



YASSER JAVIER ORTIZ TRUJILLO

PROGRAMACIÓN
MÓVIL
COMUNICACIÓN

CONTACTO

- 0991592109
- El Guabo, El Oro,
Ecuador
- 2010guabo@gmail.com
- <https://ec.linkedin.com/in/yjot-dev>

EDUCACIÓN

Ingeniero de sistemas
UTMACH | 2015 – 2021

Magister en educación
mención gestión del
aprendizaje mediado por TIC
UTPL | 2023 – 2024

PERFIL

Desarrollador Android con sólidas competencias en comunicación efectiva, colaboración en equipo, toma de decisiones estratégicas y aprendizaje autónomo. Enfocado en aportar soluciones innovadoras y de calidad en el desarrollo de aplicaciones móviles.

EXPERIENCIA LABORAL

Programador Android

Independiente (**Del 22 de mayo del 2021 al presente**)

Censista y redactor de informes

INEC (**Del 3 de noviembre del 2022 al 4 de enero del 2023**)

Investigador y redactor sobre Inteligencia artificial y Ciencia de datos

Grupo de investigación en la FIC de la UTMACH (**Del 1 de noviembre del 2019 al 31 de enero del 2020**)

CERTIFICADOS ACADÉMICOS

Universidad Técnica De Machala

Ciclo de conferencias de Ingeniería de sistemas y tecnologías de la información
(del 07 al 09 de octubre del 2020)

GEG Corporation

Marketing Digital Facebook y WhatsApp Business
(11 de enero del 2019)

Universidad Técnica De Machala

Ciclo de conferencias de Ingeniería de sistemas y tecnologías de la información
(del 22 al 24 de agosto del 2018)

Gobierno Provincial Autónomo de El Oro

Curso de informática
(18 de abril del 2006)

LOGROS ACADÉMICOS

HABILIDADES



Adaptabilidad



Resolutiva



Proactividad



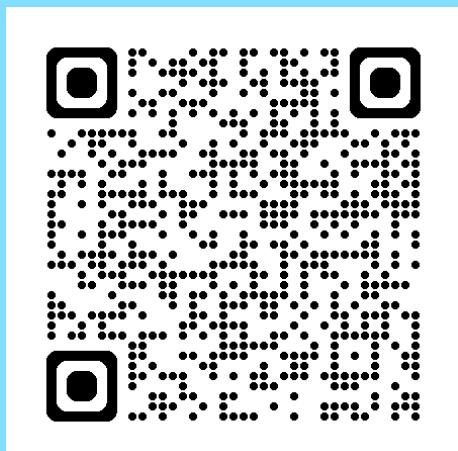
Comunicación

PROGRAMAS TÉCNICOS

Android Studios, Visual Studios Code, MySQL Workbench, Lucidchart y GitHub

MI SITIO WEB

Has clic [aquí](#) o escanea el código QR:



Universidad Técnica De
Machala

Mención de honor por
obtener el 3er lugar en
concurso de
Programación Básica
**(18 de septiembre del
2017)**

Universidad Técnica De
Machala

Mención de honor por
obtener el 3er lugar en
concurso de
Programación Básica
**(22 de diciembre del
2016)**



REPÚBLICA DEL ECUADOR
EN SU NOMBRE Y POR AUTORIDAD DE LA LEY



UNIVERSIDAD TÉCNICA DE MACHALA

Facultad De Ingeniería Civil

Confiere el Título de:

INGENIERO DE SISTEMAS

A:

ORTIZ TRUJILLO YASSER JAVIER

Por haber cumplido los requisitos legales, estatutarios y reglamentarios; por tanto, se le reconocerán los derechos y guardarán las consideraciones que le corresponden a su formación de TERCER NIVEL; modalidad de estudio PRESENCIAL

Machala, 14 de abril de 2022

Ing. Freddy Leonardo Espinoza Urgiles, MEng.

DECANO



Abg. Angel Lenin Erazo Bermico, MEng.

SECRETARIO ABOGADO



Ing. Cesar Javier Quezada Abad, PhD

RECTOR



REFRENDOADO N° 59331

Machala, 17 de mayo de 2022

Nº 016835

Abg. Gerardo Gabriel Fernandez Valdivia, MEng.

SECRETARIO GENERAL





LA REPÚBLICA DEL ECUADOR
EN SU NOMBRE Y POR AUTORIDAD DE LA LEY
LA UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA,
A TRAVÉS DEL PROGRAMA DE MAESTRÍA EN EDUCACIÓN

Hacen notorio que:

YASSER JAVIER ORTIZ TRUJILLO

ha cumplido con los requisitos exigidos por la Ley de Educación Superior, el Estatuto y la reglamentación universitaria, y que, en virtud de la aprobación que obtuvo y de la promesa legal que prestó, se le confiere el título de

MAGISTER EN EDUCACION, MENCION EN GESTION DEL APRENDIZAJE MEDIADO POR TIC

que le habilita para ejercer las funciones inherentes a la profesión.

Dado, firmado y sellado por la Directora de Programa, el Secretario General y el Rector de la Universidad, en Loja, a los cinco días del mes de abril del año 2024.

Dra. Verónica Patricia Sánchez Burmeo

DIRECTORA DE PROGRAMA



Dr. Gabriel Ulpiano García Torres

SECRETARIO GENERAL

Dr. Santiago Acosta Aide

RECTOR



RECTORADO DE LA UNIVERSIDAD

Refrendado

Loja, 5 de abril de 2024

LOGROS ACADÉMICOS



UNIVERSIDAD TÉCNICA DE MACHALA
D.L. NO. 69-04 DE 14 DE ABRIL DE 1969
— UNIDAD ACADÉMICA DE INGENIERÍA CIVIL —

Otorga la presente

MENCIÓN DE HONOR

A:

YASSER JAVIER ORTIZ TRUJILLO

Por haber obtenido el **TERCER LUGAR** en el Concurso de **Programación Básica**, organizado con motivo de los 44 años de Unidad Académica de Ingeniería Civil.

Machala, 18 de septiembre de 2017



Ing. Magno Armijos Molina, Mgs.
DECANO UAIC



Ing. Freddy Espinoza Urgilés, Mgs.
SUBDECANO UAIC



Ab. Lenín Erazo Bermeo, Mg.Sc.
SECRETARIO - ABOGADO



UNIVERSIDAD TÉCNICA DE MACHALA

D.L. No. 69-04 DE 14 DE ABRIL DE 1969

— UNIDAD ACADÉMICA DE INGENIERÍA CIVIL —

Otorga la presente

MENCIÓN DE HONOR

A:

YASSER JAVIER ORTIZ TRUJILLO

Por haber obtenido el **Tercer Lugar** en el Concurso de Programación Básica de la Carrera de Ingeniería de Sistemas - Unidad Académica de Ingeniería Civil en el año 2016-2017.

Machala, 22 de diciembre de 2016

Ing. Cesar Quezada Abad, MBA.
RECTOR UTMACH

Ing. Civ. Magno Armijos Molina
DECANO UAIC

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL
UNIVERSIDAD TÉCNICA DE MACHALA
1969
DECANATO
Machala - El Oro
SUBDECANATO
Machala - El Oro

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL
UNIVERSIDAD TÉCNICA DE MACHALA
1969
DECANATO
Machala - El Oro
SUBDECANATO
Machala - El Oro

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL
UNIVERSIDAD TÉCNICA DE MACHALA
1969
SECRETARIO ABOGADO UAIC
Machala - El Oro
GENERAL

CERTIFICADOS ACADÉMICOS



Semana de la ciencia



PreCongreso

Ciclo de Conferencias de Ingeniería de Sistemas y Tecnologías de la Información

Certifica a:

YASSER JAVIER ORTIZ TRUJILLO

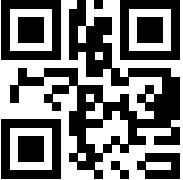
Por haber participado en calidad de **ASISTENTE** en el ciclo de conferencias organizado en el marco de la Semana de la Ciencia, con una duración de 24 horas, del 7 al 9 de octubre de 2020.

Machala, 9 de octubre de 2020


César Quezada Abad
Rector UTMACH


Luis Brito Gaona
Director de Investigación




Amarilis Borja Herrera
Vicerrectora Académica

19792



Certified Agency



CERTIFICADO

Otorgado a:

YASSER JAVIER ORTÍZ TRUJILLO

Como testimonio de su participación activa en calidad de **ASISTENTE** a la conferencia:
“Marketing Digital Facebook y WhatsApp Business”

Superando los objetivos trazados satisfactoriamente, es todo cuanto certificamos en honor a la verdad
para fines que convengan al interesado.

EMITIDO EL 11 DE ENERO DEL 2019



Duración:
4 sesiones / 2 conferencias
Carga:
40 horas académicas
País:
Ecuador
Vigencia:
Indefinida



PROGRAMA DE PARTNERSHIP EDUCACIONAL



UNIVERSIDAD TÉCNICA DE MACHALA

D.L. NO. 69-04 DE 14 DE ABRIL DE 1969

— CENTRO DE INVESTIGACIÓN —

Otorga el presente

CERTIFICADO

A: YASSER JAVIER ORTIZ TRUJILLO

Por haber **ASISTIDO** al Ciclo de Conferencias de Ingeniería de Sistemas y Tecnologías de la Información, realizado del 22 al 24 de agosto de 2018.

