International Journal of Computer and Information System (IJCIS)
Peer Reviewed – International Journal
Vol : Vol. 05, Issue 02, May 2024
e-ISSN : 2745-9659
https://ijcis.net/index.php/ijcis/index

Design of Real Time Monitoring Queue List and Android-Based Search for Nearest Ambulance at Rangkasbitung Hospital

[1st]Widura Krisna, [2st]Muhammad Zakariyah
[1,2]University of Technology Yogyakarta
E-mail: widurakrisna4@gmail.com[1], *muhammad.zakariyah@staff.uty.ac.id*[2]

Abstract—Information technology has now changed almost all agencies, both private and public, especially at Dr. Adjidarmo Hospital, some problems at Dr. Adjidarmo Hospital requires solutions. One of the problems that exists is the accumulation of queues because they still use a manual system and take a long time and difficulty for patients to find ambulance information.The use of technology to improve this system is the best solution that can be done using the ADDIE method. ADDIE stands for the five stages of the development process, namely Analysis, Design, Development, Implementation, and Evaluation. The idea to simplify patient bureaucracy can be done with the health service android application with the Ambulance feature. Patients can use smartphones as a medium to find information on the whereabouts of ambulances. Patients can also determine where the location point is by using Google Maps which can find where a place or location is located. For queuing systems use firebase as a data store.

Keywords: *Fifth keyword Android, Google Maps, Smartphone.Firebase, ADDIE, Queue, Ambulance*

I. INTRODUCTION

Rangkasbitung is a city in Banten province. Rangkasbitung Regional Hospital still has several problems, especially in the community sector, namely hospitals. This is because the queues are still manual and outdated, and it is difficult to find an ambulance because there is no information about ambulances, especially for community needs.

Based on observations, the number of hospital patients is 300 patients per day, each examination takes around 30 minutes and patients have to queue for an estimated 3 to 5 hours. The quality and service of hospitals is still not optimal, therefore researchers suspect that the queues made by hospitals still use manuals and call patients individually. Currently this method is considered less efficient, an Android-based queuing system is one solution to overcome this.

As technology advances, it is best to use technology to improve the service system. To improve queue services for patients and ambulance services can be realized with an Android application. Patients can use their smartphone to search for information about the location of the nearest ambulance and can see a list of ongoing queues.

Therefore, the author aims to create an application for monitoring the queue list and searching for the nearest ambulance. It is hoped that the application created will make it easier for patients to find information on the nearest ambulance and patients can see the queue number online, and patients do not need to go to the hospital to see the queue number[6].

II. RESEARCH METHODS

The research framework is created in such a way that it is easier to understand the research. This research framework must be carried out before the research steps are taken, this research framework functions as preparation for the research so that it is more mature as in Figure 1

