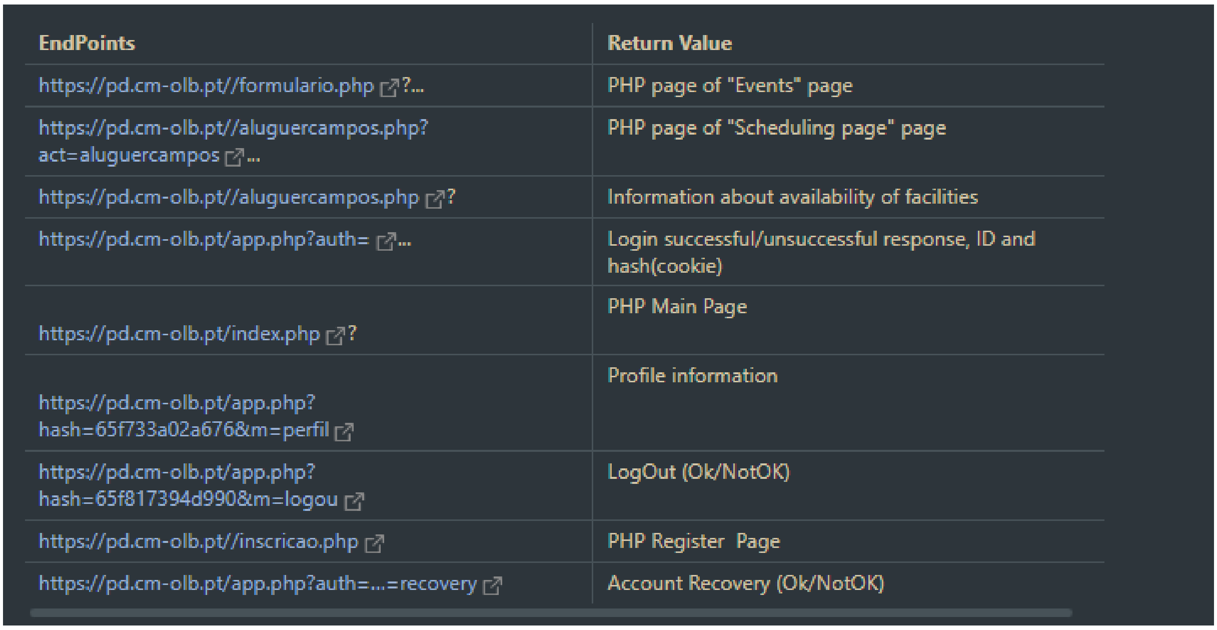


```
HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 15:41:30 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Type: text/html; charset=iso-8859-1
Keep-Alive: timeout=15, max=95
Connection: Keep-Alive
content-length: 46225
```