Machala, 24 de agosto de 2018

Ing. César Quezada Abad, PhD.
RECTOR UTMACH



Ing. Amarilis Bonja Herrera, PhD.
VICERRECTORA ACADÉMICA



Tomás Fontaines Ruiz, PhD.
DIRECTOR DE INVESTIGACIÓN





GOBIERNO PROVINCIAL AUTÓNOMO
DE EL ORO

PATRONATO DE ACCIÓN
SOCIAL

Confiere el presente:

Certificado

A ***YASSER JAVIER ORTIZ TRUJILLO***

Por su asistencia y aprobación al CURSO DE INFORMATICA, dictado en el Aula Móvil con una duración de

20 horas

Guabo 18 Abril 2006



Ing. Agr. Montgomery Sánchez Reyes
PREFECTO PROVINCIAL DE EL ORO



Sra. Gloria Ordóñez de Sánchez
PRESIDENTA DEL PATRONATO DE ACCIÓN SOCIAL



Eyel Chamaidán Guevara
COORDINADOR

EXPERIENCIA LABORAL

File Edit View Navigate Code Refactor Build Run Tools Git Window Help

AR Accident Reporter master INFINIX Infinix X6528 app Running Devices

Android app manifests kotlin+java com.yjotdev.accidentreporter application composable mvvm navigation theme domain entity port usecase utils infrastructure adapter di repositories MainActivity MainApplication com.yjotdev.accidentreporter (android) com.yjotdev.accidentreporter (test) java (generated) res res (generated) Gradle Scripts