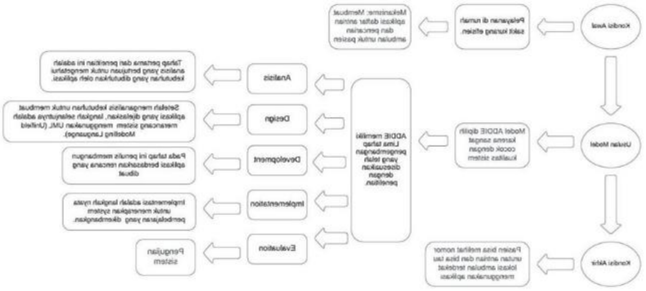

Kondisi Awal
Pelayanan di rumah sakit kurang efisien.
Mekanisme: Membuat aplikasi daftar antrian dan pencarian ambulan untuk pasien
Usulan Model
Model ADDIE dipilih karena sangat cocok dengan kualitas sistem
ADDIE memiliki Lima tahap pengembangan yang telah disesuaikan dengan penelitian.
Analisis
Tahap pertama dan penelitian ini adalah analisis yang bertujuan untuk mengetahui kebutuhan yang dibutuhkan oleh aplikasi.
Design
Setelah menganalisis kebutuhan untuk membuat aplikasi yang dijelaskan, langkah selanjutnya adalah merancang sistem menggunakan UML (Unified Modeling Language).
Development
Pada tahap ini penulis membangun aplikasi berdasarkan rencana yang dibuat
Implementation
Implementasi adalah langkah nyata untuk menerapkan system pembelajaran yang dikembangkan.
Evaluation
Pengujian sistem
Kondisi Akhir
Pasien bisa melihat nomor urutan antrian dan bisa tau lokasi ambulan terdekat menggunakan aplikasi

Figure 1. Research Framework

Research Data

At this stage the researcher obtained data from Dr. Adjidarmo Rangkasibitung Regional Hospital. Data source

How to get Data

At this stage, researchers conducted observations and interviews with patients for one week. Observations and interviews were carried out to obtain data and to find out what assessment variables influence queues and ambulances at Dr. Adjidarmo Rangkasibitung Regional Hospital. Observations were carried out from

Monday to Saturday, by observing the number of patient arrivals, patient waiting time for outpatient treatment and the length of time patients received treatment. Meanwhile, interview the patient by asking questions. The following are questions for patients How long did you wait in line to get a ticket or registration card. How long do you wait in line to get an examination by a doctor? Do you know how to order a research ambulance

Patient arrival time, patient treatment time and average level of patient service are the results of interviews and observations at Dr. Adjidarmo Rangkasibitung Regional Hospital. The following statistical description of patient arrival times is given in tables 1 and 2.

Table 1. Daily Patient Data

Day	Arrival of Patients (People)
Monday	173
Tuesday	189
Wednesday	162
Thursday	128
Friday	112
Saturday	127

Table 2. Patient Arrival Data

Time Period (Hours)	Arrival of Patients (people)
09.00-10.00	10
10.00-12.00	20
12.00-13.00	17
13.00-15.00	6

Analysis

The first stage of this research is analysis which aims to determine the needs required by the application. With the help of analysis, it can be determined for what purpose this application is developed and to whom this application is shown.

Design

After analyzing the need to create the application described, the next step is to design a system using UML (Unifield Modeling Language). The design is carried out using UML to describe the relationship between users and applications. The UML used to explain how an application works is Use Case Diagrams, Flowcharts and Activity Diagrams.

Development

At this stage, the author prepares an application based on the plan created, which is a systematic description of the design form created based on the analysis made. The goal of user interface design is to present the application in a way that looks simple. The queue and ambulance list application for Android has one type of interface, namely for use with Android phones. As follows :

Interface Design on Android Smartphones

Interface design for Android mobile devices aims to help researchers design and build applications used by Android application users. The Android interface design for users and admins is as follows:

1. Landing Page Login and Register
2. Register Page
3. Login Page
4. User Home Page
5. Admin Main Page

Implementation

Implementation is a concrete step to implement the learning system being developed. So at this stage, everything that is developed is installed or configured according to the role or task so that it can be implemented. The implementation stage of this research was carried out by direct media testing. The results of this test will be used as a basis for the assessment stage (Evaluation).

Evaluation

System testing is carried out using the black box testing method. In Black Box testing, the application is tested based on details such as the appearance of the application, the functions in the application and the suitability of the function flow with the customer's desired process. Black box testing is more about testing the appearance (interface) of an application so that it is easier for users to use. This test does not view and test the program source code. Black box testing ignores the control structure, so that attention is focused only on domain information.

The aim of this method is to check after the final stage of the project that the software or application is functioning properly and serving the users effectively.

III. RESULT AND ANALYSIS

System Implementation

Sign In and Doctor Registration

The login page is managed by the main administrator, patients can register and log in to enter the system, while doctors don't need to register because a special account has been created by the administrator, doctors only need to fill in the correct username and password to access the main page. As in the figure 2.

