



SPRINT REVIEW III

TIME 06 - SIM CARD



TIME 06



ANDREINA OLIVEIRA

Scrum Master



ANDRÉ RONDI

Developer



EDIVALDO JUNIOR

Developer



STHEFANYE OLIVEIRA

Developer



WESLLEN COSTA

Developer

PRODUCT OWNER'S

A blurred background image of a person sitting at a desk in an office, looking at a computer screen. The person is wearing a dark t-shirt and has long hair. The office has large windows and greenery outside.

ÁUREA MELO

UEA

EDGARD SILVA

UEA

OSCAR NETO

Eldorado

FERANDO ITO

Eldorado



Objetivo

Adicionar no AOSP um recurso que salve um identificador do SIM Card para detectar e notificar se houver troca por outro SIM Card.



Critérios de Aceitação

- Identificar SIM Card.
- Notificar trocas de SIM Card.
- Controle de SIM Card.

SPRINT III

Início da Sprint: 24/05/21

Fim da Sprint: 29/05/21

Objetivo:

- Desenvolver Logs de Inserção e remoção
- Armazenamento dos Logs
- Notificar Trocas do SIM Card
- Monitoramento em Background
- App Nativo

SPRINT BACKLOG

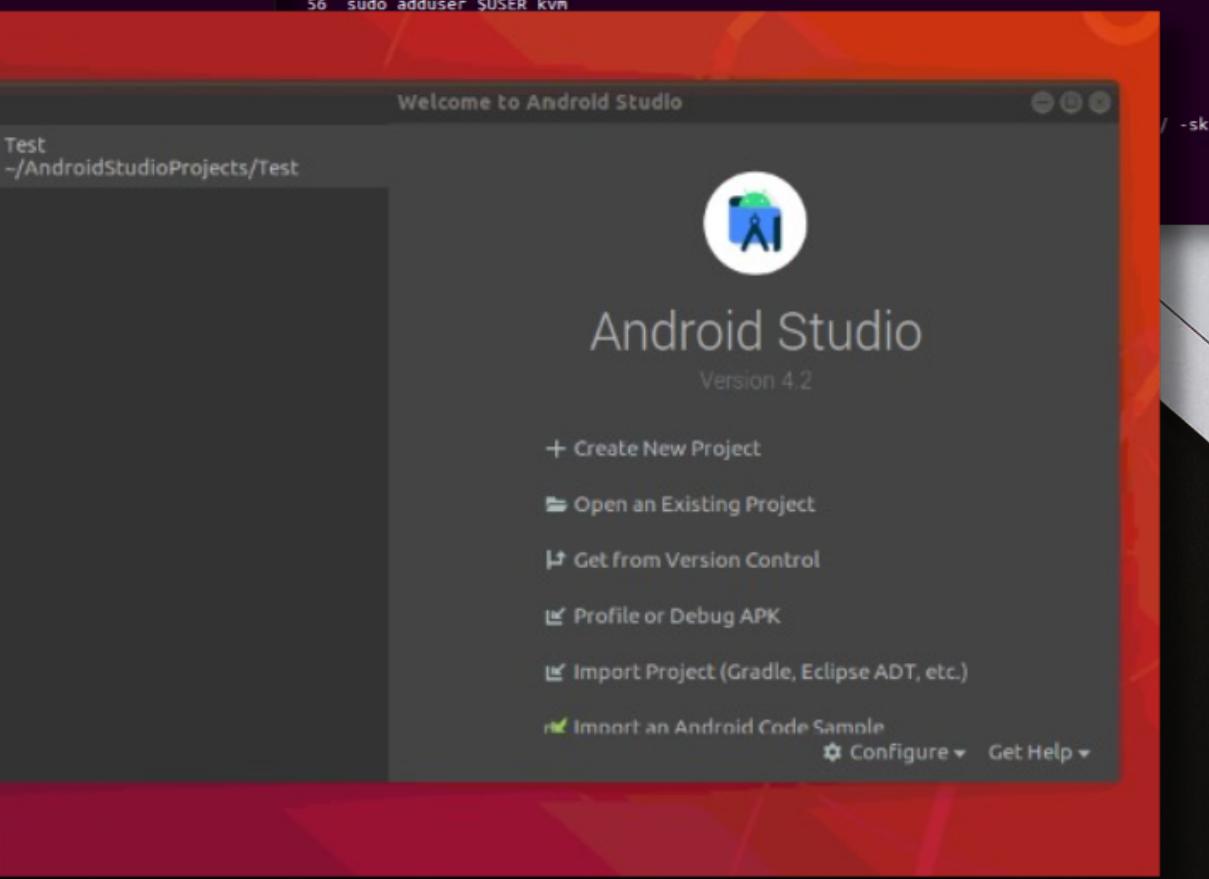
SPRINT BACKLOG				
ID	Tarefa	Previsto	Realizado	Prioridade
#31	Preparação do ambiente (2º máquina)	6:00		BAIXA
#32	Desenvolver Log de Inserção e Remoção	3:00		ALTA
#33	Criar aquivo txt para receber dados de logs	6:00		ALTA
#35	Notificar Troca de SIM Card	9:00		ALTA
#36	Tornar o App Nativo	12:00		ALTA
#37	App em background	9:00		ALTA