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>GRID</title>

    <!-- Bootstrap Core CSS -->
    <link href="bower_components/bootstrap/dist/css/bootstrap.min.css" rel="stylesheet">
```

Figure 14: Response with PHP page

3.1 Endpoints



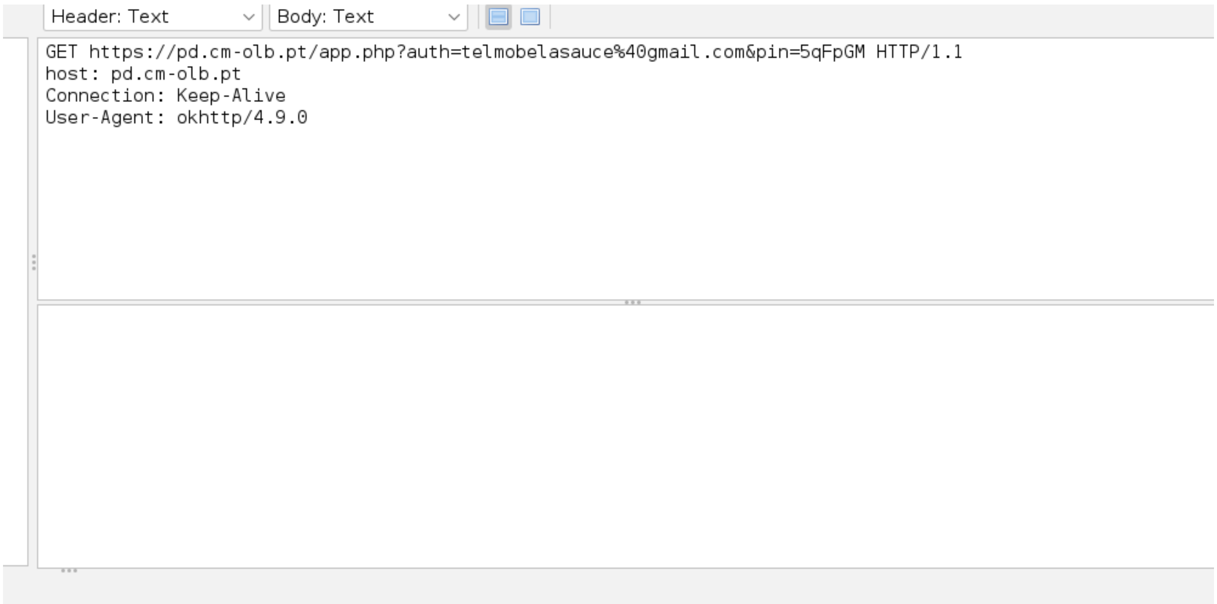
EndPoints	Return Value
https://pd.cm-olb.pt//formulario.php ?...	PHP page of "Events" page
https://pd.cm-olb.pt//aluguercampos.php?act=aluguercampos ...	PHP page of "Scheduling page" page
https://pd.cm-olb.pt//aluguercampos.php ?	Information about availability of facilities
https://pd.cm-olb.pt/app.php?auth= ...	Login successful/unsuccessful response, ID and hash(cookie)
https://pd.cm-olb.pt/index.php ?	PHP Main Page
https://pd.cm-olb.pt/app.php?hash=65f733a02a676&m=perfil	Profile information
https://pd.cm-olb.pt/app.php?hash=65f817394d990&m=logou	LogOut (Ok/NotOK)
https://pd.cm-olb.pt//inscricao.php	PHP Register Page
https://pd.cm-olb.pt/app.php?auth=...=recovery	Account Recovery (Ok/NotOK)

Figure 15:

3.2 Authentication

The first activity being launched on the application is ”SplashActivity” and the corresponding XML file is ”**include_splash_login.xml**”.

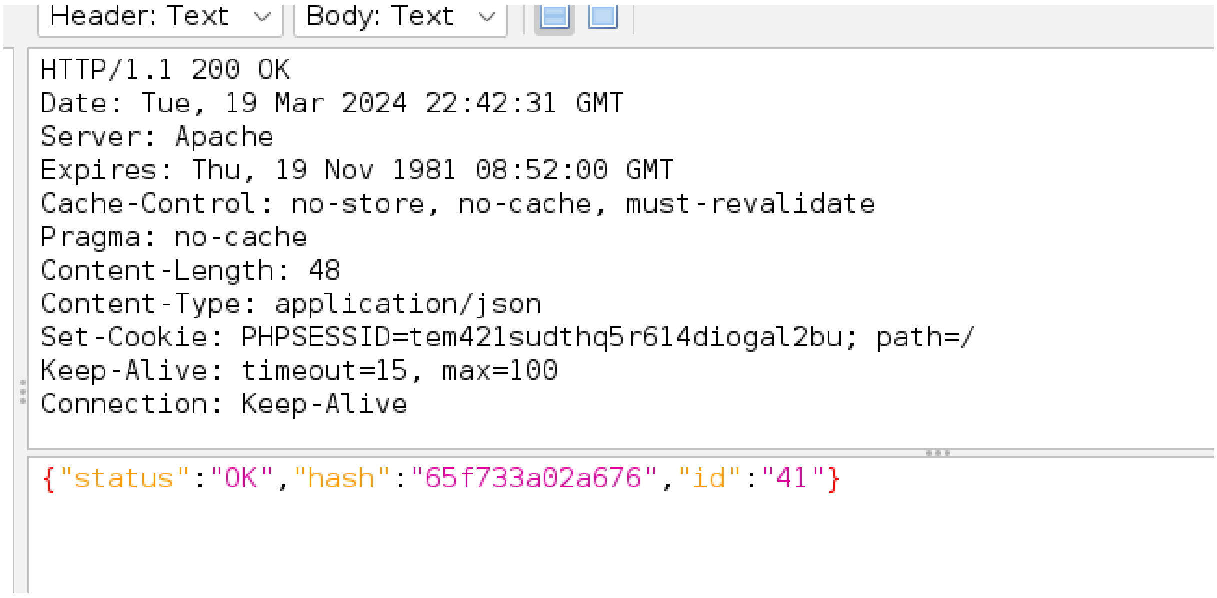
The user upon entering the application gets a form to fill out with a username and password. after entering the correct values the app call to the API with both parameters in plain text ”**https://pd.cm-olb.pt/app.php?auth=telmobelasauce%40gmail.com&pin=5qFpGM**”.



```
GET https://pd.cm-olb.pt/app.php?auth=telmobelasauce%40gmail.com&pin=5qFpGM HTTP/1.1
host: pd.cm-olb.pt
Connection: Keep-Alive
User-Agent: okhttp/4.9.0
```

Figure 16: Login Request

After the response is a ”hash” which will act as an identifier for the user, allowing the app to send this hash connecting each call of the API to the user.

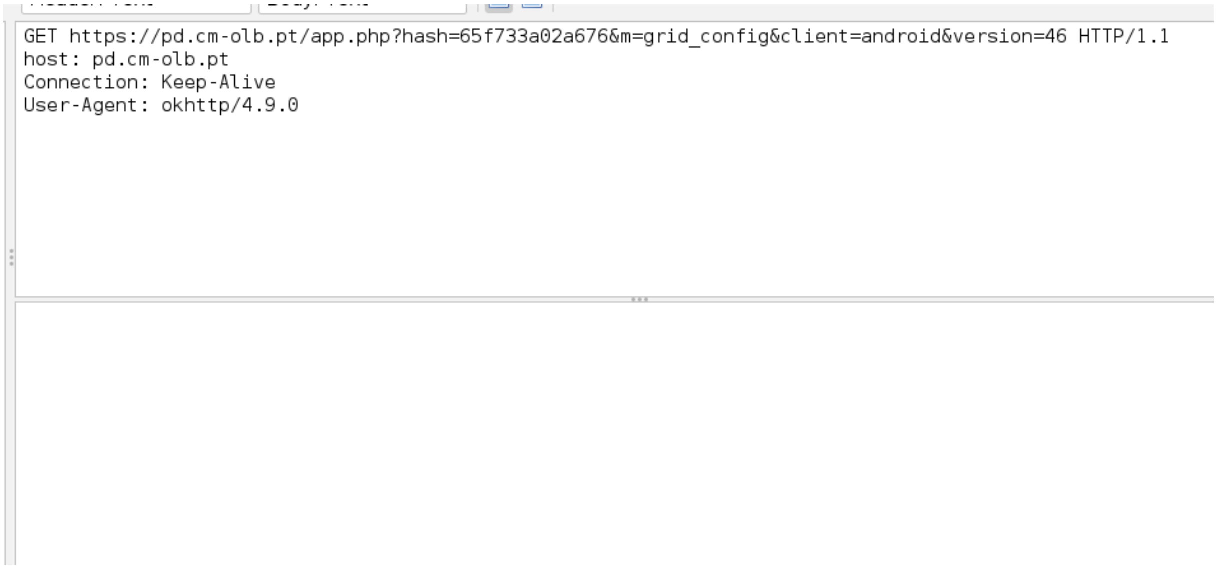