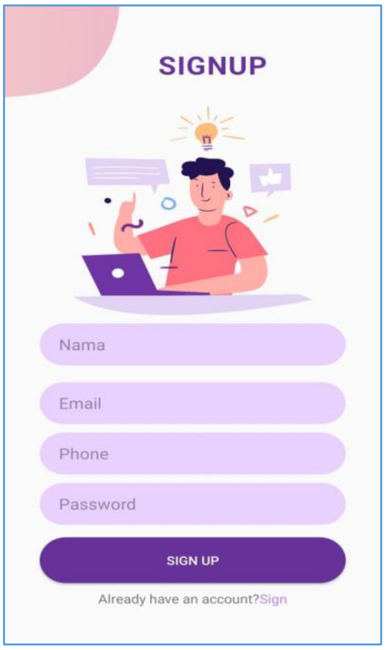


Figure 2. Form Pages Sign Up

In addition to filling in the registration is complete, then continue to enter the logn system page as presented in figure 3.

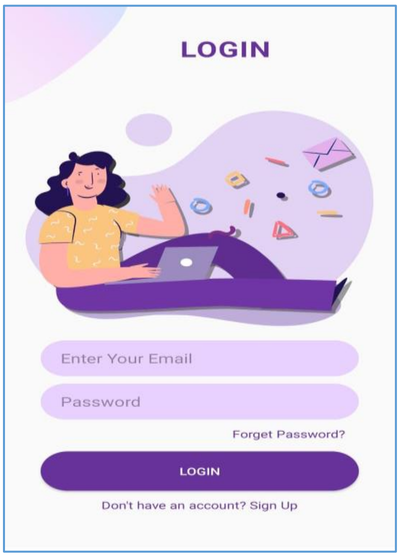


Figure 3. Form Register and Login

Main Page

The main page for patients is like Figure 5.4, while for doctors it is like Figure 5.5. The difference between the main page for doctors and patients is that in the menu, the patient menu has two menus, namely the queue menu and the ambulance menu. For doctors there is only one menu, namely the queue menu show in figure 4

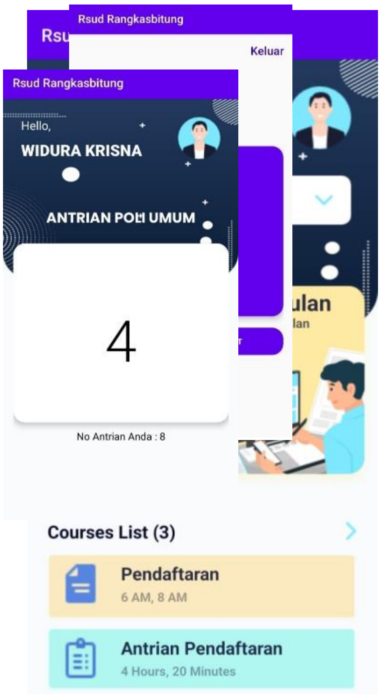


Figure 4. Admin Main Page

Poly Page

The poly page makes it easier for patients to select the poly for which they want to see the queue number as in figure 5.

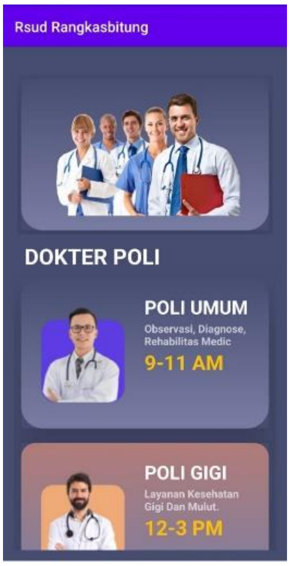

Rsud Rangkasbitung
DOKTER POLI
POLI UMUM
Observasi, Diagnose, Rehabilitas Medic
9-11 AM
POLI GIGI
Layanan Kesehatan Gigi Dan Mulut.
12-3 PM