Previsto	Realizado	Restante
45:00:00	0:00:00	45:00:00

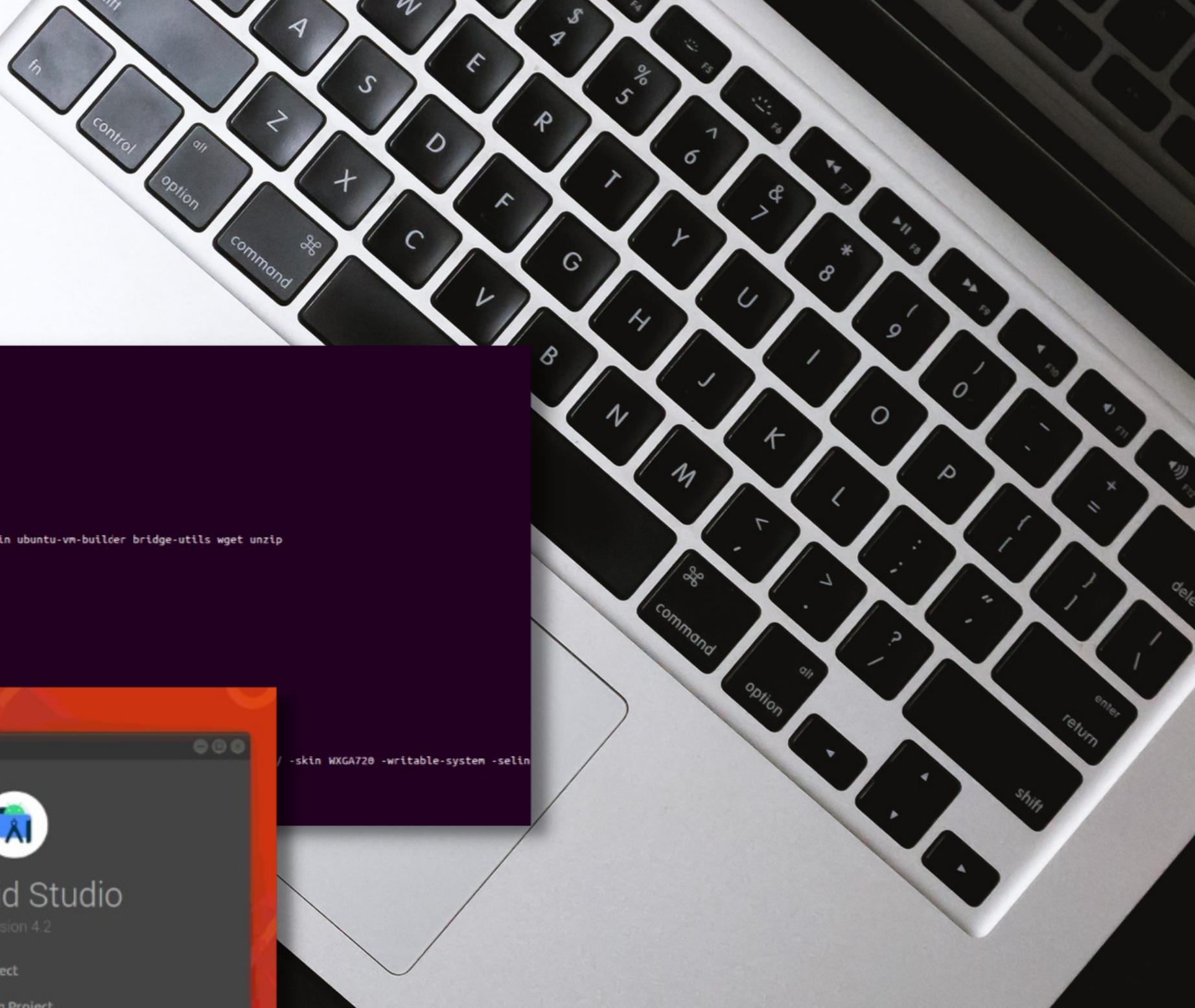
#31

AMBIENTE

```
aluno@aluno-Lenovo-Y720-15IKB:~/android_aosp_source_code$ lunch aosp_x86_64-eng
=====
PLATFORM_VERSION_CODENAME=REL
PLATFORM_VERSION=9
TARGET_PRODUCT=aosp_x86_64
TARGET_BUILD_VARIANT=eng
TARGET_BUILD_TYPE=release
TARGET_ARCH=x86_64
TARGET_ARCH_VARIANT=x86_64
TARGET_2ND_ARCH=x86
TARGET_2ND_ARCH_VARIANT=x86_64
HOST_ARCH=x86_64
HOST_2ND_ARCH=x86
HOST_OS=linux
HOST_OS_EXTRA=Linux-5.4.0-73-generic-x86_64-Ubuntu-18.04.5-LTS
HOST_CROSS_OS=windows
HOST_CROSS_ARCH=x86
HOST_CROSS_2ND_ARCH=x86_64
HOST_BUILD_TYPE=release
BUILD_ID=PQ1A.181205.006.A1
OUT_DIR=out
=====
aluno@aluno-Lenovo-Y720-15IKB:~/android_aosp_source_code$ emulator \
> -skindir $HOME/android_aosp_source_code/development/tools/emulator/skins \
> -writable-system \
> -selinux permissive \
> -no-audio \
> -show-kernel -shell-serial pty \
> -qemu -serial tcp::5000,server,nowait
emulator: WARNING: Couldn't find crash service executable /home/aluno/a
emulator: WARNING: system partition size adjusted to match image file (
```



```
30 lunch
31 env | grep "ANDROID\|TARGET"
33 m -j5
34 ls out/target/product/generic_x86_64/
35 emulador -help
36 emulator -help
37 emulator
38 sudo emulator
39 cd packages/apps/Settings
40 nano res/values/strings.xml
41 cd ..
42 croot
43 m -j6
44 emulator
45 sudo apt install -y qemu-kvm libvirt-bin ubuntu-vm-builder bridge-utils wget unzip
46 sudo usermod -aG kvm raj
47 sudo usermod -aG kvm aluno
48 sudo usermod -aG libvirt aluno
49 emulator
50 ls
51 emulator
52 sudo android
53 sudo apt install qemu-kvm
54 ls -al /dev/kvm
55 grep kvm /etc/group
56 sudo adduser $USER kvm
```

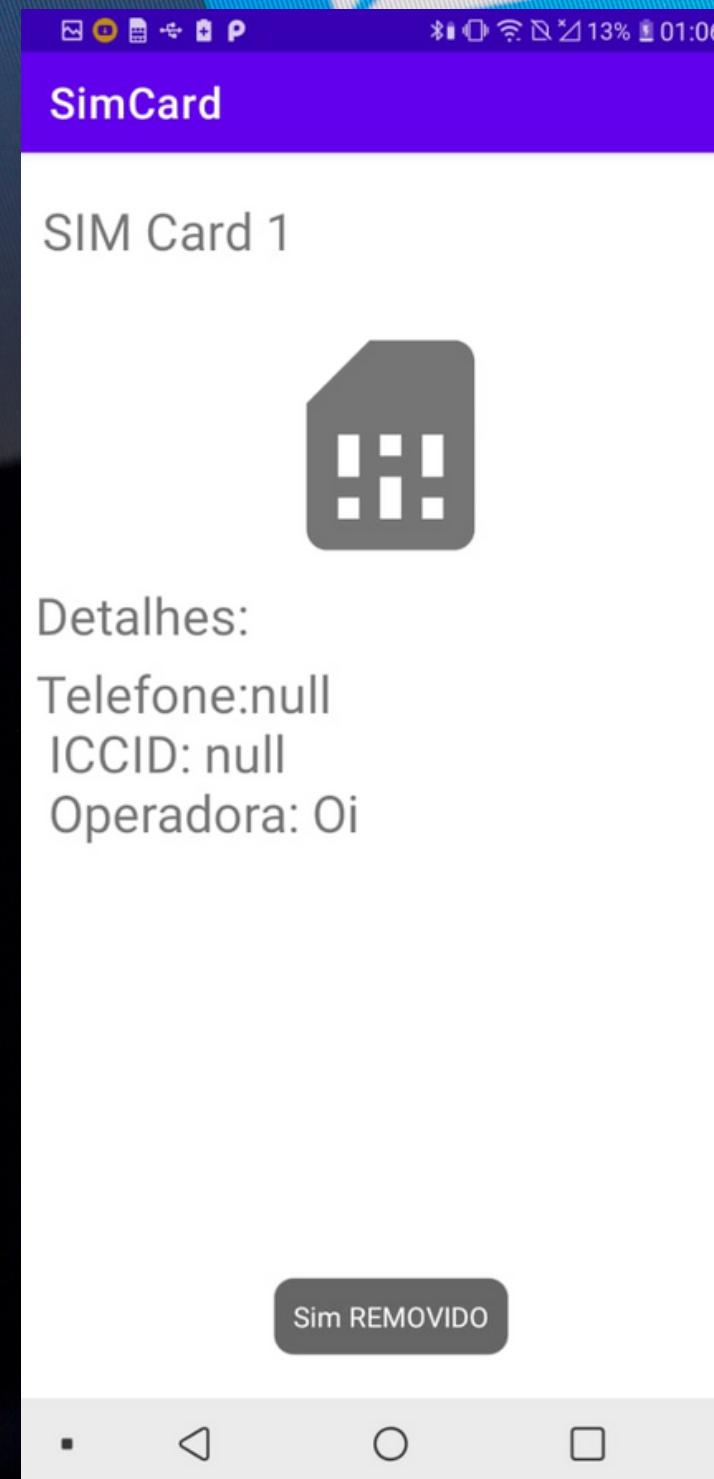
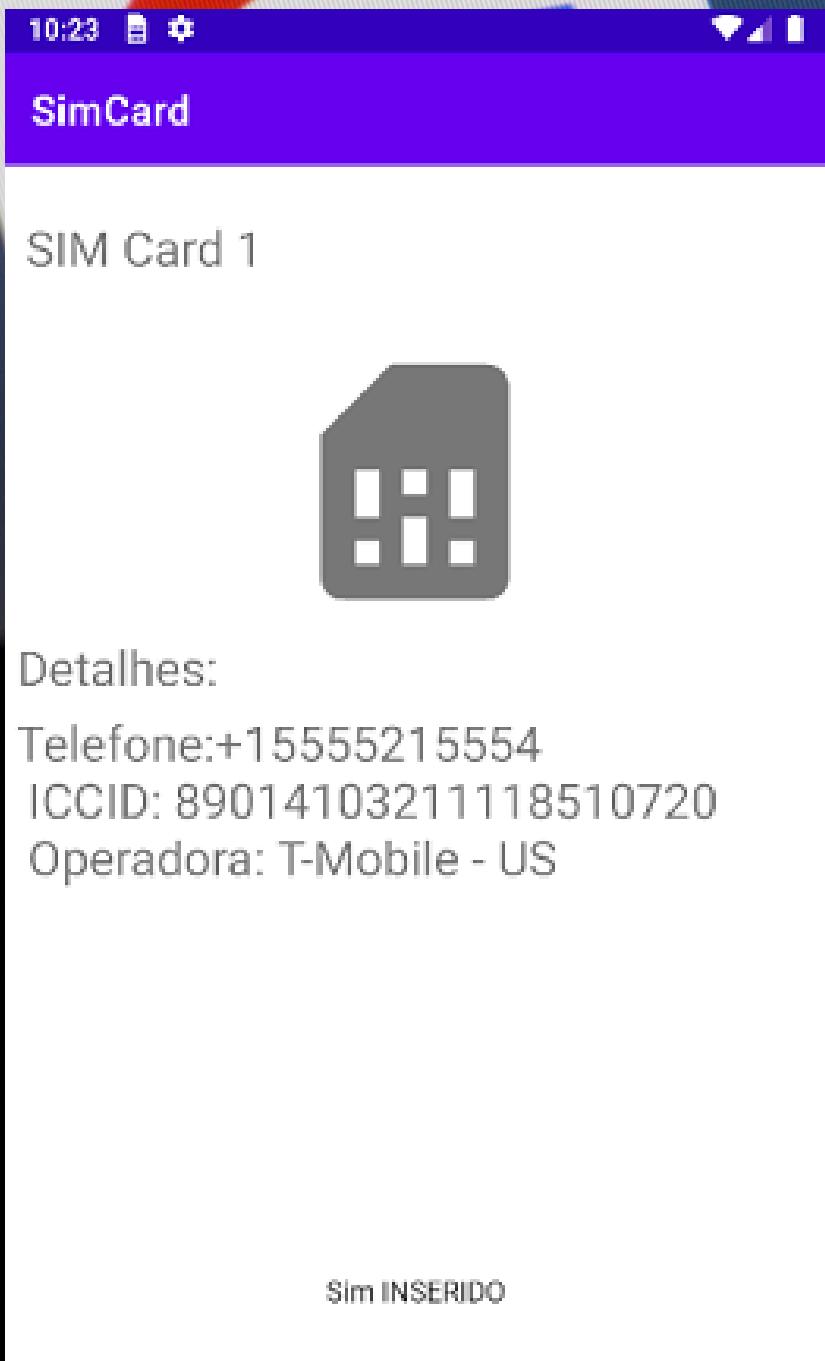