```
HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 22:42:31 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 48
Content-Type: application/json
Set-Cookie: PHPSESSID=tem421sudthq5r614diogal2bu; path=/
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
```

```
{"status":"OK","hash":"65f733a02a676","id":"41"}
```

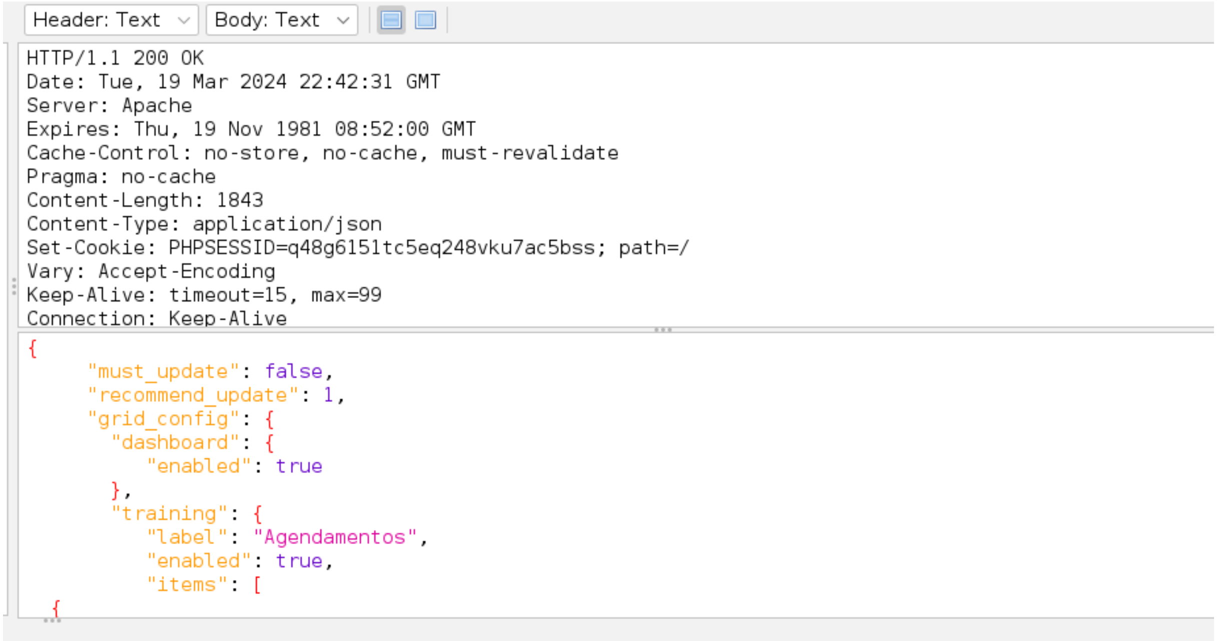
Figure 17: Login Response

After a request to get the allowed permissions and access rights are given. This request is sent to get information that will be stored on the class ”gridConfig” using the class ”GridDatabase_Impls”.



```
GET https://pd.cm-olb.pt/app.php?hash=65f733a02a676&m=grid_config&client=android&version=46 HTTP/1.1
host: pd.cm-olb.pt
Connection: Keep-Alive
User-Agent: okhttp/4.9.0
```

Figure 18: Grid Request

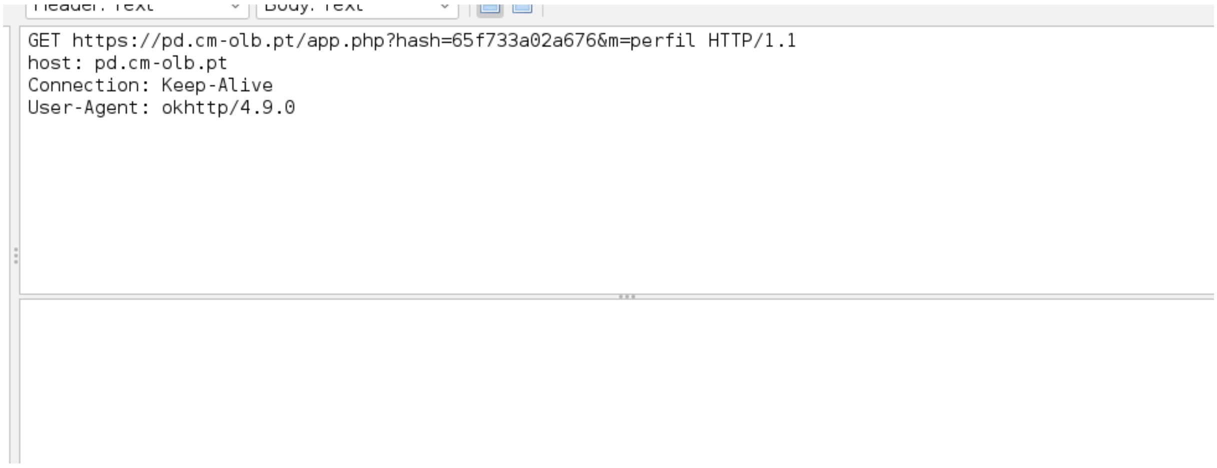