Figure 5. Admin Main Page

Queue Page

After the doctor and patient select the queue number, the queue number will be displayed. For the doctor's queue page as in Figure 5.7, on this page the doctor can move or edit the queue number, whereas in the picture, it is a queue page for patients. On this page, the patient can only see the current queue number. taking place

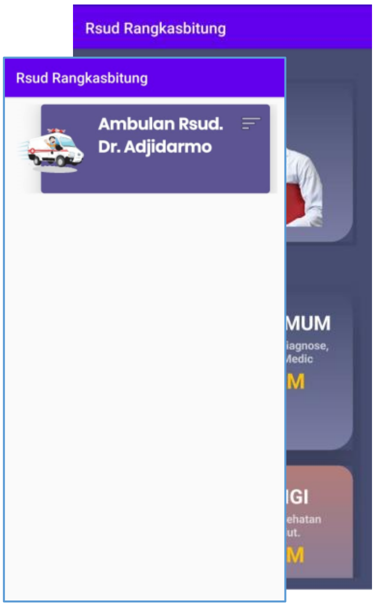


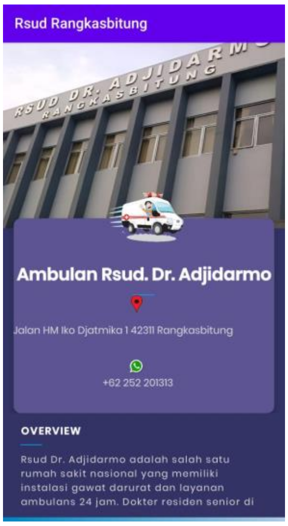
Figure 6. Patient Queue

Ambulance Page

This page can only be seen by patients, on this page there is a list of ambulances in Rangkasbitung along with information on the ambulance selected by the patient. As in figure 7.

Ambulance

1. Ambulance Information Page This page contains information on the ambulance selected by the patient, as in Figure 7.


Rsud Rangkasbitung
RSUD DR. ADJIDARMO RANGKASBITUNG
Ambulan Rsud. Dr. Adjidarmo
Jalan HM Iko Djatmika 1 42311 Rangkasbitung
+62 252 201313
OVERVIEW

Rsud Dr. Adjidarmo adalah salah satu rumah sakit nasional yang memiliki instalasi gawat darurat dan layanan ambulans 24 jam. Dokter residen senior di

Figure 7. Ambulance Information

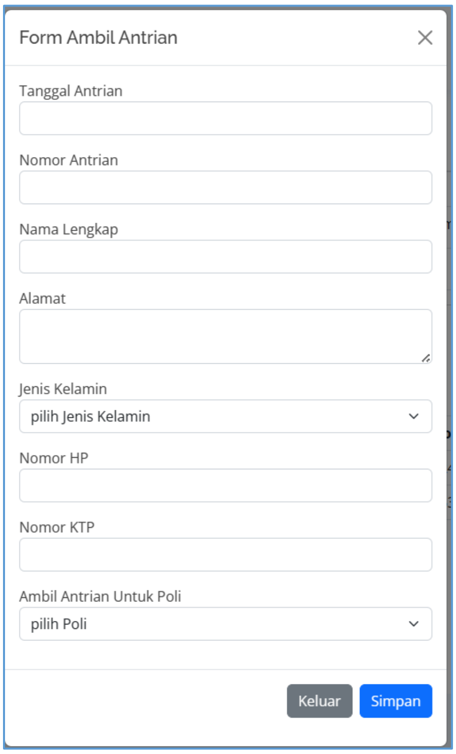


Figure 8. Queue Registration Page

2. Website Login Page

The login display will display a login form for admin. The account that will be able to log in is the account that has been registered on the website. Admin can fill in the login form by entering the email and password according to the registered account. The Login button will validate the account that has been entered. It can be seen in figure 9.