ANDRÉ RONDI

Previsto: 6h
Realizado: 6h

#35

NOTIFICAR INSERÇÃO / REMOÇÃO



```
private fun observeData() {
    viewModel!!.observeLiveSimState().observe(owner: this,
    { value ->
        if (value) {
            createNotificationChannel()
            sendNotification()
            Toast.makeText( context: this, text: "Sim INSERIDO", Toast.LENGTH_SHORT).show()
            Log.i(TAG, msg: "ESTADO DO SimCard: SIM INSERIDO / SimState = " + value)
        } else {
            Toast.makeText( context: this, text: "Sim REMOVIDO", Toast.LENGTH_SHORT).show()
            Log.i(TAG, msg: "ESTADO DO SimCard: SIM REMOVIDO / SimState = " + value)
        }
    }
}
```



STHEFANYE

Previsto: 9h
Realizado: 7h

#32

LOG INSERT/REMOVED

```
tety@sthe:~$ adb logcat | grep Time06
05-29 02:38:36.210 20023 20023 I Time06 : ESTADO DO SimCard: SIM INSERIDO / SimState = true
05-29 02:38:36.299 2802 2802 D NotificationServiceEx: notification=Notification(channel=Time06Notify pri=0 contentView=null vibrate=null sound=null defaults=0x0 flags=0x0 color=0x00000000 backgroundColor=0x00000000 vis=PRIVATE)
05-29 02:38:36.301 20023 20023 I Time06 : ESTADO DO SimCard: SIM INSERIDO / SimState = true
05-29 02:38:36.396 2802 2802 D NotificationServiceEx: notification=Notification(channel=Time06Notify pri=0 contentView=null vibrate=null sound=null defaults=0x0 flags=0x0 color=0x00000000 backgroundColor=0x00000000 vis=PRIVATE)
05-29 02:38:44.899 20023 20023 I Time06 : Numero: +5592999823003
05-29 02:38:44.904 20023 20023 I Time06 : ICCID: 8955319929986822644
05-29 02:38:44.910 20023 20023 I Time06 : Operadora: Oi
```

```
05-29 02:39:05.688 20023 20023 I Time06 : ESTADO DO SimCard: SIM REMOVIDO / SimState = false
05-29 02:39:05.733 20023 20023 I Time06 : ESTADO DO SimCard: SIM REMOVIDO / SimState = false
^[[05-29 02:39:46.490 20023 20023 I Time06 : ESTADO DO SimCard: SIM REMOVIDO / SimState = false
05-29 02:39:46.545 20023 20023 I Time06 : ESTADO DO SimCard: SIM REMOVIDO / SimState = false
05-29 02:39:50.643 20023 20023 I Time06 : Numero: null
05-29 02:39:50.650 20023 20023 I Time06 : ICCID: null
05-29 02:39:50.652 20023 20023 I Time06 : Operadora: null
05-29 02:39:50.721 20023 20023 I Time06 : ESTADO DO SimCard: SIM REMOVIDO / SimState = false
05-29 02:39:50.882 20023 20023 T Time06 : ESTADO DO SimCard: STM REMOVIDO / SimState = false
```



STHEFANYE

Previsto: 3h
Realizado: 2h

#33

ARQUIVO TXT

```
private fun checkDirectory(): File {  
    val path = this.context.filesDir.absolutePath  
    val logTxtDirectory = File(path, "LogTxt")  
    if (!logTxtDirectory.isDirectory) logTxtDirectory.mkdirs()  
    Log.i("Diretorio", path)  
    return logTxtDirectory  
}  
  
private fun txtGenerator(){  
  
    val fileTxt = File(checkDirectory(), "logSimCard.txt")  
    fileTxt.appendText(getSerialNumber())  
    val inputAsString = FileInputStream(fileTxt).bufferedReader().use { it.readText() }  
    Log.i("Retorno", inputAsString)  
}
```