```
HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 22:42:31 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 1843
Content-Type: application/json
Set-Cookie: PHPSESSID=q48g6151tc5eq248vku7ac5bss; path=/
Vary: Accept-Encoding
Keep-Alive: timeout=15, max=99
Connection: Keep-Alive
```

```
{
     "must_update": false,
     "recommend_update": 1,
     "grid_config": {
       "dashboard": {
          "enabled": true
       },
       "training": {
          "label": "Agendamentos",
          "enabled": true,
          "items": [
  {
```

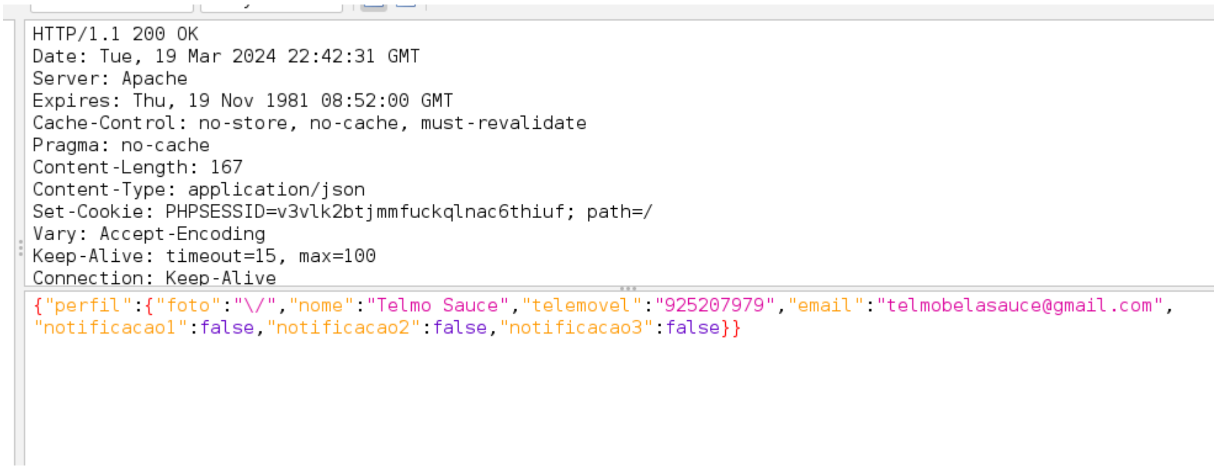
Figure 19: Grid Response

The next package is the profile that as previously stated will store the values on The Profile class using the next call ”**https://pd.cm-olb.pt/app.php?hash=65f733a02a676&m=perfil**”.



```
GET https://pd.cm-olb.pt/app.php?hash=65f733a02a676&m=perfil HTTP/1.1
host: pd.cm-olb.pt
Connection: Keep-Alive
User-Agent: okhttp/4.9.0
```

Figure 20: Profile Request

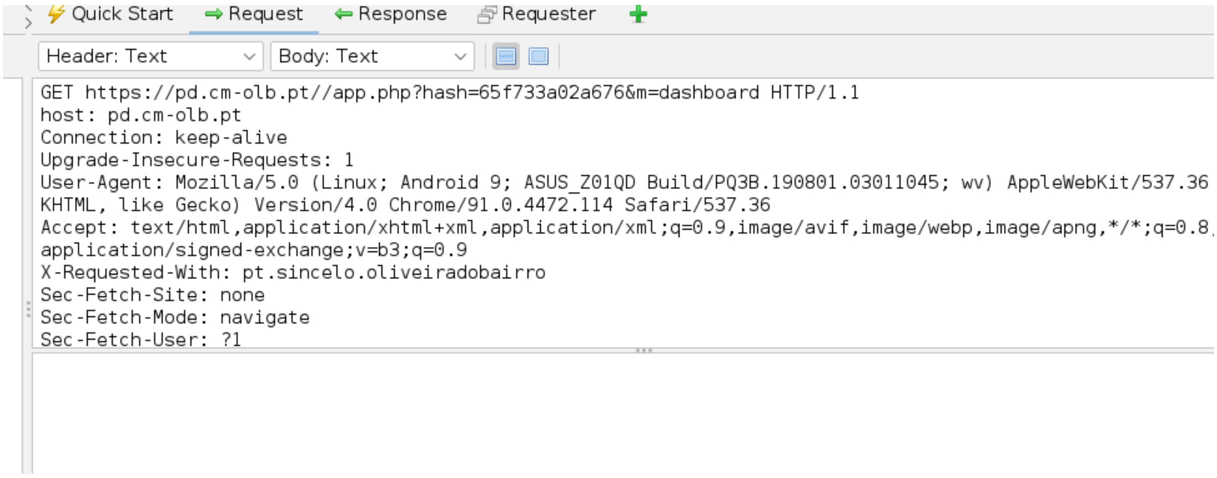


```
HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 22:42:31 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 167
Content-Type: application/json
Set-Cookie: PHPSESSID=v3vlk2btjmmfuckqlnac6thiuf; path=/
Vary: Accept-Encoding
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
```