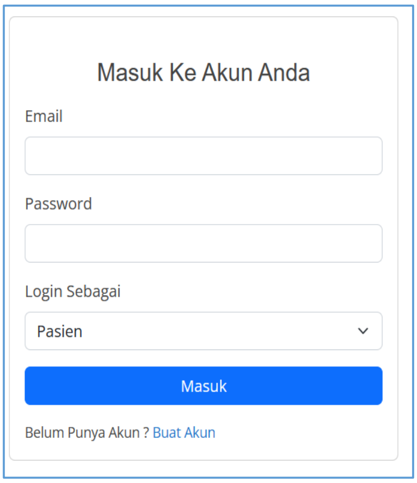
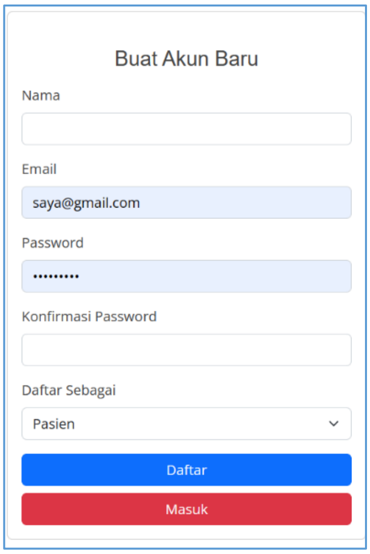


Figure 9. Website Login

3. Website Register Page

   The register display will display a registration form for admin. This form will ask the admin to enter several things figure 9.

.



Buat Akun Baru

Nama

Email
saya@gmail.com

Password
•••••••••

Konfirmasi Password

Daftar Sebagai
Pasien

Daftar

Masuk

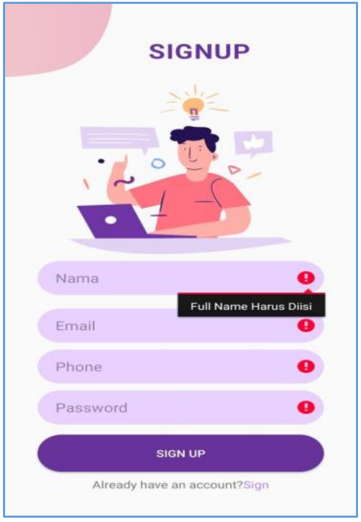
Figure 10. Create Acount Login

Discussion

After the implementation process, the next step is the testing process. This step explains the testing of the real time application for monitoring the queue list and searching for the nearest ambulance based on Android. The tests carried out to test the application are by carrying out two tests, including error testing if an error occurs and black box testing.

Results Error Testing

Application users often make mistakes, especially in the login and register sections. Users often forget rules or information when logging in and registering which can prevent users from being able to log in and register, such as not filling in the password or email when logging in and filling in the data when registering that the user enters. incomplete. Incomplete data can affect system performance in producing valid data. Therefore, the system gives a warning to the user if an error occurs as in figure 10 and figure 11



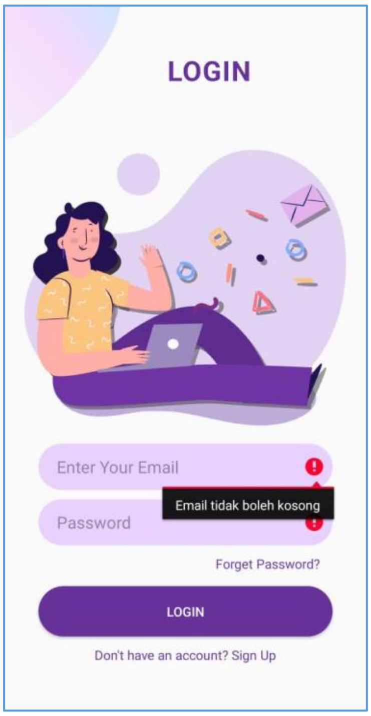
SIGNUP

Nama
Full Name Harus Diisi
Email
Phone
Password

SIGN UP

Already have an account?Sign

Figure 11. Registration Error



LOGIN

Enter Your Email
Email tidak boleh kosong
Password

Forget Password?