WESLLEN

Previsto: 6h
Realizado: 6h

#37

BACKGROUND

```
override fun onStartCommand(intent: Intent?, flags: Int, startId: Int): Int {  
    showLog("OnStartCommand")  
  
    val runnable = Runnable {  
        for(i in 1..1000){  
            showLog("Service doing something " + i.toString())  
            Thread.sleep(1000)  
        }  
    }  
  
    val thread = Thread(runnable)  
    thread.start()  
  
    return super.onStartCommand(intent, flags, startId)  
}
```

```
scientist.rb x default.rb x observation.rb  
1 | The mutable result of running an experiment.  
2 class Scientist::Result  
3 # An Array of candidate Observations.  
4 attr_reader :candidates  
5 # The control Observation to which the rest are compared.  
6 attr_reader :control  
7 # An Experiment.  
8 attr_reader :experiment  
9  
10 Array of observations which didn't match the control.  
11 reader :ignored  
12 Array of observations which didn't match the control.  
13 reader :mismatched  
14 Array of Observations in execution order.  
15 reader :observations  
16 Internal: Create a new result.  
17 experiment - the Experiment this result is for.  
18 observations - an Array of Observations, in execution order.  
19 control - the control Observation  
20  
21 initialize(experiment, observations = [], control = null)  
22     experiment = experiment  
23     observations = observations  
24     control = control  
25     candidates = observations - [control]  
26     evaluate_candidates  
27  
28     freeze  
29  
30     public: the experiment's context  
31     context  
32     experiment.context  
33  
34     public: the name of  
35     experiment_name  
36     experiment.name  
37  
38     end  
39  
40     # Public: was the result matched?  
41     def matched?  
42         ...  
43     lib/scientist/result.rb 1:1
```