```
{"perfil":{"foto":"\/","nome":"Telmo Sauce","telemovel":"925207979","email":"telmobelasauce@gmail.com",
"notificacao1":false,"notificacao2":false,"notificacao3":false}}
```

Figure 21: Profile Response

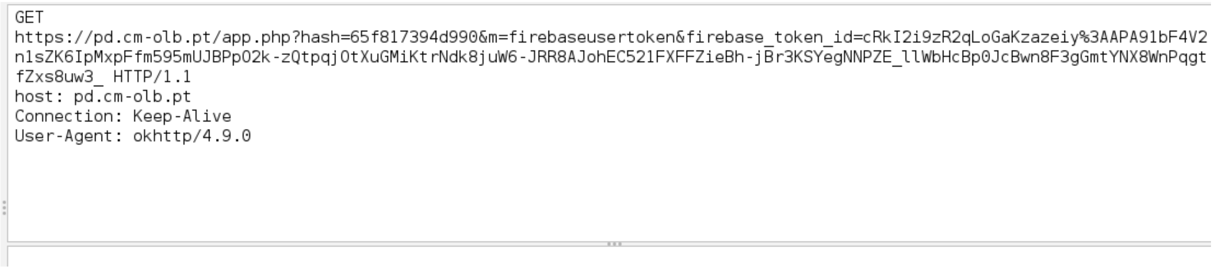
The following call, it's "**https://pd.cm-olb.pt//app.php**?hash=65f733a02a676&m=dashboard" Finally the app will ask for the dashboard to show on the app which will respond with the PHP to show in this case was "**https://pd.cm-olb.pt/index.php?hash=65f733a02a676&sb=1**". This dashboard is requested each time we get to the main page displaying the appropriate dashboard.



```
GET https://pd.cm-olb.pt//app.php?hash=65f733a02a676&m=dashboard HTTP/1.1
host: pd.cm-olb.pt
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Linux; Android 9; ASUS_Z01QD Build/PQ3B.190801.03011045; wv) AppleWebKit/537.36
KHTML, like Gecko) Version/4.0 Chrome/91.0.4472.114 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,
application/signed-exchange;v=b3;q=0.9
X-Requested-With: pt.sincelo.oliveiradobairro
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
```

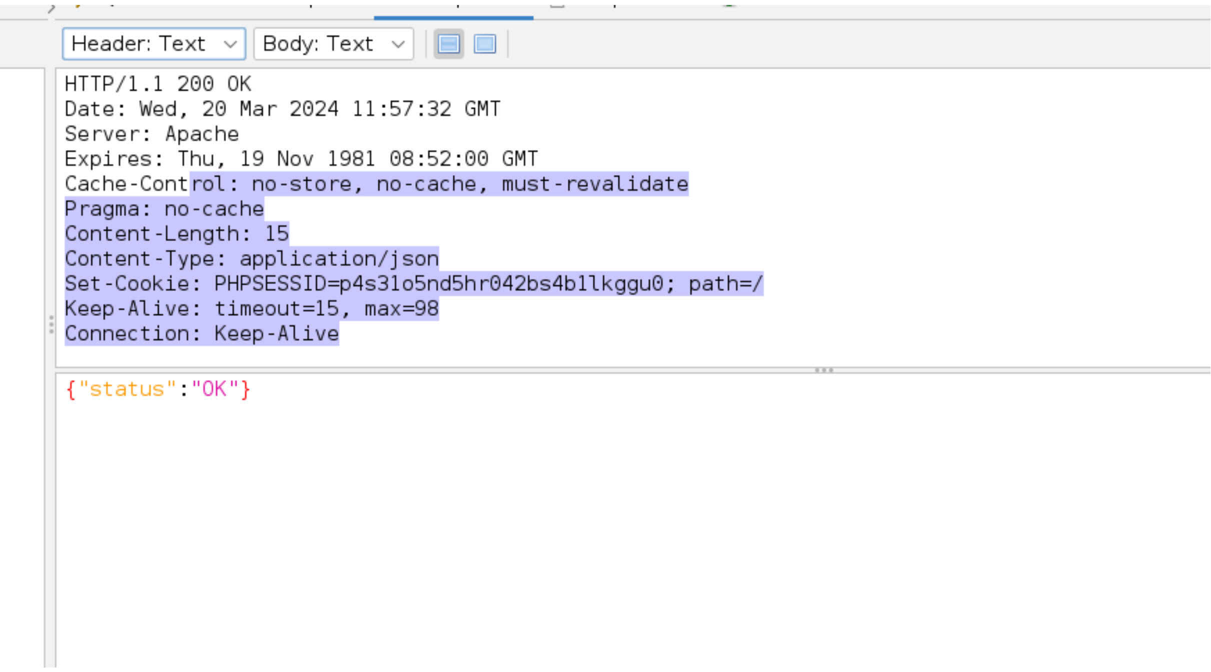
Figure 22: GET Dashboard

Finally, the app will send a Firebase cookie to make the connections to their database available. This cookie is the same independent of session and account.



