

Development of Client Server-Based Queuing Applications at The Samsat Gowa Office

Arsan Kumala Jaya^{1*}, Akbar Hendra², Muhammad Sabirin Hadis³, Muhammad Rizal⁴, Randy Angriawan⁵, Annisa Nurul Puteri⁶

¹Department of Digital Business, Nitro Institute of Business and Finance Makassar, Makassar, Indonesia

²⁻³Department of Informatics, Universitas Teknologi Akba Makassar, Makassar, Indonesia

⁴Department of Informatics, Universitas Dipa Makassar, Makassar, Indonesia

⁵⁻⁶Department of Informatics, Universitas Teknologi Akba Makassar, Makassar, Indonesia

Abstract

The government has established a SAMSAT (One-Stop Manunggal Administration System) office which has duties in the motor vehicle tax service. Queuing activities make people spend their time waiting. Therefore, it is necessary to create a queuing system that can inform the estimated queue time remotely. The queuing system does not require users to wait physically, so the waiting time needed to queue can be used by taxpayers to carry out their own personal and work activities that are more useful. Queuing application development aims to develop queuing service features from previous applications which only provide queued information on taxpayers. Features developed in the form of queuing data processing with different activities, monitoring information on time and queue status periodically. The research method used is SDLC (Software Development Life Cycle) with a prototype model. The test results using the Gray Box Testing method show that queuing applications in government agencies providing motorized vehicle tax services have been successfully developed.

Keywords: Queuing Applications, Client Server, SDLC Prototype Model.

Received: 10 August 2022

Revised: 13 August 2022

Accepted: 30 December 2022

Introduction

The increasing number of motorized vehicles in Gowa Regency is very influential in the receipt of the realization of Motor Vehicle Tax (PKB) and Motorized Vehicle Title Transfer Fee (BBN-KB). The gowa samsat office is a place for PKB and BBN-KB services for taxpayers. However, as the number of taxpayers in Gowa Regency increases, the number of queues at the Gowa Samsat Office also increases. Based on potential data as of December 31 2022, the number of motorized vehicles registered at the Gowa Samsat Office was 296,452 units (Wahidah, 2023).

Taxpayers who make PKB payments at the Gowa Samsat Office have different processing processes. The processing process is the process of validation/annual renewal at the address of the STNK of Gowa Regency (Local) or at the address of the Non-Gowa Regency STNK (Link), the process of changing plates or changing the 5-year STNK, the process of duplicating STNK, and the BBN-KB process. These different processing processes have an effect on the duration of PKB or BBN-KB services because taxpayers who have tickets in the first queue with the license plate replacement process have a longer processing time compared to taxpayers who have tickets in the second queue with the annual approval process. vehicle tax.

Management of the PKB and BBNKB service processes at the Gowa Samsat Office can be carried out optimally so that conventional management processes are no longer effective and efficient. To meet these needs, the government has established a queue application maintenance policy which includes the development of a queue application from the previous version of the queue application, where its use is expected to make the taxpayer service process at the Gowa Samsat Office more optimal, efficient and effective.

*Corresponding author.

E-mail address: arsankumalajaya@nitromks.ac.id (Arsan Kumala Jaya)



Queuing application development aims to develop queuing applications from previous versions by covering queuing data processing with different management processes to produce fast, neat and orderly PKB and BBKNB service processes.

Method

The development of this application uses the SDLC (Software Development Life Cycle) method using the prototype model. The following are the stages of the prototyping model: (1) needs analysis, (2) building a prototype, (3) evaluating the prototype, (4) coding the system, (5) testing the system, (6) evaluating the system and (7) using the system. This technique is often used if you do not really understand the system to be developed, so you need an overview of the system to be developed. Through prototyping techniques, developers can make prototypes before developing the actual system (Maulani et al., 2022).

Needs Analysis: In the early stages of development, it is necessary to analyze the database structure and the interface that will be built. Figure 1 is a structural figure of a conventional queuing application database.