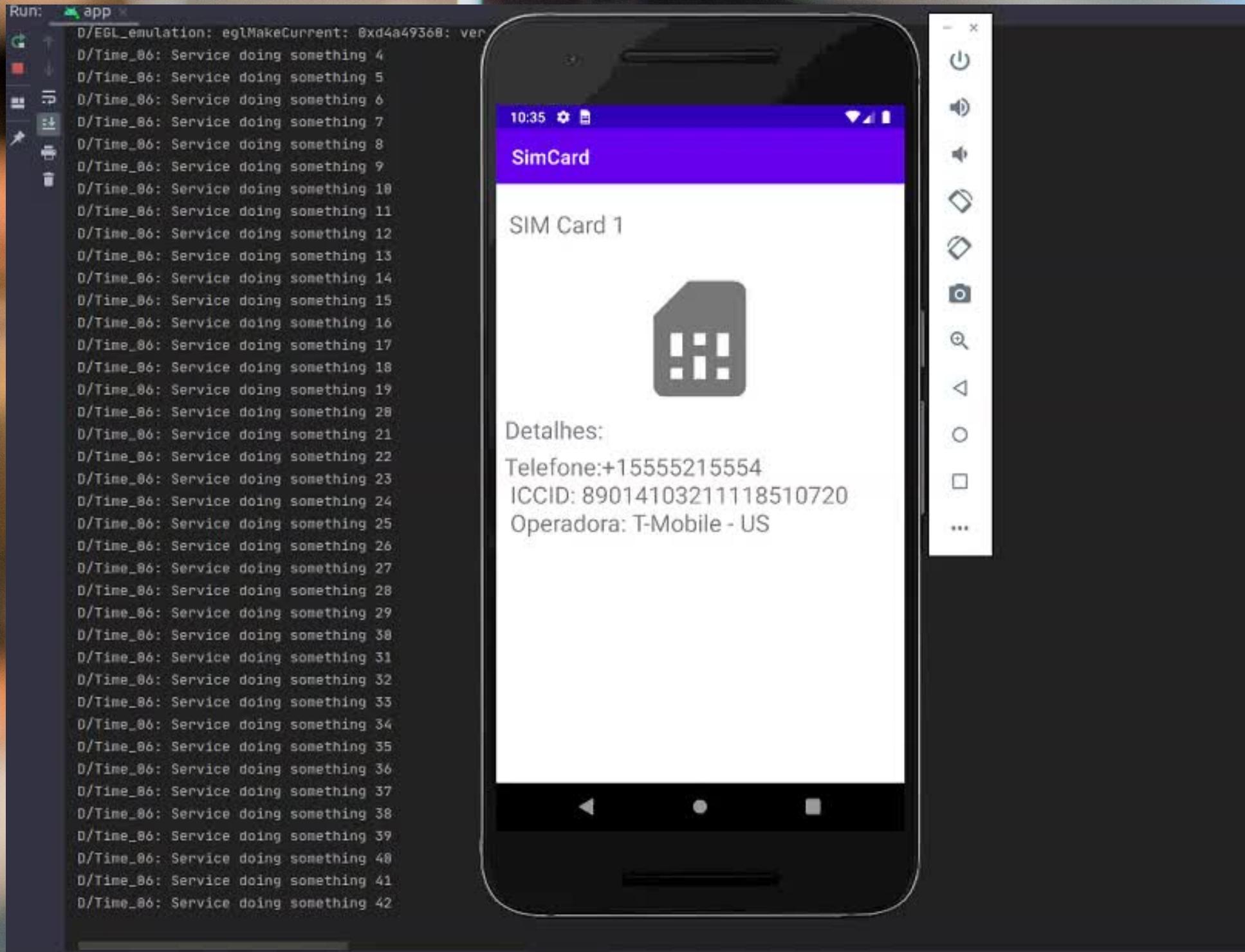


ANDREINA

Previsto: 9h
Realizado: 4h

#37

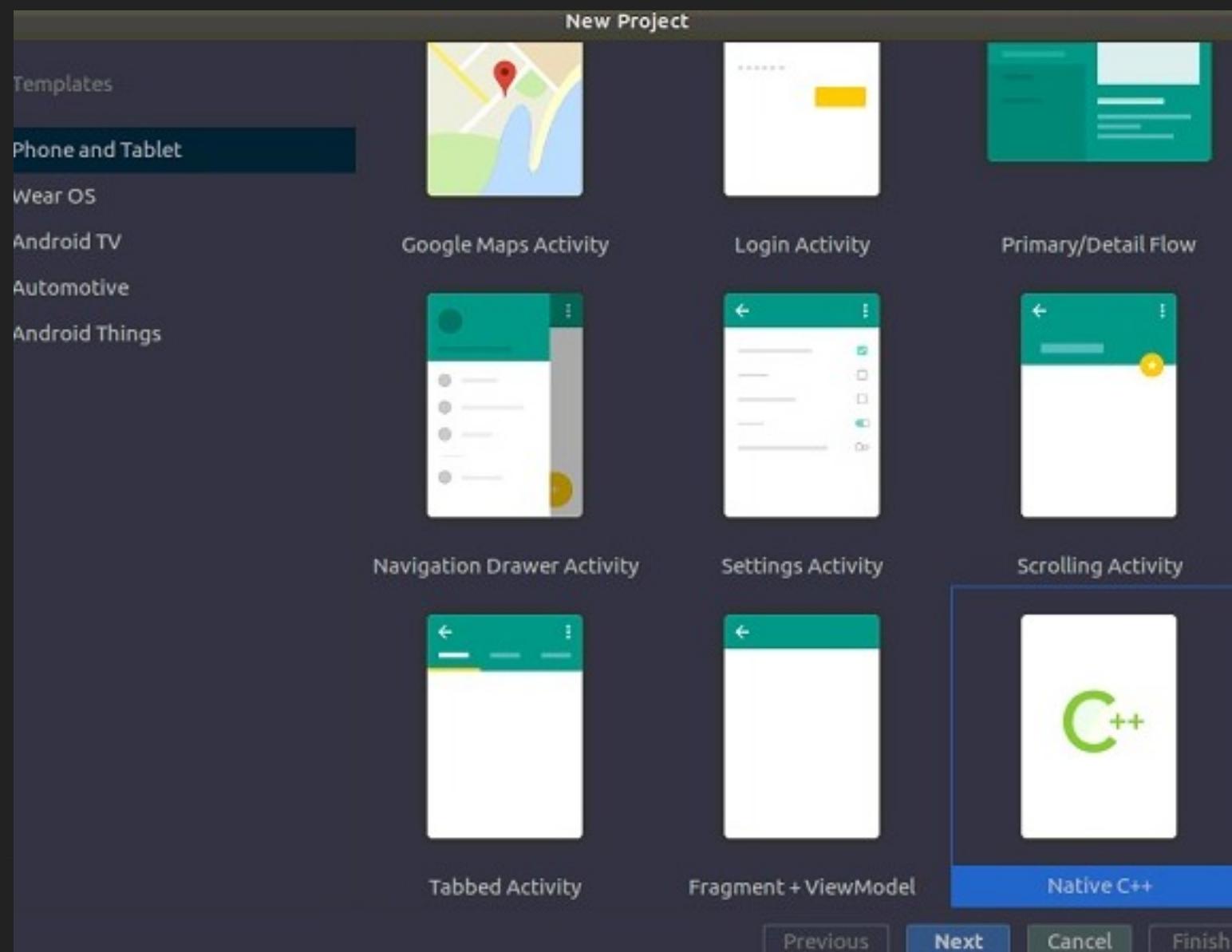
BACKGROUND



```
Run: app
D/EGL_emulation: eglMakeCurrent: 0xd4a49368: ver 2 3
D/Time_86: Service doing something 4
D/Time_86: Service doing something 5
D/Time_86: Service doing something 6
D/Time_86: Service doing something 7
D/Time_86: Service doing something 8
D/Time_86: Service doing something 9
D/Time_86: Service doing something 10
D/Time_86: Service doing something 11
D/Time_86: Service doing something 12
D/Time_86: Service doing something 13
D/Time_86: Service doing something 14
D/Time_86: Service doing something 15
D/Time_86: Service doing something 16
D/Time_86: Service doing something 17
D/Time_86: Service doing something 18
D/Time_86: Service doing something 19
D/Time_86: Service doing something 20
D/Time_86: Service doing something 21
D/Time_86: Service doing something 22
D/Time_86: Service doing something 23
D/Time_86: Service doing something 24
D/Time_86: Service doing something 25
D/Time_86: Service doing something 26
D/Time_86: Service doing something 27
D/Time_86: Service doing something 28
D/Time_86: Service doing something 29
D/Time_86: Service doing something 30
D/Time_86: Service doing something 31
D/Time_86: Service doing something 32
D/Time_86: Service doing something 33
D/Time_86: Service doing something 34
D/Time_86: Service doing something 35
D/Time_86: Service doing something 36
D/Time_86: Service doing something 37
D/Time_86: Service doing something 38
D/Time_86: Service doing something 39
D/Time_86: Service doing something 40
D/Time_86: Service doing something 41
D/Time_86: Service doing something 42
scientist.rb x default.rb x observation.rb
x scientist.rb x default.rb x observation.rb
1 | The mutable result of running an experiment.
2 | class Scientist::Result
3 |   # An Array of candidate Observations.
4 |   attr_reader :candidates
5 |   # The control Observation to which the rest are compared.
6 |   attr_reader :control
7 |
8 |   # An Experiment.
9 |   attr_reader :experiment
10 |  # An Array of observations which didn't match the control.
11 |  attr_reader :ignored
12 |  # An Array of observations which didn't match the control.
13 |  attr_reader :mismatched
14 |  # An Array of Observations in execution order.
15 |  attr_reader :observations
16 |
17 |  # Internal: Create a new result.
18 |
19 |  # experiment - the Experiment this result is for.
20 |  # observations: - an Array of Observations, in execution order.
21 |  # control: - the control Observation
22 |
23 |  # Initialize
24 |  def initialize(experiment, observations = [], control = nil)
25 |    @experiment = experiment
26 |    @observations = observations
27 |    @control = control
28 |    @candidates = observations - [control]
29 |    evaluate_candidates
30 |
31 |    freeze
32  end
33
34
35
36
37
38 |  # Public: the experiment's context
39 |  def context
40 |    experiment.context
41 |  end
42
43 |  # Public: the name of the experiment
44 |  def experiment_name
45 |    experiment.name
46 |  end
47
48 |  # Public: was the result matched?
49 |  def matched?
50 |    ...
51 |
52 |  # Internal: Evaluate the candidates.
53 |  private
54 |  def evaluate_candidates
55 |    @candidates.each do |candidate|
56 |      if candidate == @control
57 |        @ignored.push(candidate)
58 |      else
59 |        @mismatched.push(candidate)
60 |      end
61 |    end
62 |  end
63
64 |  # Internal: Freeze the result.
65 |  private
66 |  def freeze
67 |    @control = nil
68 |    @candidates = nil
69 |    @ignored = nil
70 |    @mismatched = nil
71 |    @observations = nil
72 |  end
73
74 |  # Internal: Hash the result.
75 |  private
76 |  def hash
77 |    @control.hash
78 |  end
79
80 |  # Internal: Equality check.
81 |  private
82 |  def ==(other)
83 |    other.is_a?(Scientist::Result) && other.control == control
84 |  end
85
86 |  # Internal: To string.
87 |  private
88 |  def to_s
89 |    "Result<#{control}>@"
90 |  end
91
92 |  # Public: To string.
93 |  public
94 |  def to_s
95 |    to_s
96 |  end
97
98 |  # Public: To JSON.
99 |  public
100 |  def to_json
101 |    {
102 |      control: control.to_s,
103 |      ignored: ignored.map(&:to_s),
104 |      mismatched: mismatched.map(&:to_s),
105 |      observations: observations.map(&:to_s)
106 |    }.to_json
107 |  end
108
109 |  # Public: From JSON.
110 |  public
111 |  def self.from_json(json)
112 |    result = new
113 |    result.control = Observation.from_json(json['control'])
114 |    result.ignored = json['ignored'].map { |o| Observation.from_json(o) }
115 |    result.mismatched = json['mismatched'].map { |o| Observation.from_json(o) }
116 |    result.observations = json['observations'].map { |o| Observation.from_json(o) }
117 |    result
118 |  end
119
120 |  # Public: To YAML.
121 |  public
122 |  def to_yaml
123 |    to_json.to_yaml
124 |  end
125
126 |  # Public: From YAML.
127 |  public
128 |  def self.from_yaml(yaml)
129 |    from_json(JSON.parse(yaml))
130 |  end
131
132 |  # Public: To XML.
133 |  public
134 |  def to_xml
135 |    to_json.to_xml
136 |  end
137
138 |  # Public: From XML.
139 |  public
140 |  def self.from_xml(xml)
141 |    from_json(JSON.parse(xml))
142 |  end
143
144 |  # Public: To CSV.
145 |  public
146 |  def to_csv
147 |    to_json.to_csv
148 |  end
149
150 |  # Public: From CSV.
151 |  public
152 |  def self.from_csv(csv)
153 |    from_json(JSON.parse(csv))
154 |  end
155
156 |  # Public: To JSON API.
157 |  public
158 |  def to_json_api
159 |    to_json
160 |  end
161
162 |  # Public: From JSON API.
163 |  public
164 |  def self.from_json_api(json)
165 |    from_json(JSON.parse(json))
166 |  end
167
168 |  # Public: To JSON API.
169 |  public
170 |  def to_json_api!
171 |    to_json
172 |  end
173
174 |  # Public: From JSON API.
175 |  public
176 |  def self.from_json_api!
177 |    from_json(JSON.parse!)
178 |  end
179
180 |  # Public: To JSON API.
181 |  public
182 |  def to_json_api!