```
GET
https://pd.cm-olb.pt/app.php?hash=65f817394d990&m=firebaseusertoken&firebase_token_id=cRkI2i9zR2qLoGaKzazeiy%3AAPA91bF4V2
n1sZK6IpMxpFfm595mUJBPp02k-zQtpqj0tXuGMiKtrNdk8juW6-JRR8AJohEC521FXFFZieBh-jBr3KSYegNNPZE_llWbHcBp0JcBwn8F3gGmtYNX8WnPqgt
fZxs8uw3_ HTTP/1.1
host: pd.cm-olb.pt
Connection: Keep-Alive
User-Agent: okhttp/4.9.0
```

Figure 23: Firebase request



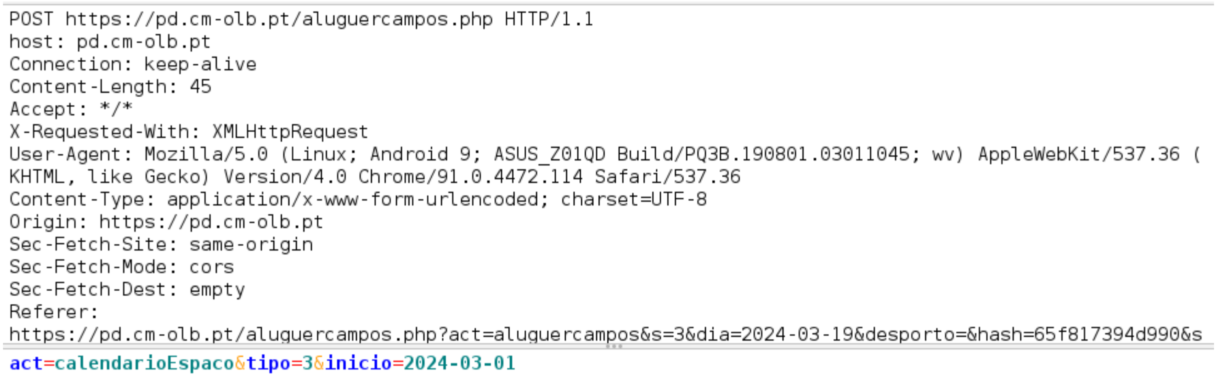
```
HTTP/1.1 200 OK
Date: Wed, 20 Mar 2024 11:57:32 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 15
Content-Type: application/json
Set-Cookie: PHPSESSID=p4s31o5nd5hr042bs4b1lkggu0; path=/
Keep-Alive: timeout=15, max=98
Connection: Keep-Alive
```

```
{"status":"OK"}
```

Figure 24: Firebase Response

3.3 Schedule

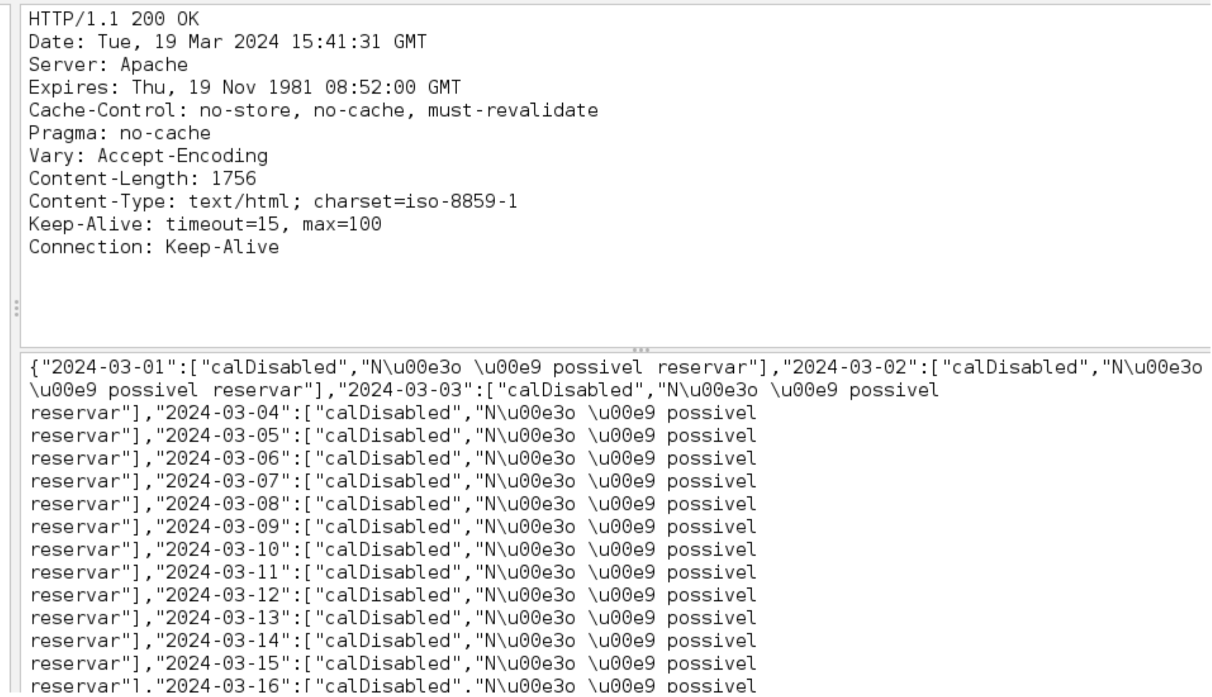
In this section, we will observe the behavior of the app when accessing the scheduling section of the app, although we didn't analyze it fully trying to not disturb the well-functioning of the APP with the scheduling of fake appointments just to see it's inner-workings. Opening the schedule tab of a football field triggers the following POST where the request has the action, the type of building I want to schedule, and a date that has the earliest date shown in the calendar :



```
POST https://pd.cm-olb.pt/aluguercampos.php HTTP/1.1
host: pd.cm-olb.pt
Connection: keep-alive
Content-Length: 45
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Linux; Android 9; ASUS_Z01QD Build/PQ3B.190801.03011045; wv) AppleWebKit/537.36 (
KHTML, like Gecko) Version/4.0 Chrome/91.0.4472.114 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: https://pd.cm-olb.pt
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
https://pd.cm-olb.pt/aluguercampos.php?act=aluguercampos&s=3&dia=2024-03-19&desporto=&hash=65f817394d990&s
```