AppViewModel.kt

```
1 package com.yjotdev.accidentreporter.a
2
3 > import ...
4
5 @HiltViewModel
6 class AppViewModel @Inject constructor(
7     private val selectReportUseCase: SelectReportUseCase,
8     private val insertReportUseCase: InsertReportUseCase,
9     private val updateReportUseCase: UpdateReportUseCase,
10    private val deleteReportUseCase: DeleteReportUseCase,
11    private val createTokenUseCase: CreateTokenUseCase,
12    private val getTokenUseCase: GetTokenUseCase
13 ): ViewModel() {
14
15     private val _uiState = MutableStateFlow(AppModel())
16     val uiState: StateFlow<AppModel> = _uiState.asState()
17
18     override fun onCleared() {
19         super.onCleared()
20         resetViewModel()
21     }
22
23     private fun resetViewModel() {
24         _uiState.value = AppModel()
25     }
26
27     fun insertReport(report: Report) {
28         insertReportUseCase.insertReport(report)
29     }
30
31     fun updateReport(report: Report) {
32         updateReportUseCase.updateReport(report)
33     }
34
35     fun deleteReport(id: String) {
36         deleteReportUseCase.deleteReport(id)
37     }
38
39     fun getReport(id: String): Report? {
40         return getTokenUseCase.getReport(id)
41     }
42
43     fun insertToken(token: Token) {
44         createTokenUseCase.insertToken(token)
45     }
46
47     fun updateToken(token: Token) {
48         updateTokenUseCase.updateToken(token)
49     }
50
51     fun deleteToken(id: String) {
52         deleteTokenUseCase.deleteToken(id)
53     }
54
55     fun getToken(): Token? {
56         return getTokenUseCase.getToken()
57     }
58
59     fun insertReportUseCase(): SelectReportUseCase {
60         return selectReportUseCase
61     }
62
63     fun insertReportUseCase(): InsertReportUseCase {
64         return insertReportUseCase
65     }
66
67     fun updateReportUseCase(): UpdateReportUseCase {
68         return updateReportUseCase
69     }
70
71     fun deleteReportUseCase(): DeleteReportUseCase {
72         return deleteReportUseCase
73     }
74
75     fun createTokenUseCase(): CreateTokenUseCase {
76         return createTokenUseCase
77     }
78
79     fun getTokenUseCase(): GetTokenUseCase {
80         return getTokenUseCase
81     }
82
83     fun insertTokenUseCase(): InsertTokenUseCase {
84         return insertTokenUseCase
85     }
86
87     fun updateTokenUseCase(): UpdateTokenUseCase {
88         return updateTokenUseCase
89     }
90
91     fun deleteTokenUseCase(): DeleteTokenUseCase {
92         return deleteTokenUseCase
93     }
94
95     fun insertReportUseCase(): SelectReportUseCase {
96         return selectReportUseCase
97     }
98
99     fun insertReportUseCase(): InsertReportUseCase {
100    return insertReportUseCase
101}
102
103     fun updateReportUseCase(): UpdateReportUseCase {
104    return updateReportUseCase
105}
106
107     fun deleteReportUseCase(): DeleteReportUseCase {
108    return deleteReportUseCase
109}
110
111     fun createTokenUseCase(): CreateTokenUseCase {
112    return createTokenUseCase
113}
114
115     fun getTokenUseCase(): GetTokenUseCase {
116    return getTokenUseCase
117}
118
119     fun insertTokenUseCase(): InsertTokenUseCase {
120    return insertTokenUseCase
121}
122
123     fun updateTokenUseCase(): UpdateTokenUseCase {
124    return updateTokenUseCase
125}
126
127     fun deleteTokenUseCase(): DeleteTokenUseCase {
128    return deleteTokenUseCase
129}
130
131     fun insertReportUseCase(): SelectReportUseCase {
132    return selectReportUseCase
133}
134
135     fun insertReportUseCase(): InsertReportUseCase {
136    return insertReportUseCase
137}
138
139     fun updateReportUseCase(): UpdateReportUseCase {
140    return updateReportUseCase
141}
142
143     fun deleteReportUseCase(): DeleteReportUseCase {
144    return deleteReportUseCase
145}
146
147     fun createTokenUseCase(): CreateTokenUseCase {
148    return createTokenUseCase
149}
150
151     fun getTokenUseCase(): GetTokenUseCase {
152    return getTokenUseCase
153}
154
155     fun insertTokenUseCase(): InsertTokenUseCase {
156    return insertTokenUseCase
157}
158
159     fun updateTokenUseCase(): UpdateTokenUseCase {
160    return updateTokenUseCase
161}
162
163     fun deleteTokenUseCase(): DeleteTokenUseCase {
164    return deleteTokenUseCase
165}
166
167     fun insertReportUseCase(): SelectReportUseCase {
168    return selectReportUseCase
169}
170
171     fun insertReportUseCase(): InsertReportUseCase {
172    return insertReportUseCase
173}
174
175     fun updateReportUseCase(): UpdateReportUseCase {
176    return updateReportUseCase
177}
178
179     fun deleteReportUseCase(): DeleteReportUseCase {
180    return deleteReportUseCase
181}
182
183     fun createTokenUseCase(): CreateTokenUseCase {
184    return createTokenUseCase
185}
186
187     fun getTokenUseCase(): GetTokenUseCase {
188    return getTokenUseCase
189}
190
191     fun insertTokenUseCase(): InsertTokenUseCase {
192    return insertTokenUseCase
193}
194
195     fun updateTokenUseCase(): UpdateTokenUseCase {
196    return updateTokenUseCase
197}
198
199     fun deleteTokenUseCase(): DeleteTokenUseCase {
200    return deleteTokenUseCase
201}
202
203     fun insertReportUseCase(): SelectReportUseCase {
204    return selectReportUseCase
205}
206
207     fun insertReportUseCase(): InsertReportUseCase {
208    return insertReportUseCase
209}
210
211     fun updateReportUseCase(): UpdateReportUseCase {
212    return updateReportUseCase
213}
214
215     fun deleteReportUseCase(): DeleteReportUseCase {
216    return deleteReportUseCase
217}
218
219     fun createTokenUseCase(): CreateTokenUseCase {
220    return createTokenUseCase
221}
222
223     fun getTokenUseCase(): GetTokenUseCase {
224    return getTokenUseCase
225}
226
227     fun insertTokenUseCase(): InsertTokenUseCase {
228    return insertTokenUseCase
229}
230
231     fun updateTokenUseCase(): UpdateTokenUseCase {
232    return updateTokenUseCase
233}
234
235     fun deleteTokenUseCase(): DeleteTokenUseCase {
236    return deleteTokenUseCase
237}
238
239     fun insertReportUseCase(): SelectReportUseCase {
240    return selectReportUseCase
241}
242
243     fun insertReportUseCase(): InsertReportUseCase {
244    return insertReportUseCase
245}
246
247     fun updateReportUseCase(): UpdateReportUseCase {
248    return updateReportUseCase
249}
250
251     fun deleteReportUseCase(): DeleteReportUseCase {
252    return deleteReportUseCase
253}
254
255     fun createTokenUseCase(): CreateTokenUseCase {
256    return createTokenUseCase
257}
258
259     fun getTokenUseCase(): GetTokenUseCase {
260    return getTokenUseCase
261}
262
263     fun insertTokenUseCase(): InsertTokenUseCase {
264    return insertTokenUseCase
265}
266
267     fun updateTokenUseCase(): UpdateTokenUseCase {
268    return updateTokenUseCase
269}
270
271     fun deleteTokenUseCase(): DeleteTokenUseCase {
272    return deleteTokenUseCase
273}
274
275     fun insertReportUseCase(): SelectReportUseCase {
276    return selectReportUseCase
277}
278
279     fun insertReportUseCase(): InsertReportUseCase {
280    return insertReportUseCase
281}
282
283     fun updateReportUseCase(): UpdateReportUseCase {
284    return updateReportUseCase
285}
286
287     fun deleteReportUseCase(): DeleteReportUseCase {
288    return deleteReportUseCase
289}
290
291     fun createTokenUseCase(): CreateTokenUseCase {
292    return createTokenUseCase
293}
294
295     fun getTokenUseCase(): GetTokenUseCase {
296    return getTokenUseCase
297}
298
299     fun insertTokenUseCase(): InsertTokenUseCase {
300    return insertTokenUseCase
301}
302
303     fun updateTokenUseCase(): UpdateTokenUseCase {
304    return updateTokenUseCase
305}
306
307     fun deleteTokenUseCase(): DeleteTokenUseCase {
308    return deleteTokenUseCase
309}
310
311     fun insertReportUseCase(): SelectReportUseCase {
312    return selectReportUseCase
313}
314
315     fun insertReportUseCase(): InsertReportUseCase {
316    return insertReportUseCase
317}
318
319     fun updateReportUseCase(): UpdateReportUseCase {
320    return updateReportUseCase
321}
322
323     fun deleteReportUseCase(): DeleteReportUseCase {
324    return deleteReportUseCase
325}
326
327     fun createTokenUseCase(): CreateTokenUseCase {
328    return createTokenUseCase
329}
330
331     fun getTokenUseCase(): GetTokenUseCase {
332    return getTokenUseCase
333}
334
335     fun insertTokenUseCase(): InsertTokenUseCase {
336    return insertTokenUseCase
337}
338
339     fun updateTokenUseCase(): UpdateTokenUseCase {
340    return updateTokenUseCase
341}
342
343     fun deleteTokenUseCase(): DeleteTokenUseCase {
344    return deleteTokenUseCase
345}
346
347     fun insertReportUseCase(): SelectReportUseCase {
348    return selectReportUseCase
349}
350
351     fun insertReportUseCase(): InsertReportUseCase {
352    return insertReportUseCase
353}
354
355     fun updateReportUseCase(): UpdateReportUseCase {
356    return updateReportUseCase
357}
358
359     fun deleteReportUseCase(): DeleteReportUseCase {
360    return deleteReportUseCase
361}
362
363     fun createTokenUseCase(): CreateTokenUseCase {
364    return createTokenUseCase
365}
366
367     fun getTokenUseCase(): GetTokenUseCase {
368    return getTokenUseCase
369}
370
371     fun insertTokenUseCase(): InsertTokenUseCase {
372    return insertTokenUseCase
373}
374
375     fun updateTokenUseCase(): UpdateTokenUseCase {
376    return updateTokenUseCase
377}
378
379     fun deleteTokenUseCase(): DeleteTokenUseCase {
380    return deleteTokenUseCase
381}
382
383     fun insertReportUseCase(): SelectReportUseCase {
384    return selectReportUseCase
385}
386
387     fun insertReportUseCase(): InsertReportUseCase {
388    return insertReportUseCase
389}
390
391     fun updateReportUseCase(): UpdateReportUseCase {
392    return updateReportUseCase
393}
394
395     fun deleteReportUseCase(): DeleteReportUseCase {
396    return deleteReportUseCase
397}
398
399     fun createTokenUseCase(): CreateTokenUseCase {
400    return createTokenUseCase
401}
402
403     fun getTokenUseCase(): GetTokenUseCase {
404    return getTokenUseCase
405}
406
407     fun insertTokenUseCase(): InsertTokenUseCase {
408    return insertTokenUseCase
409}
410
411     fun updateTokenUseCase(): UpdateTokenUseCase {
412    return updateTokenUseCase
413}
414
415     fun deleteTokenUseCase(): DeleteTokenUseCase {
416    return deleteTokenUseCase
417}
418
419     fun insertReportUseCase(): SelectReportUseCase {
420    return selectReportUseCase
421}
422
423     fun insertReportUseCase(): InsertReportUseCase {
424    return insertReportUseCase
425}
426
427     fun updateReportUseCase(): UpdateReportUseCase {
428    return updateReportUseCase
429}
430
431     fun deleteReportUseCase(): DeleteReportUseCase {
432    return deleteReportUseCase
433}
434
435     fun createTokenUseCase(): CreateTokenUseCase {
436    return createTokenUseCase
437}
438
439     fun getTokenUseCase(): GetTokenUseCase {
440    return getTokenUseCase
441}
442
443     fun insertTokenUseCase(): InsertTokenUseCase {
444    return insertTokenUseCase
445}
446
447     fun updateTokenUseCase(): UpdateTokenUseCase {
448    return updateTokenUseCase
449}
450
451     fun deleteTokenUseCase(): DeleteTokenUseCase {
452    return deleteTokenUseCase
453}
454
455     fun insertReportUseCase(): SelectReportUseCase {
456    return selectReportUseCase
457}
458
459     fun insertReportUseCase(): InsertReportUseCase {
460    return insertReportUseCase
461}
462
463     fun updateReportUseCase(): UpdateReportUseCase {
464    return updateReportUseCase
465}
466
467     fun deleteReportUseCase(): DeleteReportUseCase {
468    return deleteReportUseCase
469}
470
471     fun createTokenUseCase(): CreateTokenUseCase {
472    return createTokenUseCase
473}
474
475     fun getTokenUseCase(): GetTokenUseCase {
476    return getTokenUseCase
477}
478
479     fun insertTokenUseCase(): InsertTokenUseCase {
480    return insertTokenUseCase
481}
482
483     fun updateTokenUseCase(): UpdateTokenUseCase {
484    return updateTokenUseCase
485}
486
487     fun deleteTokenUseCase(): DeleteTokenUseCase {
488    return deleteTokenUseCase
489}
490
491     fun insertReportUseCase(): SelectReportUseCase {
492    return selectReportUseCase
493}
494
495     fun insertReportUseCase(): InsertReportUseCase {
496    return insertReportUseCase
497}
498
499     fun updateReportUseCase(): UpdateReportUseCase {
500    return updateReportUseCase
501}
502
503     fun deleteReportUseCase(): DeleteReportUseCase {
504    return deleteReportUseCase
505}
506
507     fun createTokenUseCase(): CreateTokenUseCase {
508    return createTokenUseCase
509}
510
511     fun getTokenUseCase(): GetTokenUseCase {
512    return getTokenUseCase
513}
514
515     fun insertTokenUseCase(): InsertTokenUseCase {
516    return insertTokenUseCase
517}
518
519     fun updateTokenUseCase(): UpdateTokenUseCase {
520    return updateTokenUseCase
521}
522
523     fun deleteTokenUseCase(): DeleteTokenUseCase {
524    return deleteTokenUseCase
525}
526
527     fun insertReportUseCase(): SelectReportUseCase {
528    return selectReportUseCase
529}
530
531     fun insertReportUseCase(): InsertReportUseCase {
532    return insertReportUseCase
533}
534
535     fun updateReportUseCase(): UpdateReportUseCase {
536    return updateReportUseCase
537}
538
539     fun deleteReportUseCase(): DeleteReportUseCase {
540    return deleteReportUseCase
541}
542
543     fun createTokenUseCase(): CreateTokenUseCase {
544    return createTokenUseCase
545}
546
547     fun getTokenUseCase(): GetTokenUseCase {
548    return getTokenUseCase
549}
550
551     fun insertTokenUseCase(): InsertTokenUseCase {
552    return insertTokenUseCase
553}
554
555     fun updateTokenUseCase(): UpdateTokenUseCase {
556    return updateTokenUseCase
557}
558
559     fun deleteTokenUseCase(): DeleteTokenUseCase {
560    return deleteTokenUseCase
561}
562
563     fun insertReportUseCase(): SelectReportUseCase {
564    return selectReportUseCase
565}
566
567     fun insertReportUseCase(): InsertReportUseCase {
568    return insertReportUseCase
569}
570
571     fun updateReportUseCase(): UpdateReportUseCase {
572    return updateReportUseCase
573}
574
575     fun deleteReportUseCase(): DeleteReportUseCase {
576    return deleteReportUseCase
577}
578
579     fun createTokenUseCase(): CreateTokenUseCase {
580    return createTokenUseCase
581}
582
583     fun getTokenUseCase(): GetTokenUseCase {
584    return getTokenUseCase
585}
586
587     fun insertTokenUseCase(): InsertTokenUseCase {
588    return insertTokenUseCase
589}
590
591     fun updateTokenUseCase(): UpdateTokenUseCase {
592    return updateTokenUseCase
593}
594
595     fun deleteTokenUseCase(): DeleteTokenUseCase {
596    return deleteTokenUseCase
597}
598
599     fun insertReportUseCase(): SelectReportUseCase {
600    return selectReportUseCase
601}
602
603     fun insertReportUseCase(): InsertReportUseCase {
604    return insertReportUseCase
605}
606
607     fun updateReportUseCase(): UpdateReportUseCase {
608    return updateReportUseCase
609}
610
611     fun deleteReportUseCase(): DeleteReportUseCase {
612    return deleteReportUseCase
613}
614
615     fun createTokenUseCase(): CreateTokenUseCase {
616    return createTokenUseCase
617}
618
619     fun getTokenUseCase(): GetTokenUseCase {
620    return getTokenUseCase
621}
622
623     fun insertTokenUseCase(): InsertTokenUseCase {
624    return insertTokenUseCase
625}
626
627     fun updateTokenUseCase(): UpdateTokenUseCase {
628    return updateTokenUseCase
629}
630
631     fun deleteTokenUseCase(): DeleteTokenUseCase {
632    return deleteTokenUseCase
633}
634
635     fun insertReportUseCase(): SelectReportUseCase {
636    return selectReportUseCase
637}
638
639     fun insertReportUseCase(): InsertReportUseCase {
640    return insertReportUseCase
641}
642
643     fun updateReportUseCase(): UpdateReportUseCase {
644    return updateReportUseCase
645}
646
647     fun deleteReportUseCase(): DeleteReportUseCase {
648    return deleteReportUseCase
649}
650
651     fun createTokenUseCase(): CreateTokenUseCase {
652    return createTokenUseCase
653}
654
655     fun getTokenUseCase(): GetTokenUseCase {
656    return getTokenUseCase
657}
658