183 |    to_json
184 |  end
185
186 |  # Public: From JSON API.
187 |  public
188 |  def self.from_json_api!
189 |    from_json(JSON.parse!)
190 |  end
191
192 |  # Public: To JSON API.
193 |  public
194 |  def to_json_api!
195 |    to_json
196 |  end
197
198 |  # Public: From JSON API.
199 |  public
200 |  def self.from_json_api!
201 |    from_json(JSON.parse!)
202 |  end
203
204 |  # Public: To JSON API.
205 |  public
206 |  def to_json_api!
207 |    to_json
208 |  end
209
210 |  # Public: From JSON API.
211 |  public
212 |  def self.from_json_api!
213 |    from_json(JSON.parse!)
214 |  end
215
216 |  # Public: To JSON API.
217 |  public
218 |  def to_json_api!
219 |    to_json
220 |  end
221
222 |  # Public: From JSON API.
223 |  public
224 |  def self.from_json_api!
225 |    from_json(JSON.parse!)
226 |  end
227
228 |  # Public: To JSON API.
229 |  public
230 |  def to_json_api!
231 |    to_json
232 |  end
233
234 |  # Public: From JSON API.
235 |  public
236 |  def self.from_json_api!
237 |    from_json(JSON.parse!)
238 |  end
239
240 |  # Public: To JSON API.
241 |  public
242 |  def to_json_api!
243 |    to_json
244 |  end
245
246 |  # Public: From JSON API.
247 |  public
248 |  def self.from_json_api!
249 |    from_json(JSON.parse!)
250 |  end
251
252 |  # Public: To JSON API.
253 |  public
254 |  def to_json_api!
255 |    to_json
256 |  end
257
258 |  # Public: From JSON API.
259 |  public
260 |  def self.from_json_api!
261 |    from_json(JSON.parse!)
262 |  end
263
264 |  # Public: To JSON API.
265 |  public
266 |  def to_json_api!
267 |    to_json
268 |  end
269
270 |  # Public: From JSON API.
271 |  public
272 |  def self.from_json_api!
273 |    from_json(JSON.parse!)
274 |  end
275
276 |  # Public: To JSON API.
277 |  public
278 |  def to_json_api!
279 |    to_json
280 |  end
281
282 |  # Public: From JSON API.
283 |  public
284 |  def self.from_json_api!
285 |    from_json(JSON.parse!)
286 |  end
287
288 |  # Public: To JSON API.
289 |  public
290 |  def to_json_api!
291 |    to_json
292 |  end
293
294 |  # Public: From JSON API.
295 |  public
296 |  def self.from_json_api!
297 |    from_json(JSON.parse!)
298 |  end
299
300 |  # Public: To JSON API.
301 |  public
302 |  def to_json_api!
303 |    to_json
304 |  end
305
306 |  # Public: From JSON API.
307 |  public
308 |  def self.from_json_api!
309 |    from_json(JSON.parse!)
310 |  end
311
312 |  # Public: To JSON API.
313 |  public
314 |  def to_json_api!
315 |    to_json
316 |  end
317
318 |  # Public: From JSON API.
319 |  public
320 |  def self.from_json_api!
321 |    from_json(JSON.parse!)
322 |  end
323
324 |  # Public: To JSON API.
325 |  public
326 |  def to_json_api!
327 |    to_json
328 |  end
329
330 |  # Public: From JSON API.
331 |  public
332 |  def self.from_json_api!
333 |    from_json(JSON.parse!)
334 |  end
335
336 |  # Public: To JSON API.
337 |  public
338 |  def to_json_api!
339 |    to_json
340 |  end
341
342 |  # Public: From JSON API.
343 |  public
344 |  def self.from_json_api!
345 |    from_json(JSON.parse!)
346 |  end
347
348 |  # Public: To JSON API.
349 |  public
350 |  def to_json_api!
351 |    to_json
352 |  end
353
354 |  # Public: From JSON API.
355 |  public
356 |  def self.from_json_api!
357 |    from_json(JSON.parse!)
358 |  end
359
360 |  # Public: To JSON API.
361 |  public
362 |  def to_json_api!
363 |    to_json
364 |  end
365
366 |  # Public: From JSON API.
367 |  public
368 |  def self.from_json_api!
369 |    from_json(JSON.parse!)
370 |  end
371
372 |  # Public: To JSON API.
373 |  public
374 |  def to_json_api!
375 |    to_json
376 |  end
377
378 |  # Public: From JSON API.
379 |  public
380 |  def self.from_json_api!
381 |    from_json(JSON.parse!)
382 |  end
383
384 |  # Public: To JSON API.
385 |  public
386 |  def to_json_api!
387 |    to_json
388 |  end
389
390 |  # Public: From JSON API.
391 |  public
392 |  def self.from_json_api!
393 |    from_json(JSON.parse!)
394 |  end
395
396 |  # Public: To JSON API.
397 |  public
398 |  def to_json_api!
399 |    to_json
400 |  end
401
402 |  # Public: From JSON API.
403 |  public
404 |  def self.from_json_api!
405 |    from_json(JSON.parse!)
406 |  end
407
408 |  # Public: To JSON API.
409 |  public
410 |  def to_json_api!
411 |    to_json
412 |  end
413
414 |  # Public: From JSON API.
415 |  public
416 |  def self.from_json_api!
417 |    from_json(JSON.parse!)
418 |  end
419
420 |  # Public: To JSON API.
421 |  public
422 |  def to_json_api!
423 |    to_json
424 |  end
425
426 |  # Public: From JSON API.
427 |  public
428 |  def self.from_json_api!
429 |    from_json(JSON.parse!)
430 |  end
431
432 |  # Public: To JSON API.
433 |  public
434 |  def to_json_api!
435 |    to_json
436 |  end
437
438 |  # Public: From JSON API.
439 |  public
440 |  def self.from_json_api!
441 |    from_json(JSON.parse!)
442 |  end
443
444 |  # Public: To JSON API.
445 |  public
446 |  def to_json_api!
447 |    to_json
448 |  end
449
450 |  # Public: From JSON API.
451 |  public
452 |  def self.from_json_api!
453 |    from_json(JSON.parse!)
454 |  end
455
456 |  # Public: To JSON API.
457 |  public
458 |  def to_json_api!
459 |    to_json
460 |  end
461
462 |  # Public: From JSON API.
463 |  public
464 |  def self.from_json_api!
465 |    from_json(JSON.parse!)
466 |  end
467
468 |  # Public: To JSON API.
469 |  public
470 |  def to_json_api!
471 |    to_json
472 |  end
473
474 |  # Public: From JSON API.
475 |  public
476 |  def self.from_json_api!
477 |    from_json(JSON.parse!)
478 |  end
479
480 |  # Public: To JSON API.
481 |  public
482 |  def to_json_api!
483 |    to_json
484 |  end
485
486 |  # Public: From JSON API.
487 |  public
488 |  def self.from_json_api!
489 |    from_json(JSON.parse!)
490 |  end
491
492 |  # Public: To JSON API.
493 |  public
494 |  def to_json_api!
495 |    to_json
496 |  end
497
498 |  # Public: From JSON API.
499 |  public
500 |  def self.from_json_api!
501 |    from_json(JSON.parse!)
502 |  end
503