LOGIN

Don't have an account? Sign Up

Figure 12. Login Error

Unit Testing (Black Box Testing)

Testing that highlights the input process in the system and produces output. Several experiments were carried out with this system to achieve the expected system. Tests carried out by patients and doctors show in table 3.

Table 3. The results of tests carried out by black box

Units	System Testing	System Reaction	Test Results	Tested By
Sign	Enter an email address and wrong password.	Displays a message when it happens that error email address and word password not registered.	Succeed	Doctor And Patient
	Enter an email address and patient password in login form	Displays the page patient dashboard.	Succeed	Patient

System testing is carried out also in other parts related to the core unit of the system as well as the parts that are stuck in the system registration section as a whole as presented in table 4.

Table 4. Unit System Testing

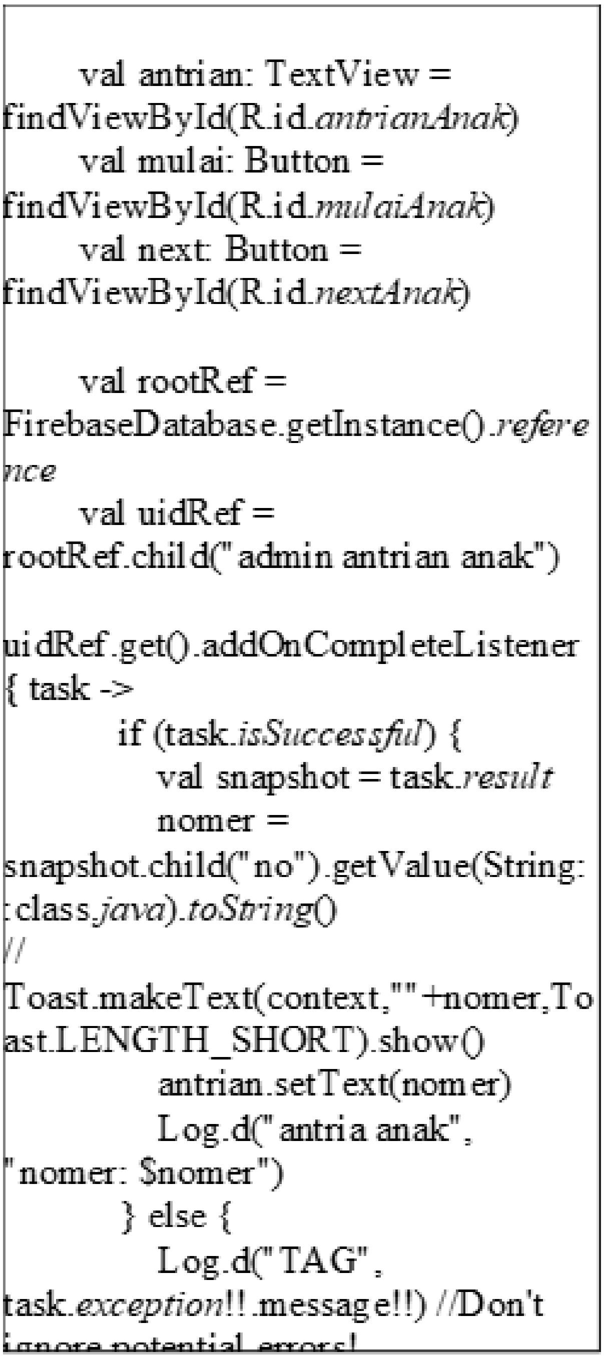
Units	System Testing	System Reaction	Test results	Tested By
Registration	Doctor When registering leave the required data.	Displays an error message that data must be filled in the column which is empty.	Succeed	Doctor
	When registering the patient, leave the required data blank.	Displays an error message that the data must be filled in empty column.	Succeed	Patient
	Fill in the email you used to register again as a customer.	Displays an error message that the field information cannot be empty and must be filled.	Succeed	Patient
Queue	The doctor added queue number.	Displays number queue	Succeed	Doctor
	The patient chooses a poly queue number	Displays number the poly queue that the patient chooses	Succeed	Patient
Ambulance	Added information ambulance	Succeed add.	Succeed	Doctor
	The patient chooses a poly queue number	Displaying information on the ambulance selected by the patient	Succeed	Patient