659     fun insertTokenUseCase(): InsertTokenUseCase {
660    return insertTokenUseCase
661}
662
663     fun updateTokenUseCase(): UpdateTokenUseCase {
664    return updateTokenUseCase
665}
666
667     fun deleteTokenUseCase(): DeleteTokenUseCase {
668    return deleteTokenUseCase
669}
670
671     fun insertReportUseCase(): SelectReportUseCase {
672    return selectReportUseCase
673}
674
675     fun insertReportUseCase(): InsertReportUseCase {
676    return insertReportUseCase
677}
678
679     fun updateReportUseCase(): UpdateReportUseCase {
680    return updateReportUseCase
681}
682
683     fun deleteReportUseCase(): DeleteReportUseCase {
684    return deleteReportUseCase
685}
686
687     fun createTokenUseCase(): CreateTokenUseCase {
688    return createTokenUseCase
689}
690
691     fun getTokenUseCase(): GetTokenUseCase {
692    return getTokenUseCase
693}
694
695     fun insertTokenUseCase(): InsertTokenUseCase {
696    return insertTokenUseCase
697}
698
699     fun updateTokenUseCase(): UpdateTokenUseCase {
700    return updateTokenUseCase
701}
702
703     fun deleteTokenUseCase(): DeleteTokenUseCase {
704    return deleteTokenUseCase
705}
706
707     fun insertReportUseCase(): SelectReportUseCase {
708    return selectReportUseCase
709}
710
711     fun insertReportUseCase(): InsertReportUseCase {
712    return insertReportUseCase
713}
714
715     fun updateReportUseCase(): UpdateReportUseCase {
716    return updateReportUseCase
717}
718
719     fun deleteReportUseCase(): DeleteReportUseCase {
720    return deleteReportUseCase
721}
722
723     fun createTokenUseCase(): CreateTokenUseCase {
724    return createTokenUseCase
725}
726
727     fun getTokenUseCase(): GetTokenUseCase {
728    return getTokenUseCase
729}
730
731     fun insertTokenUseCase(): InsertTokenUseCase {
732    return insertTokenUseCase
733}
734
735     fun updateTokenUseCase(): UpdateTokenUseCase {
736    return updateTokenUseCase
737}
738
739     fun deleteTokenUseCase(): DeleteTokenUseCase {
740    return deleteTokenUseCase
741}
742
743     fun insertReportUseCase(): SelectReportUseCase {
744    return selectReportUseCase
745}
746
747     fun insertReportUseCase(): InsertReportUseCase {
748    return insertReportUseCase
749}
750
751     fun updateReportUseCase(): UpdateReportUseCase {
752    return updateReportUseCase
753}
754
755     fun deleteReportUseCase(): DeleteReportUseCase {
756    return deleteReportUseCase
757}
758
759     fun createTokenUseCase(): CreateTokenUseCase {
760    return createTokenUseCase
761}
762
763     fun getTokenUseCase(): GetTokenUseCase {
764    return getTokenUseCase
765}
766
767     fun insertTokenUseCase(): InsertTokenUseCase {
768    return insertTokenUseCase
769}
770
771     fun updateTokenUseCase(): UpdateTokenUseCase {
772    return updateTokenUseCase
773}
774
775     fun deleteTokenUseCase(): DeleteTokenUseCase {
776    return deleteTokenUseCase
777}
778
779     fun insertReportUseCase(): SelectReportUseCase {
780    return selectReportUseCase
781}
782
783     fun insertReportUseCase(): InsertReportUseCase {
784    return insertReportUseCase
785}
786
787     fun updateReportUseCase(): UpdateReportUseCase {
788    return updateReportUseCase
789}
790
791     fun deleteReportUseCase(): DeleteReportUseCase {
792    return deleteReportUseCase
793}
794
795     fun createTokenUseCase(): CreateTokenUseCase {
796    return createTokenUseCase
797}
798
799     fun getTokenUseCase(): GetTokenUseCase {
800    return getTokenUseCase
801}
802
803     fun insertTokenUseCase(): InsertTokenUseCase {
804    return insertTokenUseCase
805}
806
807     fun updateTokenUseCase(): UpdateTokenUseCase {
808    return updateTokenUseCase
809}
810
811     fun deleteTokenUseCase(): DeleteTokenUseCase {
812    return deleteTokenUseCase
813}
814
815     fun insertReportUseCase(): SelectReportUseCase {
816    return selectReportUseCase
817}
818
819     fun insertReportUseCase(): InsertReportUseCase {
820    return insertReportUseCase
821}
822
823     fun updateReportUseCase(): UpdateReportUseCase {
824    return updateReportUseCase
825}
826
827     fun deleteReportUseCase(): DeleteReportUseCase {
828    return deleteReportUseCase
829}
830
831     fun createTokenUseCase(): CreateTokenUseCase {
832    return createTokenUseCase
833}
834
835     fun getTokenUseCase(): GetTokenUseCase {
836    return getTokenUseCase
837}
838
839     fun insertTokenUseCase(): InsertTokenUseCase {
840    return insertTokenUseCase
841}
842
843     fun updateTokenUseCase(): UpdateTokenUseCase {
844    return updateTokenUseCase
845}
846
847     fun deleteTokenUseCase(): DeleteTokenUseCase {
848    return deleteTokenUseCase
849}
850
851     fun insertReportUseCase(): SelectReportUseCase {
852    return selectReportUseCase
853}
854
855     fun insertReportUseCase(): InsertReportUseCase {
856    return insertReportUseCase
857}
858
859     fun updateReportUseCase(): UpdateReportUseCase {
860    return updateReportUseCase
861}
862
863     fun deleteReportUseCase(): DeleteReportUseCase {
864    return deleteReportUseCase
865}
866
867     fun createTokenUseCase(): CreateTokenUseCase {
868    return createTokenUseCase
869}
870
871     fun getTokenUseCase(): GetTokenUseCase {
872    return getTokenUseCase
873}
874
875     fun insertTokenUseCase(): InsertTokenUseCase {
876    return insertTokenUseCase
877}
878
879     fun updateTokenUseCase(): UpdateTokenUseCase {
880    return updateTokenUseCase
881}
882
883     fun deleteTokenUseCase(): DeleteTokenUseCase {
884    return deleteTokenUseCase
885}
886
887     fun insertReportUseCase(): SelectReportUseCase {
888    return selectReportUseCase
889}
890
891     fun insertReportUseCase(): InsertReportUseCase {
892    return insertReportUseCase
893}
894
895     fun updateReportUseCase(): UpdateReportUseCase {
896    return updateReportUseCase
897}
898
899     fun deleteReportUseCase(): DeleteReportUseCase {
900    return deleteReportUseCase
901}
902
903     fun createTokenUseCase(): CreateTokenUseCase {
904    return createTokenUseCase
905}
906
907     fun getTokenUseCase(): GetTokenUseCase {
908    return getTokenUseCase
909}
910
911     fun insertTokenUseCase(): InsertTokenUseCase {
912    return insertTokenUseCase
913}
914
915     fun updateTokenUseCase(): UpdateTokenUseCase {
916    return updateTokenUseCase
917}
918
919     fun deleteTokenUseCase(): DeleteTokenUseCase {
920    return deleteTokenUseCase
921}
922
923     fun insertReportUseCase(): SelectReportUseCase {
924    return selectReportUseCase
925}
926
927     fun insertReportUseCase(): InsertReportUseCase {
928    return insertReportUseCase
929}
930
931     fun updateReportUseCase(): UpdateReportUseCase {
932    return updateReportUseCase
933}
934
935     fun deleteReportUseCase(): DeleteReportUseCase {
936    return deleteReportUseCase
937}
938
939     fun createTokenUseCase(): CreateTokenUseCase {
940    return createTokenUseCase
941}
942
943     fun getTokenUseCase(): GetTokenUseCase {
944    return getTokenUseCase
945}
946
947     fun insertTokenUseCase(): InsertTokenUseCase {
948    return insertTokenUseCase
949}
950
951     fun updateTokenUseCase(): UpdateTokenUseCase {
952    return updateTokenUseCase
953}
954
955     fun deleteTokenUseCase(): DeleteTokenUseCase {
956    return deleteTokenUseCase
957}
958
959     fun insertReportUseCase(): SelectReportUseCase {
960    return selectReportUseCase
961}
962
963     fun insertReportUseCase(): InsertReportUseCase {
964    return insertReportUseCase
965}
966
967     fun updateReportUseCase(): UpdateReportUseCase {
968    return updateReportUseCase
969}
970
971     fun deleteReportUseCase(): DeleteReportUseCase {
972    return deleteReportUseCase
973}
974
975     fun createTokenUseCase(): CreateTokenUseCase {
976    return createTokenUseCase
977}
978
979     fun getTokenUseCase(): GetTokenUseCase {
978
979     fun insertTokenUseCase(): InsertTokenUseCase {
980    return insertTokenUseCase
981}
982
983     fun updateTokenUseCase(): UpdateTokenUseCase {
984    return updateTokenUseCase
985}
986
987     fun deleteTokenUseCase(): DeleteTokenUseCase {
988    return deleteTokenUseCase
989}
990
991     fun insertReportUseCase(): SelectReportUseCase {
992    return selectReportUseCase
993}
994
995     fun insertReportUseCase(): InsertReportUseCase {
996    return insertReportUseCase
997}
998
999     fun updateReportUseCase(): UpdateReportUseCase {
1000    return updateReportUseCase
1001}
1002
1003     fun deleteReportUseCase(): DeleteReportUseCase {
1004    return deleteReportUseCase
1005}
1006
1007     fun createTokenUseCase(): CreateTokenUseCase {
1008    return createTokenUseCase
1009}
1010
1011     fun getTokenUseCase(): GetTokenUseCase {
1010
1011     fun insertTokenUseCase(): InsertTokenUseCase {
1012    return insertTokenUseCase
1013}
1014
1015     fun updateTokenUseCase(): UpdateTokenUseCase {
1016    return updateTokenUseCase
1017}
1018
1019     fun deleteTokenUseCase(): DeleteTokenUseCase {
1020    return deleteTokenUseCase
1021}
1022
1023     fun insertReportUseCase(): SelectReportUseCase {
1024    return selectReportUseCase
1025}
1026
1027     fun insertReportUseCase(): InsertReportUseCase {
1028    return insertReportUseCase
1029}
1030
1031     fun updateReportUseCase(): UpdateReportUseCase {
1032    return updateReportUseCase
1033}
1034
1035     fun deleteReportUseCase(): DeleteReportUseCase {
1036    return deleteReportUseCase
1037}
1038
1039     fun createTokenUseCase(): CreateTokenUseCase {
1040    return createTokenUseCase
1041}
1042
1043     fun getTokenUseCase(): GetTokenUseCase {
1042
1043     fun insertTokenUseCase(): InsertTokenUseCase {
1044    return insertTokenUseCase
1045}
1046
1047     fun updateTokenUseCase(): UpdateTokenUseCase {
1048    return updateTokenUseCase
1049}
1050
1051     fun deleteTokenUseCase(): DeleteTokenUseCase {
1052    return deleteTokenUseCase
1053}
1054
1055     fun insertReportUseCase(): SelectReportUseCase {
1056    return selectReportUseCase
1057}
1058
1059     fun insertReportUseCase(): InsertReportUseCase {
1060    return insertReportUseCase
1061}
1062
1063     fun updateReportUseCase(): UpdateReportUseCase {
1064    return updateReportUseCase
1065}
1066
1067     fun deleteReportUseCase(): DeleteReportUseCase {
1068    return deleteReportUseCase
1069}
1070
1071     fun createTokenUseCase(): CreateTokenUseCase {
1072    return createTokenUseCase
1073}
1074
1075     fun getTokenUseCase(): GetTokenUseCase {
1075
1076     fun insertTokenUseCase(): InsertTokenUseCase {
1077    return insertTokenUseCase
1078}
1079
1080     fun updateTokenUseCase(): UpdateTokenUseCase {
1081    return updateTokenUseCase
1082}
1083
1084     fun deleteTokenUseCase(): DeleteTokenUseCase {
1085    return deleteTokenUseCase
1086}
1087
1088     fun insertReportUseCase(): SelectReportUseCase {
1089    return selectReportUseCase
1090}
1091
1092     fun insertReportUseCase(): InsertReportUseCase {
1093    return insertReportUseCase
1094}
1095
1096     fun updateReportUseCase(): UpdateReportUseCase {
1097    return updateReportUseCase
1098}
1099
1100     fun deleteReportUseCase(): DeleteReportUseCase {
1101    return deleteReportUseCase
1102}
1103
1104     fun createTokenUseCase(): CreateTokenUseCase {
1105    return createTokenUseCase
1106}
1107
1108     fun getTokenUseCase(): GetTokenUseCase {
1108
1109     fun insertTokenUseCase(): InsertTokenUseCase {
1110    return insertTokenUseCase
1111}
1112
1113     fun updateTokenUseCase(): UpdateTokenUseCase {
1114    return updateTokenUseCase
1115}
1116
1117     fun deleteTokenUseCase(): DeleteTokenUseCase {
1118    return deleteTokenUseCase
1119}
1120
1121     fun insertReportUseCase(): SelectReportUseCase {
1122    return selectReportUseCase
1123}
1124
1125     fun insertReportUseCase(): InsertReportUseCase {
1126    return insertReportUseCase
1127}
1128
1129     fun updateReportUseCase(): UpdateReportUseCase {
1130    return updateReportUseCase
1131}
1132
1133     fun deleteReportUseCase(): DeleteReportUseCase {
1134    return deleteReportUseCase
1135}
1136
1137     fun createTokenUseCase(): CreateTokenUseCase {
1138    return createTokenUseCase
1139}
1140
1141     fun getTokenUseCase(): GetTokenUseCase {
1142
1143     fun insertTokenUseCase(): InsertTokenUseCase {
1144    return insertTokenUseCase
1145}
1146
1147     fun updateTokenUseCase(): UpdateTokenUseCase {
1148    return updateTokenUseCase
1149}
1150
1151     fun deleteTokenUseCase(): DeleteTokenUseCase {
1152    return deleteTokenUseCase
1153}
1154
1155     fun insertReportUseCase(): SelectReportUseCase {
1156    return selectReportUseCase
1157}
1158
1159     fun insertReportUseCase(): InsertReportUseCase {
1160    return insertReportUseCase
1161}
1162
1163     fun updateReportUseCase(): UpdateReportUseCase {
1164    return updateReportUseCase
1165}
1166
1167     fun deleteReportUseCase(): DeleteReportUseCase {
1168    return deleteReportUseCase
1169}
1170
1171     fun createTokenUseCase(): CreateTokenUseCase {
1172    return createTokenUseCase
1173}
1174
1175     fun getTokenUseCase(): GetTokenUseCase {
1175
1176     fun insertTokenUseCase(): InsertTokenUseCase {
1177    return insertTokenUseCase
1178}
1179
1180     fun updateTokenUseCase(): UpdateTokenUseCase {
1181    return updateTokenUseCase
1182}
1183
1184     fun deleteTokenUseCase(): DeleteTokenUseCase {
1185    return deleteTokenUseCase
1186}
1187
1188     fun insertReportUseCase(): SelectReportUseCase {
1189    return selectReportUseCase
1190}
1191
1192     fun insertReportUseCase(): InsertReportUseCase {
1193    return insertReportUseCase
1194}
1195
1196     fun updateReportUseCase(): UpdateReportUseCase {
1197    return updateReportUseCase
1198}
1199
1200     fun deleteReportUseCase(): DeleteReportUseCase {
1201    return deleteReportUseCase
1202}
1203
1204     fun createTokenUseCase(): CreateTokenUseCase {
1205    return createTokenUseCase
1206}
1207
1208     fun getTokenUseCase(): GetTokenUseCase {
1208
1209     fun insertTokenUseCase(): InsertTokenUseCase {
1210    return insertTokenUseCase
1211}
1212
1213     fun updateTokenUseCase(): UpdateTokenUseCase {
1214    return updateTokenUseCase
1215}
1216
1217     fun deleteTokenUseCase(): DeleteTokenUseCase {
1218    return deleteTokenUseCase
1219}
1220
1221     fun insertReportUseCase(): SelectReportUseCase {
1222    return selectReportUseCase
1223}
1224
1225     fun insertReportUseCase(): InsertReportUseCase {
1226    return insertReportUseCase
1227}
1228
1229     fun updateReportUseCase(): UpdateReportUseCase {
1230    return updateReportUseCase
1231}
1232
1233     fun deleteReportUseCase(): DeleteReportUseCase {
1234    return deleteReportUseCase
1235}
1236
1237     fun createTokenUseCase(): CreateTokenUseCase {
1238    return createTokenUseCase
1239}
1240
1241     fun getTokenUseCase(): GetTokenUseCase {
1242
1243     fun insertTokenUseCase(): InsertTokenUseCase {
1244    return insertTokenUseCase
1245}
1246
1247     fun updateTokenUseCase(): UpdateTokenUseCase {
1248    return updateTokenUseCase
1249}
1250
1251     fun deleteTokenUseCase(): DeleteTokenUseCase {
1252    return deleteTokenUseCase
1253}
1254
1255     fun insertReportUseCase(): SelectReportUseCase {
1256    return selectReportUseCase
1257}
1258
1259     fun insertReportUseCase(): InsertReportUseCase {
1260    return insertReportUseCase
1261}
1262
1263     fun updateReportUseCase(): UpdateReportUseCase {
1264    return updateReportUseCase
1265}
1266
1267     fun deleteReportUseCase(): DeleteReportUseCase {
1268    return deleteReportUseCase
1269}
1270
1271     fun createTokenUseCase(): CreateTokenUseCase {
1272    return createTokenUseCase
1273}
1274
1275     fun getTokenUseCase(): GetTokenUseCase {
1275
1276     fun insertTokenUseCase(): InsertTokenUseCase {
1277    return insertTokenUseCase
1278}
1279
1280     fun updateTokenUseCase(): UpdateTokenUseCase {
1281    return updateTokenUseCase
1282}
1283
1284     fun deleteTokenUseCase(): DeleteTokenUseCase {
1285    return deleteTokenUseCase
1286}
1287
1288     fun insertReportUseCase(): SelectReportUseCase {
1289    return selectReportUseCase
1290}
1291
1292     fun insertReportUseCase(): InsertReportUseCase {
1293    return insertReportUseCase
1294}
1295
1296     fun updateReportUseCase(): UpdateReportUseCase {
1297    return updateReportUseCase
1298}
12
```