504 |  # Public: To JSON API.
505 |  public
506 |  def to_json_api!
507 |    to_json
508 |  end
509
510 |  # Public: From JSON API.
511 |  public
512 |  def self.from_json_api!
513 |    from_json(JSON.parse!)
514 |  end
515
516 |  # Public: To JSON API.
517 |  public
518 |  def to_json_api!
519 |    to_json
520 |  end
521
522 |  # Public: From JSON API.
523 |  public
524 |  def self.from_json_api!
525 |    from_json(JSON.parse!)
526 |  end
527
528 |  # Public: To JSON API.
529 |  public
530 |  def to_json_api!
531 |    to_json
532 |  end
533
534 |  # Public: From JSON API.
535 |  public
536 |  def self.from_json_api!
537 |    from_json(JSON.parse!)
538 |  end
539
540 |  # Public: To JSON API.
541 |  public
542 |  def to_json_api!
543 |    to_json
544 |  end
545
546 |  # Public: From JSON API.
547 |  public
548 |  def self.from_json_api!
549 |    from_json(JSON.parse!)
550 |  end
551
552 |  # Public: To JSON API.
553 |  public
554 |  def to_json_api!
555 |    to_json
556 |  end
557
558 |  # Public: From JSON API.
559 |  public
560 |  def self.from_json_api!
561 |    from_json(JSON.parse!)
562 |  end
563
564 |  # Public: To JSON API.
565 |  public
566 |  def to_json_api!
567 |    to_json
568 |  end
569
570 |  # Public: From JSON API.
571 |  public
572 |  def self.from_json_api!
573 |    from_json(JSON.parse!)
574 |  end
575
576 |  # Public: To JSON API.
577 |  public
578 |  def to_json_api!
579 |    to_json
580 |  end
581
582 |  # Public: From JSON API.
583 |  public
584 |  def self.from_json_api!
585 |    from_json(JSON.parse!)
586 |  end
587
588 |  # Public: To JSON API.
589 |  public
590 |  def to_json_api!
591 |    to_json
592 |  end
593
594 |  # Public: From JSON API.
595 |  public
596 |  def self.from_json_api!
597 |    from_json(JSON.parse!)
598 |  end
599
599 |  # Public: To JSON API.
600 |  public
601 |  def to_json_api!
602 |    to_json
603 |  end
604
605 |  # Public: From JSON API.
606 |  public
607 |  def self.from_json_api!
608 |    from_json(JSON.parse!)
609 |  end
610
611 |  # Public: To JSON API.
612 |  public
613 |  def to_json_api!
614 |    to_json
615 |  end
616
617 |  # Public: From JSON API.
618 |  public
619 |  def self.from_json_api!
620 |    from_json(JSON.parse!)
621 |  end
622
623 |  # Public: To JSON API.
624 |  public
625 |  def to_json_api!
626 |    to_json
627 |  end
628
629 |  # Public: From JSON API.
630 |  public
631 |  def self.from_json_api!
632 |    from_json(JSON.parse!)
633 |  end
634
635 |  # Public: To JSON API.
636 |  public
637 |  def to_json_api!
638 |    to_json
639 |  end
640
641 |  # Public: From JSON API.
642 |  public
643 |  def self.from_json_api!
644 |    from_json(JSON.parse!)
645 |  end
646
647 |  # Public: To JSON API.
648 |  public
649 |  def to_json_api!
650 |    to_json
651 |  end
652
653 |  # Public: From JSON API.
654 |  public
655 |  def self.from_json_api!
656 |    from_json(JSON.parse!)
657 |  end
658
659 |  # Public: To JSON API.
660 |  public
661 |  def to_json_api!
662 |    to_json
663 |  end
664
665 |  # Public: From JSON API.
666 |  public
667 |  def self.from_json_api!
668 |    from_json(JSON.parse!)
669 |  end
670
671 |  # Public: To JSON API.
672 |  public
673 |  def to_json_api!
674 |    to_json
675 |  end
676
677 |  # Public: From JSON API.
678 |  public
679 |  def self.from_json_api!
680 |    from_json(JSON.parse!)
681 |  end
682
683 |  # Public: To JSON API.
684 |  public
685 |  def to_json_api!
686 |    to_json
687 |  end
688
689 |  # Public: From JSON API.
690 |  public
691 |  def self.from_json_api!
692 |    from_json(JSON.parse!)
693 |  end
694
695 |  # Public: To JSON API.
696 |  public
697 |  def to_json_api!
698 |    to_json
699 |  end
700
701 |  # Public: From JSON API.
702 |  public
703 |  def self.from_json_api!
704 |    from_json(JSON.parse!)
705 |  end
706
707 |  # Public: To JSON API.
708 |  public
709 |  def to_json_api!
710 |    to_json
711 |  end
712
713 |  # Public: From JSON API.
714 |  public
715 |  def self.from_json_api!
716 |    from_json(JSON.parse!)
717 |  end
718
719 |  # Public: To JSON API.
720 |  public
721 |  def to_json_api!
722 |    to_json
723 |  end
724
725 |  # Public: From JSON API.
726 |  public
727 |  def self.from_json_api!
728 |    from_json(JSON.parse!)
729 |  end
730
731 |  # Public: To JSON API.
732 |  public
733 |  def to_json_api!
734 |    to_json
735 |  end
736
737 |  # Public: From JSON API.
738 |  public
739 |  def self.from_json_api!
740 |    from_json(JSON.parse!)
741 |  end
742
743 |  # Public: To JSON API.
744 |  public
745 |  def to_json_api!
746 |    to_json
747 |  end
748
749 |  # Public: From JSON API.
750 |  public
751 |  def self.from_json_api!
752 |    from_json(JSON.parse!)
753 |  end
754
755 |  # Public: To JSON API.
756 |  public
757 |  def to_json_api!
758 |    to_json
759 |  end
760
761 |  # Public: From JSON API.
762 |  public
763 |  def self.from_json_api!
764 |    from_json(JSON.parse!)
765 |  end
766
767 |  # Public: To JSON API.
768 |  public
769 |  def to_json_api!
770 |    to_json
771 |  end
772
773 |  # Public: From JSON API.
774 |  public
775 |  def self.from_json_api!
776 |    from_json(JSON.parse!)
777 |  end
778
779 |  # Public: To JSON API.
780 |  public
781 |  def to_json_api!
782 |    to_json
783 |  end
784
785 |  # Public: From JSON API.
786 |  public
787 |  def self.from_json_api!
788 |    from_json(JSON.parse!)
789 |  end
790
791 |  # Public: To JSON API.
792 |  public
793 |  def to_json_api!
794 |    to_json
795 |  end
796
797 |  # Public: From JSON API.
798 |  public
799 |  def self.from_json_api!
800 |    from_json(JSON.parse!)
801 |  end
802
803 |  # Public: To JSON API.
804 |  public
805 |  def to_json_api!
806 |    to_json
807 |  end
808
809 |  # Public: From JSON API.
810 |  public
811 |  def self.from_json_api!
812 |    from_json(JSON.parse!)
813 |  end
814
815 |  # Public: To JSON API.
816 |  public
817 |  def to_json_api!
818 |    to_json
819 |  end
820
821 |  # Public: From JSON API.
822 |  public
823 |
```