```
act=calendarioEspaco&tipo=3&inicio=2024-03-01
```

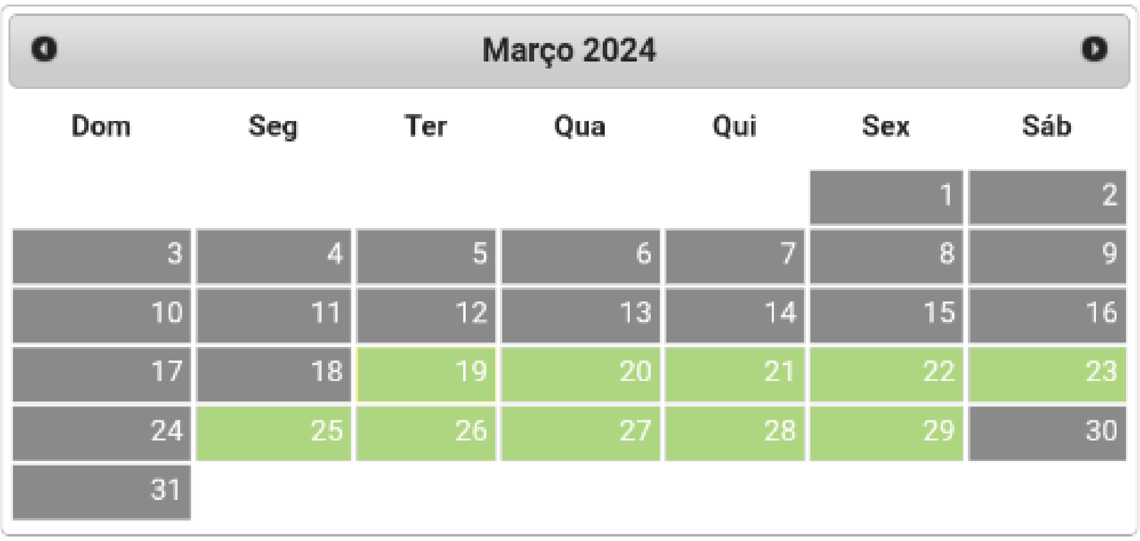
Figure 25: Request



```
HTTP/1.1 200 OK
Date: Tue, 19 Mar 2024 15:41:31 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 1756
Content-Type: text/html; charset=iso-8859-1
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
```

```
{"2024-03-01":["calDisabled","N\u00e3o \u00e9 possivel reservar"],"2024-03-02":["calDisabled","N\u00e3o
\u00e9 possivel reservar"],"2024-03-03":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-04":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-05":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-06":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-07":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-08":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-09":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-10":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-11":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-12":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-13":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-14":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-15":["calDisabled","N\u00e3o \u00e9 possivel
reservar"],"2024-03-16":["calDisabled","N\u00e3o \u00e9 possivel
```

Figure 26: Response



Março 2024

Dom	Seg	Ter	Qua	Qui	Sex	Sáb
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 27: Calendar

4 Vulnerabilites

4.1 Transmition of credentials

As said in the section before after login is done the application sends a GET request to the specific URL, ”https://pd.cm-olb.pt/app.php?auth=telmobelasauce%40gmail.com&pin=5qFpGM”.

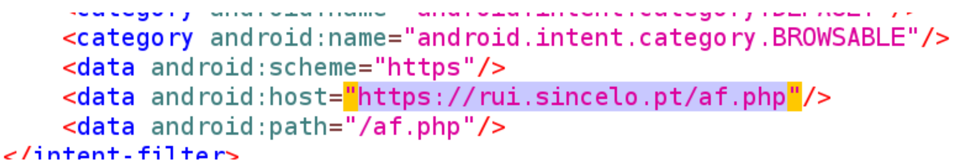
This has a couple of problems and associated CWEs.

- **CWE-598: Use of GET Request Method With Sensitive Query Strings**
  Sensitive information such as session cookies(in this case the hash talked about earlier), contact information(email), and PIN should never be in the URL since this one can saved in the browser's history, passed through Referers to other websites, stored in weblogs, or otherwise recorded in other sources. We recognize the utilization of HTTPS can mitigate this vulnerability but according to the OWASP website " *Simply using HTTPS does not resolve this vulnerability.*"

- **CWE-200: Exposure of Sensitive Information to an Unauthorized Actor** As said before the URL can be stored in the logs of the servers, leading to Unauthorized actors having access to users' personal information.

4.2 Certain components from the development environment are still active in production

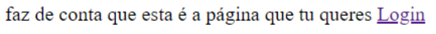
In the AndroidManifest we can find the following snippet:



```
    <category android:name="android.intent.category.DEFAULT" />
    <category android:name="android.intent.category.BROWSABLE"/>
    <data android:scheme="https"/>
    <data android:host="https://rui.sincelo.pt/af.php"/>
    <data android:path="/af.php"/>
</intent-filter>
```

Figure 28: XML Snippet

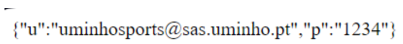
Trying to access the URL leads to this PHP page:



faz de conta que esta é a página que tu queres Login

Figure 29: Result of "https://rui.sincelo.pt/af.php"

Clicking in login redirects us to this page:



```
{"u":"uminhosports@sas.uminho.pt","p":"1234"}
```

Figure 30: Redirectig to ”https://um.sincelo.pt/af.php”

Was conducted an initial assessment to determine whether the account in question was associated with the application. When it was confirmed that it was not, and considering that it falls outside the defined scope, we made the decision not to further investigate (e.g., checking if it was a database account or a mock account). We cannot definitively classify its criticality but it needs attention.

4.3 Play Services SDK vunlnerability

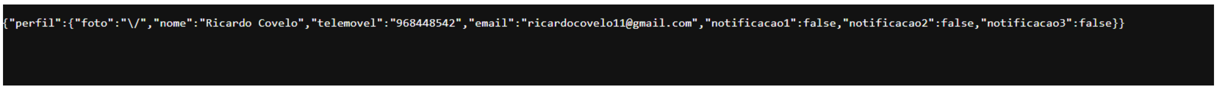
When accessing the versions of third-party services it was found some Outdated services, and some have a CWE attached to them which is:

- **CVE-2022-2390**
  Apps developed with Google Play Services SDK incorrectly had the mutability flag set to PendingIntents that were passed to the Notification service. As Google Play services SDK is so widely used, this bug affects many applications. For an application affected, this bug will let the attacker, gain access to all non-exported providers and/or gain access to other providers the victim has permission. We recommend upgrading to version 18.0.2 of the Play Service SDK as well as rebuilding and redeploying apps.

To address these vulnerabilities, it’s recommended to update all play services SDK to at least version 18.0.2. As we can see in the table 2.2 there were still play services below the required version, making them vulnerable.

4.4 Cookies

The usage of the same hash to identify the users can lead to a common vulnerability: CWE-385: Session Fixation.
The misuse of this cookie present in the URL can lead anyone who can get the cookie to get access to personal information such as shown, where the hash is ”65f817394d990”.



```
{"perfil":{"foto":"\/","nome":"Ricardo Covelo","telemovel":"968448542","email":"ricardocovelo11@gmail.com","notificacao1":false,"notificacao2":false,"notificacao3":false}}
```

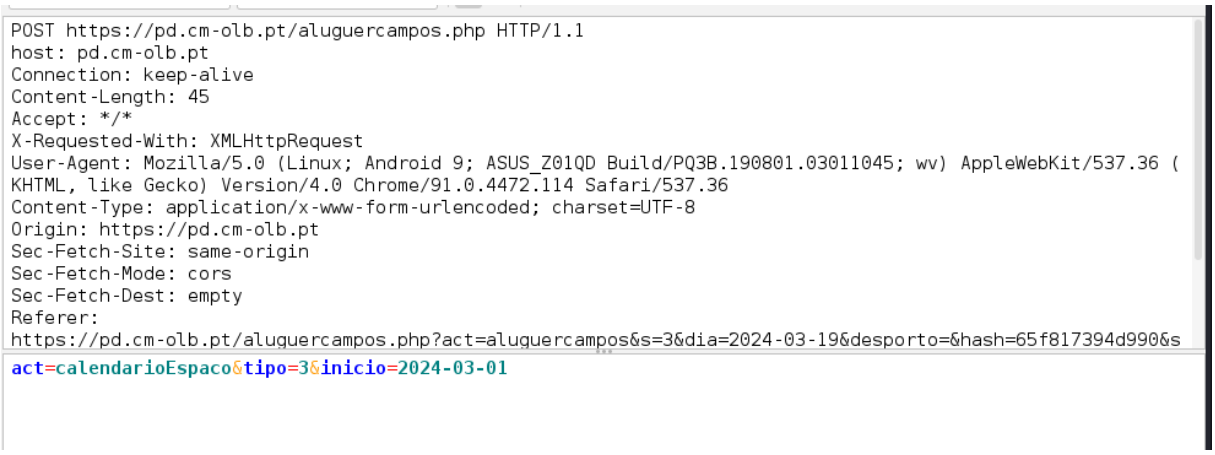
Figure 31: Result of pasting ”https://pd.cm-olb.pt/app.php?hash=65f817394d990&m=perfil” browser

In certain packets, the PHPSESSID was noticed, but it was not clear why it was being used as the hash in the URL was also serving as a session ID cookie. Even though we didn't understand its purpose we believe this isn't being used properly since the phpCookie is constantly changing and it isn't used in most packages.

5 Bad practices

These are some bad practices that although have not a specific CWE can lead to vulnerabilities:

- In some requests, the POSTS are used as GET requests such as the following where it specifies the GET details in the POST request body.



```
POST https://pd.cm-olb.pt/aluguercampos.php HTTP/1.1
host: pd.cm-olb.pt
Connection: keep-alive
Content-Length: 45
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Linux; Android 9; ASUS_Z01QD Build/PQ3B.190801.03011045; wv) AppleWebKit/537.36 (
KHTML, like Gecko) Version/4.0 Chrome/91.0.4472.114 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: https://pd.cm-olb.pt
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
https://pd.cm-olb.pt/aluguercampos.php?act=aluguercampos&s=3&dia=2024-03-19&desporto=&hash=65f817394d990&s

act=calendarioEspaco&tipo=3&inicio=2024-03-01
```

Figure 32: Improper use of POST

6 Conclusion

In conclusion, the analysis provided valuable insights into the application's inner workings, by combining static and dynamic analysis we were able to understand how the app behaved. This helped us understand the app's strengths and weaknesses, especially in terms of security. This analysis will not just contribute to refining the application but also offer essential direction to its developers to implement more secure procedures on the application.