BlacBox Test Results

Program Code

Source Code Admin Queue Feature The admin queue feature is a feature for changing queues by poly doctor admins. The following is the source code for the admin queue feature.

a. Source Code User Queue Feature
   The user queue feature is a feature for viewing queues in real time by poly doctor admins. The following is the source code for the user queue feature.



```
     val antrian: TextView =
findViewById(R.id.antrianAnak)
     val mulai: Button =
findViewById(R.id.mulaiAnak)
     val next: Button =
findViewById(R.id.nextAnak)

     val rootRef =
FirebaseDatabase.getInstance().refere
nce
     val uidRef =
rootRef.child("admin antrian anak")

uidRef.get().addOnCompleteListener
{ task ->
        if (task.isSuccessful) {
            val snapshot = task.result
            nomer =
snapshot.child("no").getValue(String:
:class.java).toString()
//
Toast.makeText(context,""+nomer,To
ast.LENGTH_SHORT).show()
            antrian.setText(nomer)
            Log.d("antria anak",
"nomer: $nomer")
        } else {
            Log.d("TAG",
task.exception!!.message!!) //Don't
ignore potential errors!
```

Figure 13. Scripting Root

b. Maps Feature Source Code
   The maps feature is a feature to see the location of the nearest ambulance which has been added by the admin. The following is the source code for the maps feature following is the source code for the maps feature.

```
class PolimataActivity :
AppCompatActivity() { private lateinit var
context: Context override fun
onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContentView(R.layout.activity_polimata)
context = this val queueU: TextView =
findViewById(R.id .noeyeQueue) val rootRef
= FirebaseDatabase.getInstance().reference
val uidRef = rootRef.child("eyequeue")
uidRef.get().addOnCompleteListener { task -
> if (task.isSuccessful) { val snapshot =
task.result val number =
snapshot.child("no").getValue(String::class.ja
va)//
Toast.makeText(context,""+number,Toast.LE
NGTH_SHORT).show()
queueU.setText(number) Log.d ("eye queue",
"number: $number") } else { Log.d("TAG",
task.exception!!.message!!) //Don't ignore
potential errors! } }
```

Figure 14. Scripting Activity

c. Source Code for Login and Register Features
The login and register features are features for entering the application and will be directed to the first page of the application. The following is the login and register source code.

```
auth.signInWithEmailAndPassword(email,
password).addOnCompleteListener { if
(it.isSuccessful) {

val rootRef =
FirebaseDatabase.getInstance().reference val
ordersRef =
rootRef.child("user").orderByChild("email").e
qualTo(email) val valueEventListener = object
: ValueEventListener { override fun
onDataChange(dataSnapshot: DataSnapshot) {
for (ds in dataSnapshot.children) { test =
ds.child("level").getValue(String::class.java).t
oString() if(test == "user"){ val intent =
Intent(context, DashboardActivity ::class.java)
startActivity(intent) } else {
startActivity(Intent(context,
UserActivity::class.java)) } } }
override fun onCancelled(databaseError:
```