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help

EP Emprendimiento Primaria Version control

samsung SM-A217M app

Android

- > api_client
- > api_interface
- > data
- > ui
 - > model
 - EmailModel
 - Exercise1Model
 - Exercise2Model
 - Exercise3Model
 - ProgressModel
 - ProjectModel
 - StoryModel
 - UserModel
 - > theme
 - > view
 - > utils
 - LevelView.kt
 - LoginView.kt
 - MenuView.kt
 - NavigationView.kt
 - OpinionView.kt
 - ProjectListView.kt
 - RecoverKeyView.kt
 - RegisterView.kt
 - UnitsView.kt
 - UserInfoView.kt
 - >.viewmodel
 - ProgressViewModel

45 if(level == "0") {

38 fun UnitsView(

50) {

101 contentScale = ContentScale.Fit

102)

103 }

104 }else{

105 when(level){

106 "1.1" -> LevelView(

107 modifier = Modifier.fillMaxSize(),

108 progressVm = progressVm,

109 exercise1 = Exercise1.data[0],

110 exercise2 = Exercise2.data[0],

111 exercise3 = Exercise3.data[0],

112 numLevel = 1,

113 totalLevels = totalLevels,

114 onCallback = { level = "0" }

115)

116 "1.2" -> LevelView(

117 modifier = Modifier.fillMaxSize(),

118 progressVm = progressVm,

✓ Up-to-date | ✓ 4

Running Devices +

19:14

Elija verdadero o falso
El ingreso es la cantidad de bienes obtenidos por el precio de venta de un producto o servicio.