#36

APP NATIVO

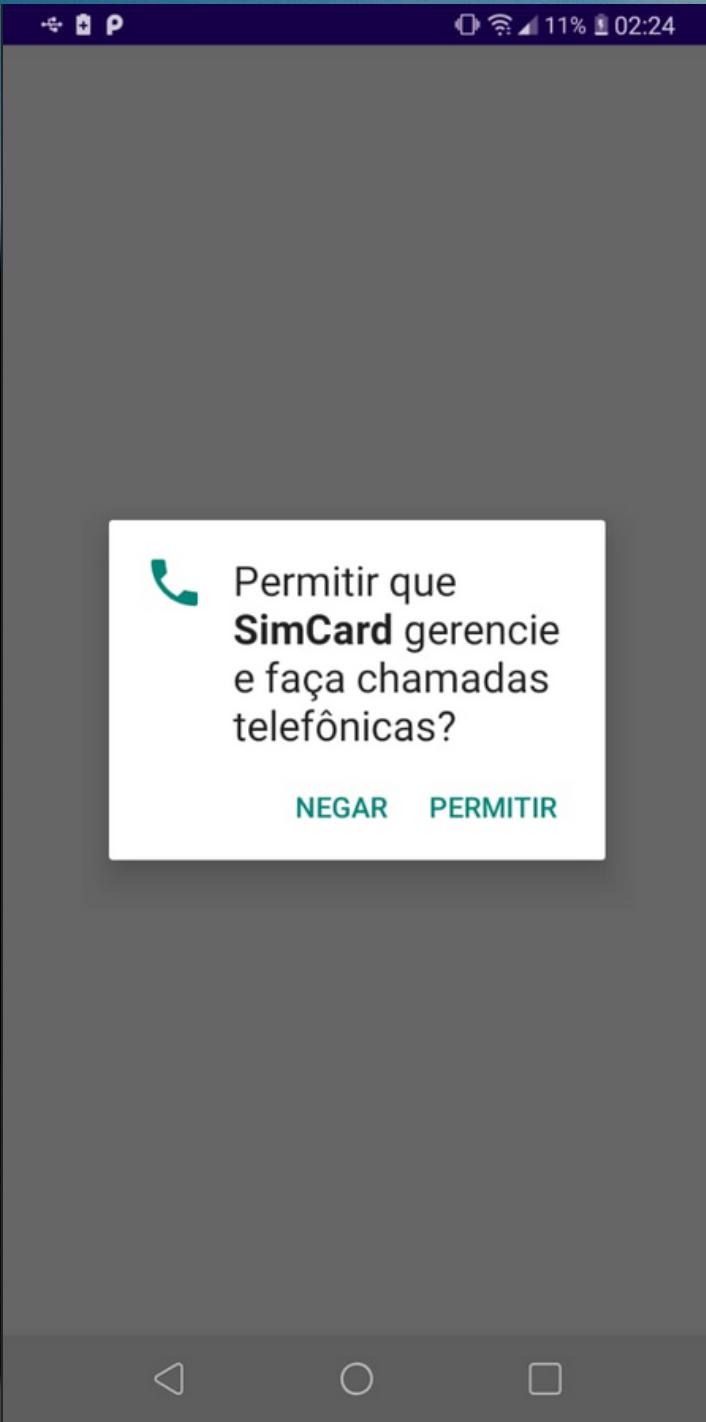


ANDREINA

Previsto: 12h
Realizado: 8h

#41

PERMISSÕES APP



```
private fun getPermission(): Boolean {  
    return (ContextCompat.checkSelfPermission(  
        this,  
        Manifest.permission.READ_PHONE_STATE  
    ) == PackageManager.PERMISSION_GRANTED)  
  
}  
  
private fun setPermission() {  
    val permissionsList = listOf<String>(  
        Manifest.permission.READ_PHONE_STATE  
    )  
    ActivityCompat.requestPermissions(this, permissionsList.toTypedArray(), 1)  
}
```



STHEFANYE

Previsto: 4h
Realizado: 2h

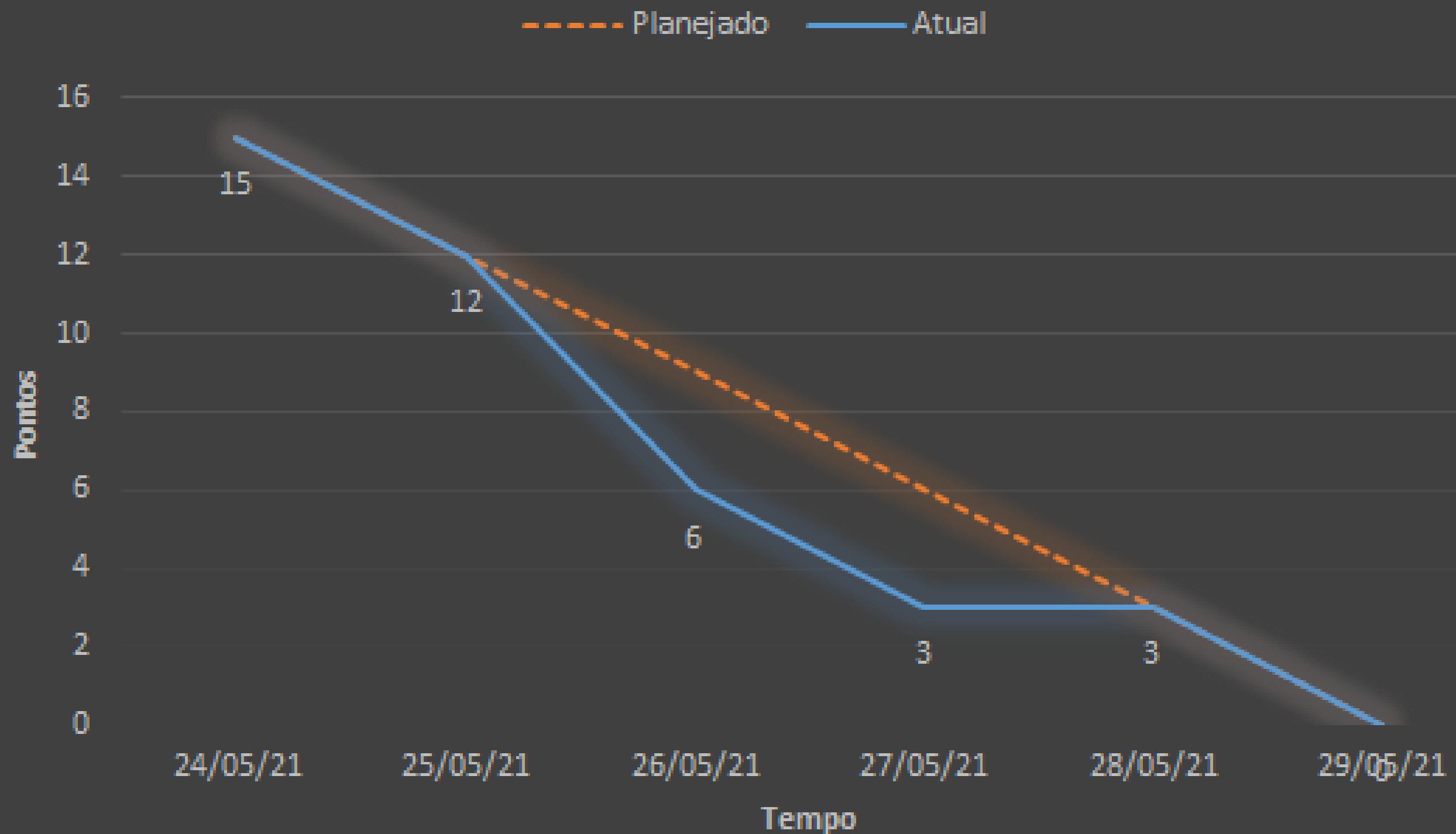
SPRINT BACKLOG

SPRINT BACKLOG				
ID	Tarefa	Previsto	Realizado	Prioridade
#31	Preparação do ambiente (2º máquina)	6:00	6:00	ENTREGUE
#32	Desenvolver Log de Inserção e Remoção	3:00	2:00	ENTREGUE
#33	Criar aquivo txt para receber dados de logs	6:00	6:00	ENTREGUE
#35	Notificar Troca de SIM Card	9:00	7:00	ENTREGUE
#36	Tornar o App Nativo	8:00	8:00	CANCELADO
#37	App em background	9:00	4:00	ENTREGUE
#41	Solicitar Permissões para o APP	4:00	2:00	ADIANTADO

Previsto	Realizado	Restante
45:00:00	35:00:00	10:00:00

BURNDOWN CHART

Burndown Chart - Sprint III



REFERÊNCIAS



PESQUISAS

- Logs Insert/Remove
- App Background



GIT LAB LINKS

- Repositório
- Backlog/Burndown