Figure 15. Scripting Sign

d. Source Code CRUD Website Features
The Crud feature on the website is a feature for entering ambulance information data which will be displayed on. The following is the login and registration source code.

```
private fun getLocationMarker() { try { val
stream = assets.open("sample_maps.json")
val size = stream.available() val buffer =
ByteArray(size) stream.read(buffer)
stream.close() val strContent =
String(buffer, StandardCharsets.UTF_8) try
{ val jsonObject = JSONObject(strContent)
val jsonArrayResult =
jsonObject.getJSONArray("results") for (i
in 0 until jsonArrayResult.length()) { val
jsonObjectResult =
jsonArrayResult.getJSONObject(i) val
modelMain = ModelMain() modelMain.strName =
jsonObjectResult.getString("name")
modelMain.strVicinity =
jsonObjectResult.getString("vicinity")

//get lat long val jsonObjectGeo =
jsonObjectResult.getJSONObject("geometry")
val jsonObjectLoc =
jsonObjectGeo.getJSONObject("location")
modelMain.latLoc =
jsonObjectLoc.getDouble("lat")
modelMain.longLoc =
jsonObjectLoc.getDouble("lng" )
modelMainList.add(modelMain) }
initMarker(modelMainList) } catch (e:
JSONException) { e.printStackTrace() } }
catch (ignored: IOException) {
Toast.makeText( this@AmbulanActivity,
"Oops, something went wrong. Try repeating
again in a few moments.",
Toast.LENGTH_SHORT ).show() }}

@SuppressLint("UseCompatLoadingForDrawables
")private fun initMarker(modelList:
List<ModelMain>) { for (i in
modelList.indices) { overlayItem =
ArrayList() overlayItem.add( OverlayItem(
modelList[i].strName, modelList[i]
.strVicinity, GeoPoint(
modelList[i].latLoc, modelList[i].longLoc )
) ) val info = ModelMain() info.strName =
modelList[i].strName info.strVicinity =
modelList[i].strVicinity
```

Figure 16. Scripting API

VI. CONCLUSION

After going through the application testing process along with the analysis and design stages, so that it becomes an application for monitoring queue lists and searching for the nearest ambulance that can be used by patients and doctors, it can be concluded after testing the application as follows:

The application can display information on the location of the nearest ambulance to the scene in Rangkasbitung by using the maps feature and information available in the application

This application provides the results of monitoring patient queue status in real time by using Firebase as a place to store data so that the data becomes real time data which can keep patient queue numbers connected to the queue numbers of poly doctors at Dr. Adjidarmo Regional Hospital.

REFERENCES

[1] Ardiansyah, MR (2022). Queuing System Application for Patients in Android-Based Outpatient Clinics (Case Study: Bengawan Medistra Prambanan Clinic) (Doctoral dissertation, UPN'Veteran" Yogyakarta).

[2] Bagaskara, M., Harris, A., & Nugroho, A. (2020). Android-Based Ambulance Service Application Design in Jambi City. Scientific Journal of Informatics Engineering Students, 1(4), 263-275.

[3] Fitrianto, AE, Maulindar, J., & Pradana, AI (2022, June). Design of a Mobile-Based Patient Queuing Application at the Surakarta MTA Pratama Clinic In Proceedings of the National Seminar on Information Technology and Business (pp. 481-487).

[4] Maiyendra, N.A. (2019). Design of a Tourism Tour Promotion Information System and Booking Tour Packages for the Kerinci Jambi Region at CV. Open Source Based Rinai. JURSIMA (Journal of Information Systems and Management), 7(1), 1-11.

[5] NOVA, Marda, et al. Android-Based Ambulance Service Application for Emergency Situations. 2018.

[6] Nuruzzaman, ZZ, Kharisma, AP, & Ananta, MT (2022). Development of an Android-based Gatoel Mojokerto Hospital Queue Service Application. IF (Journal of Informatics), 6(2), 187-196.

[7]Rusmawan, U. (2019). Final Project and Programming Thesis Writing Techniques. Elex media komputindo. https://books.google.co.id/books?id=3cSZDwA A QBAJ

[8]Yendrianof, D., Romindo, R., Sari, AN, Tantriawan, H., Putri, EE, Manuhutu,

[9]MA, Turaina, R., Defiariany, D., Putri, NE, Priyantoro, T., & others. (2022). Information Systems Analysis and Design. We Write Foundation. https://books.google.co.id/books?id=C65sEAA AQBAJ.