Verdadero Falso

Continuar

45:23 CRLF UTF-8 4 spaces

App Educativa de Emprendimiento

A File Edit View Navigate Code Refactor Build Run Tools VCS Window Help

Player Music Version control

samsung SM-A217M app C D Running Devices

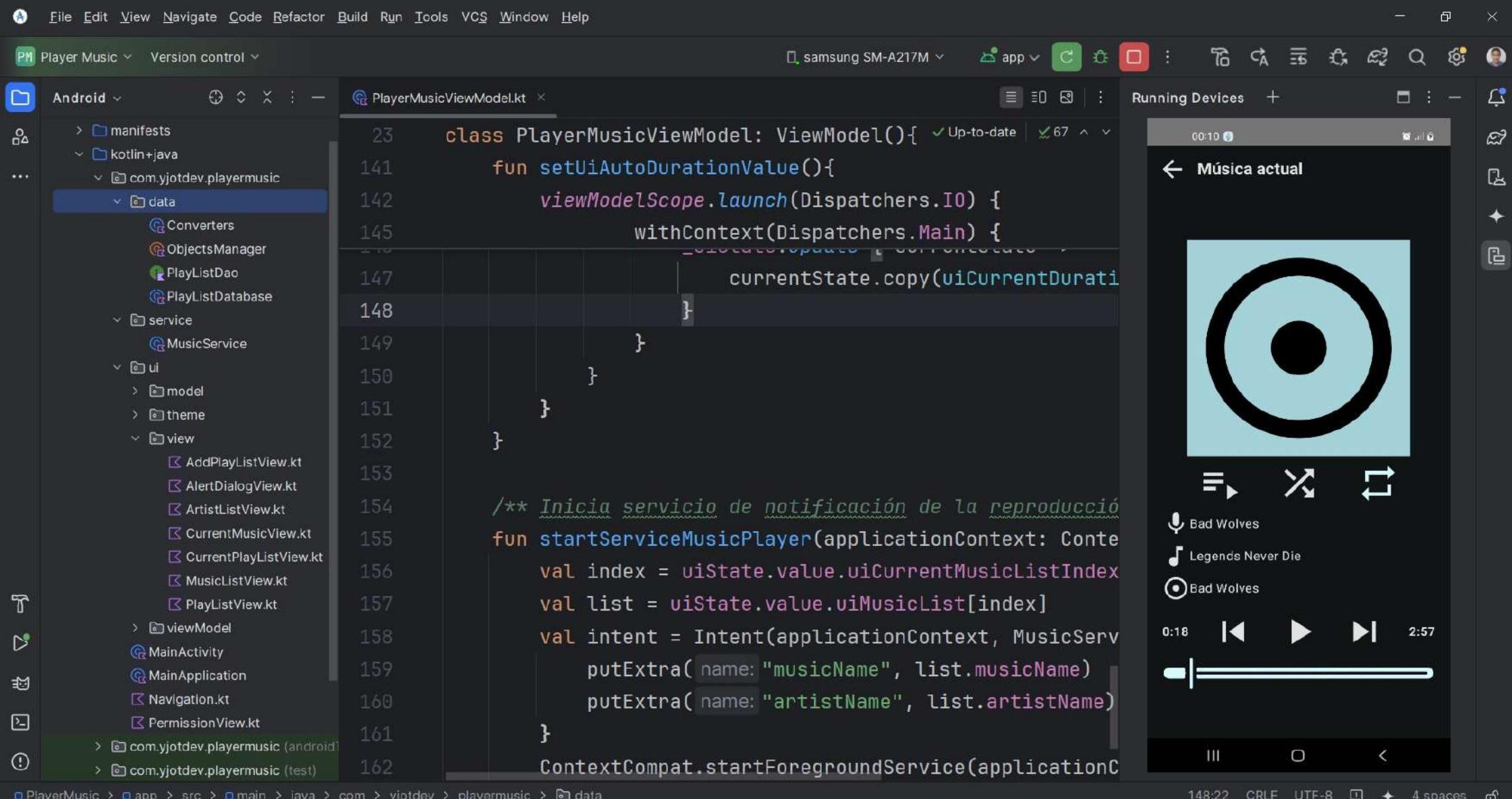
Android PlayerMusicViewModel.kt

```
23     class PlayerMusicViewModel: ViewModel(){ Up-to-date | 67 ↻
141         fun setUiAutoDurationValue(){
142             viewModelScope.launch(Dispatchers.IO) {
143                 withContext(Dispatchers.Main) {
144                     currentState.copy(uiCurrentDuration = uiCurrentDuration)
145                 }
146             }
147         }
148     }
149
150     }
151 }
152 }
153
154 /**
155 * Inicia servicio de notificación de la reproducción
156 */
157 fun startServiceMusicPlayer(applicationContext: ContextCompat,
158     val index = uiState.value.uiCurrentMusicListIndex
159     val list = uiState.value.uiMusicList[index]
160     val intent = Intent(applicationContext, MusicService::class.java)
161         putExtra("musicName", list.musicName)
162         putExtra("artistName", list.artistName)
163     }
164     ContextCompat.startForegroundService(applicationContext, intent)
165 }
```

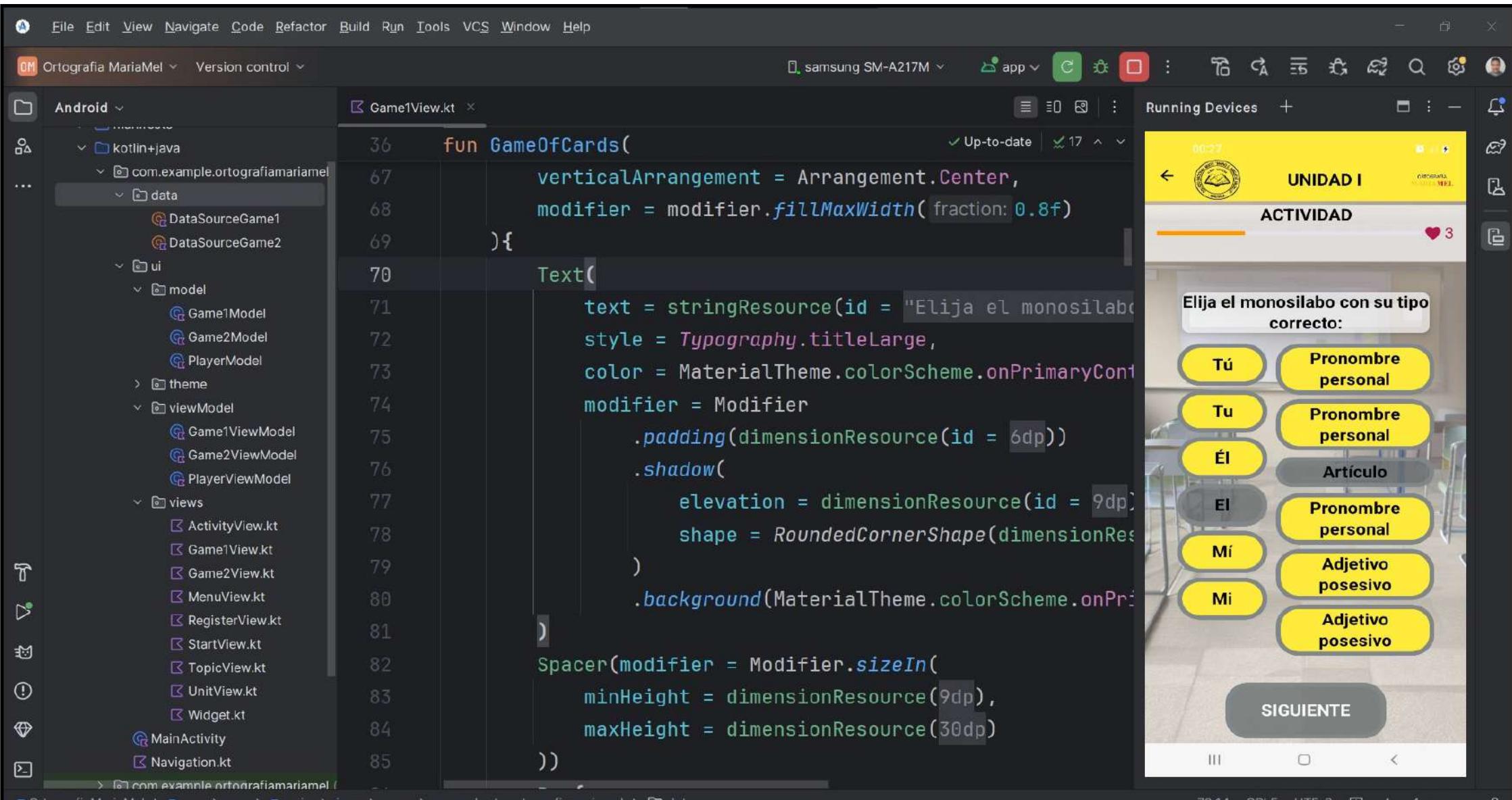
Bad Wolves
Legends Never Die
Bad Wolves

0:18 2:57

148:22 CRLF UTF-8 4 spaces



App Reproductor de Música



App Educativa de Ortografía

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help login - Userinterface.kt [login.app.main]

login > app > src > main > java > com > example > login > interfaces

Userinterface.kt

```
1 package com.example.login.interfaces
2
3 import ...
4
5 interface UserInterface {
6     @POST("find_user/")
7     suspend fun findUser(@Body user:UserModel): Response<UserModel>
8
9     @GET("find_user_by_email/{email}")
10    suspend fun findUserByEmail(@Path("email") email:String): Response<UserModel>
11
12    @POST("insert_user/")
13    suspend fun insertUser(@Body user:UserModel): Response<Unit>
14
15    @PUT("update_user/{id}")
16    suspend fun updateUser(@Path("id") id:Int, @Body user:UserModel): Response<Unit>
17
18    @DELETE("delete_user/{id}")
19    suspend fun deleteUser(@Path("id") id:Int): Response<Unit>
20
21 }
```

Emulator: Nexus 5X API 33

GESTION DE USUARIO

yasser

2010guabo@gmail.com

•••

EDITAR USUARIO

ELIMINAR USUARIO

+

-

1:1

Layout Inspector

Version Control Profiler Logcat App Quality Insights Build TODO Problems Terminal Services App Inspection

Launch succeeded (a minute ago)

22:2 CRLF UTF-8 4 spaces

App Login de usuario con API y BD

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help CalculadorDeMateriales - PisoFragment.kt [CalculadorDeMateriales.app.main]

Resource Manager Project Bookmarks Build Variants Structure

Emulator: Nexus 5X API 33

Gradle Device Manager Notifications Emulator Device File Explorer

App Calculadora de materiales

PisoFragment.kt

```
21     val baldosasPorCaja = binding.editBaldos 15 ^ v
22     if(m2DeCajaDeBaldosa.isNotEmpty() && m2DeLaHabitacion
23         calcular(m2DeCajaDeBaldosa.toFloat(), m2DeLaHabitacion)
24     }else{
25         Toast.makeText(this.context, text:"EXISTEN CAMPOS VACIOS", duration:Toast.LENGTH_SHORT).show()
26     }
27 }
28
29
30 override fun onCreateView(
31     inflater: LayoutInflater, container: ViewGroup?,
32     savedInstanceState: Bundle?
33 ): View {
34     binding = FragmentPisoBinding.inflate(inflater)
35     return binding.root
36 }
37
38 private fun calcular(m2DeCajaDeBaldosa: Float, m2DeLaHabitacion: Float): Int {
39     val totalCantidadDeCajasDeBaldosa = m2DeLaHabitacion / m2DeCajaDeBaldosa
40     val totalCantidadDeBaldosas = (totalCantidadDeCajasDeBaldosa * baldosasPorCaja).toInt()
41
42     binding.result.text = "SE NECESITAN ${totalCantidadDeCajasDeBaldosa} CAJAS DE BALDOSAS EQUIVALENTE A $totalCantidadDeBaldosas BALDOSAS"
43 }
44 }
```

Nexus 5X API 33

1.26 CALCULAR

CALCULAR NUMERO DE BALDOSAS PARA EL PISO

M ² DE LA CAJA DE BALDOSA	2.5
M ² DE LA HABITACION	12
BALDOSAS POR CAJA	4

SE NECESITAN 5 CAJAS DE BALDOSAS EQUIVALENTE A 20 BALDOSAS

TEMA OSCURO

Version Control Profiler Logcat App Quality Insights Build TODO Problems Terminal Services App Inspection Layout Inspector

Launch succeeded (2 minutes ago) 44:2 LF UTF-8 4 spaces

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help ClasificarPeces - ImageClassifier.kt [ClasificarPeces.app.main]

ain > java > com > example > clasificarpeces > ImageClassifier > ClassifierListener > onResults MainActivity Pixel 4 API 31 Emulator: Pixel 4 API 31 Project Resource Manager Gradle Device Manager Device File Explorer Emulator Notifications

App Clasificador de raza de peces

```
ImageClassifier.kt
```

```
63
64     fun classify(image: Bitmap, rotation: Int) {
65         if (imageClassifier == null) {
66             setupImageClassifier()
67         }
68
69         var inferenceTime = SystemClock.uptimeMillis()
70
71         val imageProcessor =
72             ImageProcessor.Builder()
73                 .build()
74
75         val tensorImage = imageProcessor.process(TensorImage.fromBitmap
76
77         val imageProcessingOptions = ImageProcessingOptions.builder()
78             .setOrientation(getOrientationFromRotation(rotation))
79             .build()
80
81         val results = imageClassifier?.classify(tensorImage, imageProcessingOptions)
82         inferenceTime = SystemClock.uptimeMillis() - inferenceTime
83         imageClassifierListener?.onResults(
84             results,
85             inferenceTime
86         )
87     }
88
89     private fun getOrientationFromRotation(rotation: Int): ImageProcessingOptions.Orientation {
90         return when (rotation) {
91             Surface.ROTATION_0, Surface.ROTATION_180 -> ImageProcessingOptions.Orientation.BOTTOM_RIGHT
92             Surface.ROTATION_90, Surface.ROTATION_270 -> ImageProcessingOptions.Orientation.TOP_LEFT
93         }
94     }

```

Emulator: Pixel 4 API 31

ANALIZAR IMAGEN

guppy ---> 98 %

9:28 ANALYSIS



A File Edit View Navigate Code Refactor Build Run Tools VCS Window Help

ZonasTurísticasElGuabo Version control

samsung SM-A217M app C D Running Devices 15:56

Android MapsActivity.kt

```
14 class MapsActivity : AppCompatActivity(), OnMapReadyListener {  
15     private lateinit var mMap: GoogleMap  
16  
17     override fun onMapReady(googleMap: GoogleMap) {  
18         mMap = googleMap  
19  
20         // Posicion inicial de camara  
21         val posStart = LatLng( latitude: -3.2099019884381983, longitude: -78.52000000000001 )  
22         mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(posStart, 15f))  
23  
24         // Agregar marcas de locacion  
25         val pos1 = LatLng( latitude: -3.2041386029873475, longitude: -78.51500000000001 )  
26         mMap.addMarker(MarkerOptions().position(pos1).title("El Mirador"))  
27         val pos2 = LatLng( latitude: -3.105959818945006, longitude: -78.52000000000001 )  
28         mMap.addMarker(MarkerOptions().position(pos2).title("El Guabo"))  
29         val pos3 = LatLng( latitude: -3.2535029218247113, longitude: -78.51500000000001 )  
30         mMap.addMarker(MarkerOptions().position(pos3).title("Tiquipaya"))  
31         val pos4 = LatLng( latitude: -3.178280150740988, longitude: -78.52000000000001 )  
32         mMap.addMarker(MarkerOptions().position(pos4).title("La Reña"))  
33         val pos5 = LatLng( latitude: -3.188914657907403, longitude: -78.51500000000001 )  
34         mMap.addMarker(MarkerOptions().position(pos5).title("Tiquipaya"))  
35  
36     }  
37  
38     companion object {  
39         @JvmStatic  
40         fun main(args: Array<String>) {  
41             val intent = Intent(Intent.ACTION_MAIN)  
42             intent.addCategory(Intent.CATEGORY_LAUNCHER)  
43             intent.component = ComponentName("com.example.zonasturisticaselguabo", "com.example.zonasturisticaselguabo.InfoActivity")  
44             LocalBroadcastManager.getInstance(this).sendBroadcast(intent)  
45         }  
46     }  
47  
48     override fun onCreateOptionsMenu(menu: Menu): Boolean {  
49         menuInflater.inflate(R.menu.activity_main, menu)  
50         return true  
51     }  
52  
53     override fun onOptionsItemSelected(item: MenuItem): Boolean {  
54         when (item.itemId) {  
55             R.id.action_settings -> openSettings()  
56             else -> return super.onOptionsItemSelected(item)  
57         }  
58         return true  
59     }  
60  
61     private fun openSettings() {  
62         val intent = Intent(Settings.ACTION_SETTINGS)  
63         LocalBroadcastManager.getInstance(this).sendBroadcast(intent)  
64     }  
65 }
```

Mapa de El Guabo

App Mapa Turístico de El Guabo

The screenshot displays the Visual Studio IDE interface for a Xamarin project named "DepreciationCalculator".

Solution Explorer: Shows the solution structure with two projects: "DepreciationCalculator" (containing files like App.xaml, AssemblyInfo.cs, DatosPersistentes.cs, Deprecacion.xaml, Idiomas.xaml, and Deprecacion.xaml.cs) and "DepreciationCalculator.Android".

Code Editor: The file "Deprecacion.xaml.cs" is open, containing C# code for calculating depreciation based on initial value, years, and usage level.

```
23     float valorC, valorDepreciacion, precioPorAnio, porcentaje = 0;
24     switch (NivelDeProblemas.SelectedIndex)
25     {
26         case 0: //Ninguno
27             porcentaje = 1;
28             break;
29         case 1: //Leve
30             porcentaje = 0.75f;
31             break;
32         case 2: //Medio
33             porcentaje = 0.50f;
34             break;
35         case 3: //Grave
36             porcentaje = 0.25f;
37             break;
38     }
39     valorC = (float)Math.Round((decimal)(valorA * porcentaje), 2);
40     precioPorAnio = (valorB * 10);
41     valorC -= precioPorAnio;
42     if (valorC > precioPorAnio)
43     {
44         valorDepreciacion = (float)Math.Round((decimal)((valorA - valorC) / valorB));
45         ResultadoTraducido(valorDepreciacion, valorC);
46     }
47 }
48 }
```

Output Window: Shows the message "No se encontraron problemas." (No errors found).

Task List: Shows 0 Errors, 0 Warnings, and 0 Messages.

Project Explorer: Shows the project structure with files like App.xaml, AssemblyInfo.cs, DatosPersistentes.cs, Deprecacion.xaml, Idiomas.xaml, and Deprecacion.xaml.cs.

Properties: Shows the properties of the selected file.

Toolbox: Shows various UI components available for drag-and-drop.

Code Editor Status Bar: Shows the status "CRLF".

Emulator: Displays the "DEPRECIACIÓN" screen of the mobile application. It has three input fields: Valor A (500), Anios (5), and Nivel de Problemas (Leve). A large orange button labeled "ANALIZAR" is present. Below it, a text block states: "Si su producto es nuevo cada año disminuirá el precio en \$35, si su producto es usado el precio actual sería de \$325".

App Calculador de depreciación de un producto

Archivo Editar Ver Git Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar (Ctrl+Q) Detector de suelos

Detector de suelos Application Insights Live Share

Inicio.xaml.cs

Detector_de_suelos

Detector_de_suelos.Inicio

18 0 referencias
19 private async void ElegirImage(object sender, EventArgs e)
20 {
21 try
22 {
23 PermissionStatus status = await Permissions.CheckStatusAsync<Permissions.Camera>();
24 if (status != PermissionStatus.Granted)
25 {
26 _ = await Permissions.RequestAsync<Permissions.Camera>();
27 }
28 await ob.ElegirImage(sender, e);
29 await Navigation.PushAsync(new Analisis(ob.foto));
30 } catch (Exception) { }
31 }
32 0 referencias
33 private async void TomarFoto(object sender, EventArgs e)
34 {
35 try
36 {
37 PermissionStatus status = await Permissions.CheckStatusAsync<Permissions.Camera>();
38 if (status != PermissionStatus.Granted)
39 {
40 _ = await Permissions.RequestAsync<Permissions.Camera>();
41 }
42 await ob.TomarFoto(sender, e);
43 await Navigation.PushAsync(new Analisis(ob.foto));
44 }
45 }
46 No se encontraron problemas.

Explorador de soluciones

Solución "Detector de suelos" (2 de 2 proyectos)

- Detector de suelos
 - Dependencias
 - Controlador
 - Modelo
 - Vistas
 - Analisis.xaml
 - Inicio.xaml
 - Inicio.xaml.cs
- App.xaml
- AssemblyInfo.cs

Detector de suelos.Android

11:10 Inicio

¡Identifica el nombre del suelo según su color!

Elige una foto para analizar y clasificar el suelo por su color

ELEGIR UNA IMAGEN

TOMAR UNA FOTO

Pila de llamadas

Nombre

Variables locales

Buscar (Ctrl+E)

Profundidad de búsqueda:

Nombre Valor Tipo

Este elemento no admite la vista previa

Agregar al control de código fuente

App Analizador y clasificador de tonalidad del suelo

Archivo Editar Ver Git Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar (Ctrl+Q) Red Car Wash Application Insights Live Share

IniciarSesion.xaml.cs

```
private void BtnVerClave(object sender, EventArgs e)
{
    Clave.IsPassword = false;
    if (verClave)
    {
        Clave.IsPassword = true;
        verClave = false;
    }
    else
    {
        verClave = true;
    }
}

private void BtnIniciarSesion(object sender, EventArgs e)
{
    MetodoIniciarSesion(Correo.Text, Clave.Text);
}

private void BtnRegistrarse(object sender, EventArgs e)
{
    _ = Navigation.PushAsync(new Registrarse());
}
```

Explorador de soluciones

Solución "Red Car Wash" (2 de 2 proyectos)

- Red Car Wash
 - Dependencias
 - Controlador
 - Modelo
- Vista
 - ColorDeVehiculo.xaml
 - InformeMes.xaml
 - Informes.xaml
 - IniciarSesion.xaml
 - IniciarSesion.xaml.cs
 - Inicio.xaml
 - MarcaDeVehiculo.xaml
 - Negocio.xaml
 - Personal.xaml
 - RecuperarClave.xaml
 - Registrarse.xaml
 - RegistroDeEmpleado.xaml
 - RegistroDeTipo Lavado.xaml
 - RegistroDeVehículo.xaml
 - Servicios.xaml
 - TipoDePago.xaml
 - TipoDeVehiculo.xaml
 - UsuarioCliente.xaml
 - UsuarioCliente.xaml.cs
 - UsuarioEmpleado.xaml
 - UsuarioListaEmpleados.xaml
 - App.xaml
- Red Car Wash.Android

INICIO DE SESIÓN

Bienvenido, inicie sesión

Ingrese su correo

Ingrese su clave VER

INICIAR SESIÓN

REGISTRARSE

RECUPERAR CLAVE

Variables locales

Buscar (Ctrl+E)

Nombre

Pla de llamadas Configuración de excepciones Ventana Inmediato

Agregar al control de código fuente

App Gestor de lavado de vehiculos

Archivo Editar Ver Git Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar (Ctrl+Q) ConstitucionDelEcuador Application Insights Live Share

Árbol visual dinámico

Explorador de soluciones

Buscar en Explorador de soluciones (Ctrl+F)

Solución "ConstitucionDelEcuador" (2 de 2 proyectos)

- ConstitucionDelEcuador
 - Dependencias
 - Control
 - Entidad
 - GUI
 - GUI_Constitucion.xaml
 - GUI_Inicio.xaml
 - GUI_Menu.xaml
 - GUI_Menu.xaml.cs
 - GUI_MenuLateral.xaml
 - GUI_Seccion.xaml
 - App.xaml
 - AssemblyInfo.cs
- ConstitucionDelEcuador.Android
 - Connected Services
 - Properties
 - Referencias
 - Assets
 - Resources
 - MainActivity.cs

Explorador de propiedades dinámico

ConstitucionDelEcuador

ConstitucionDelEcuador.GUI.GUI_Menu.xaml.cs

```
0 referencias
23 public void MetodoAcercaDe(object sender, EventArgs e)
24 {
25     VerAnuncio();
26     IdTitulo = 1;
27     Navigation.PushAsync(new GUI_Seccion());
28 }
29

0 referencias
30 public void MetodoContacto(object sender, EventArgs e)
31 {
32     VerAnuncio();
33     IdTitulo = 2;
34     Navigation.PushAsync(new GUI_Seccion());
35 }
36

0 referencias
37 public void MetodoResumen(object sender, EventArgs e)
38 {
39     VerAnuncio();
40     IdTitulo = 3;
41     Navigation.PushAsync(new GUI_Seccion());
42 }
43

0 referencias
44 public void MetodoConstitucion(object sender, EventArgs e)
45 {
```

100 % No se encontraron problemas.

Variables locales

Buscar (Ctrl+E) Profundidad de búsqueda:

Nombre	Valor	Tipo

App Lector de la constitución del Ecuador

Errores de enlaces XAML Variables locales Inspección 1 Pila de llamadas Configuración de excepciones Ventana Inmediato Recarga activa de XAML conectada Agregar al control de código fuente



UNIVERSIDAD TÉCNICA DE MACHALA

D.L. NO. 69-04 DE 14 DE ABRIL DE 1969

Calidad, Pertinencia y Calidez

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

SECCIÓN/CARRERA: INGENIERÍA DE SISTEMAS - TECNOLOGÍAS DE LA INFORMACIÓN

ING. JIMMY MOLINA RÍOS, MG.

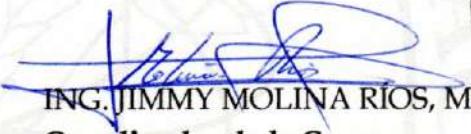
Coordinador de la Carrera

CERTIFICO:

Que el señor: ORTIZ TRUJILLO YASSER JAVIER portador de la Cédula de Identidad N. 0705267268, ha concluido sus Prácticas Pre Profesionales y/o Pasantías de la Carrera de Ingeniería de Sistemas; una vez revisado la documentación según informe archivado con el código: IS-PPP-19-027; estableciendo la tutoría institucional a cargo de la ING. MAZÓN OLIVO BERTHA EUGENIA, MG; en el Grupo de Investigación de la Facultad de Ingeniería Civil de la UTMACH y tutoría académica a cargo del ING. RIVAS ASANZA WILMER BRAULIO, MG.; con quienes el interesado realizó sus prácticas pre profesionales y/o pasantías en el período comprendido entre el 1 de noviembre de 2019 hasta el 31 de enero de 2020, cumpliendo las 240 horas reglamentarias; quedando facultado para que continúe con el trámite correspondiente como requisito para su titulación.

Es todo cuanto puedo certificar.

Machala, febrero 27 de 2020.


ING. JIMMY MOLINA RÍOS, MG.

Coordinador de la Carrera.

Email: jmolina@utmachala.edu.ec

JRMR/Carlos Jumbo