ID	No	Tanggal	Nomor	No HP	Layanan	NIK	Email
20503 295	03-11-2020	DD0506B2	085240291993	Loket 1		No KTP	Email
20504 296	03-11-2020	DD1245A1	085299350022	Loket 1		No KTP	Email
20505 297	03-11-2020	DD1038AL	082348316461	Loket 1		No KTP	Email
20506 298	03-11-2020	DD5206L1	081343617009	Loket 1		No KTP	Email
20507 299	03-11-2020	DD4069KH	085343383599	Loket 1		No KTP	Email
20508 300	03-11-2020	DD1038AL	082291124081	Loket 1		No KTP	Email
20509 301	03-11-2020	DD3459B2	000000000000	Loket 1		No KTP	Email
20510 302	03-11-2020	DD4692LK	082393462041	Loket 1		No KTP	Email
20511 303	03-11-2020	DD2494L6	081342471900	Loket 1		No KTP	Email
20512 304	03-11-2020	DD1038AL	083447691981	Loket 1		No KTP	Email
20513 305	03-11-2020	DD6170L1	085344785983	Loket 1		No KTP	Email
20514 306	03-11-2020	DD3459L1	082189057506	Loket 1		No KTP	Email
20515 307	03-11-2020	DD1038AL	082397060821	Loket 1		No KTP	Email
20516 308	03-11-2020	DD1121BL	082338970750	Loket 1		No KTP	Email
20517 309	03-11-2020	DD5248L1	085338697794	Loket 1		No KTP	Email
20518 310	03-11-2020	DD1038AL	081356389042	Loket 1		No KTP	Email
20519 311	03-11-2020	DD1639VM	082447881721	Loket 1		No KTP	Email
20520 312	03-11-2020	DD1705VR	082345622486	Loket 1		No KTP	Email
20521 313	03-11-2020	DD4833NU	085242645401	Loket 1		No KTP	Email
20522 314	03-11-2020	DD0438C0	087729739668	Loket 1		No KTP	Email
20523 315	03-11-2020	DD2054L1	085299147029	Loket 1		No KTP	Email
20524 316	03-11-2020	DD1370BP	081342701227	Loket 1		No KTP	Email
20525 317	03-11-2020	DD1386L1	0813446311	Loket 1		No KTP	Email
20526 318	03-11-2020	DD03097U	082450318431	Loket 1		No KTP	Email
20527 319	03-11-2020	DD1052L1	081342779431	Loket 1		No KTP	Email
20528 320	03-11-2020	DD4503LP	085342219214	Loket 1		No KTP	Email
20529 321	03-11-2020	DD0438C0	082396847813	Loket 1		No KTP	Email
20530 322	03-11-2020	DD1073VM	085397858582	Loket 1		No KTP	Email
20531 323	03-11-2020	DD4832LK	085614837520	Loket 1		No KTP	Email
20532 324	03-11-2020	DD2494L6	000000000000	Loket 1		No KTP	Email
20533 325	03-11-2020	DD4794M2	085253294813	Loket 1		No KTP	Email
20534 326	03-11-2020	DD1389L1	000000000000	Loket 1		No KTP	Email
20535 327	03-11-2020	DD4186L1	000000000000	Loket 1		No KTP	Email
20536 328	03-11-2020	DD1432BK	085216461814	Loket 1		No KTP	Email
20537 329	03-11-2020	DD4833NU	08667159049	Loket 1		No KTP	Email
20538 330	03-11-2020	DD1406L1	085242314481	Loket 1		No KTP	Email
20539 331	03-11-2020	DD6275NV	085242275206	Loket 1		No KTP	Email
20540 332	03-11-2020	DD4057BP	082187022129	Loket 1		No KTP	Email
20541 333	03-11-2020	DD1544NV	085238288854	Loket 1		No KTP	Email
20542 334	03-11-2020	DD1544NV	089069461201	Loket 1		No KTP	Email

Figure 1. The Conventional Queuing Application Database.

Figure 1 shows a database view of a conventional queuing application with a DataPlat table. Microsoft Access is a database tool that is used with a table structure in the form of ID, No, Tanggal, No HP, Layanan, NIK and Email. The next stage is the development of the application interface design. This design includes: (1) display design, (2) reporting design, and (3) form design.



Figure 2. Conventional Queue Application Interface Display

Figure 2 is the display design figure of a conventional queuing application. In this figure, the input process still uses one queue data input button, namely PENDAFTARAN/PENETAPAN/ KOREKTOR. This button is used for all PKB and BBN-KB service processes. In figure 3 is the reporting view design figure from the previous application. While figure 4 is the display of the input form for entering queue data into the data base..

No	Tanggal	Nomor	No HP	Layanan	NIK	Email
2	30/10/20	DD345	89000000000	Loket 1	No KTP	Email
3	02/11/20	DD234LY	00000000000	Loket 1	No KTP	Email
4	02/11/20	DD4103LL	00000000000	Loket 1	No KTP	Email
5	02/11/20	DD3306OY	00000000000	Loket 1	No KTP	Email
6	02/11/20	DD1626UY	081342977226	Loket 1	No KTP	Email
7	02/11/20	DD5171NU	082291266803	Loket 1	No KTP	Email
8	02/11/20	DD3893LY	082188859506	Loket 1	No KTP	Email
9	02/11/20	DD3845LY	085342906266	Loket 1	No KTP	Email
10	02/11/20	DD6874LK	085211832419	Loket 1	No KTP	Email
11	02/11/20	DD4646VR	085396202177	Loket 1	No KTP	Email
12	02/11/20	DD5617OD	082192351911	Loket 1	No KTP	Email
13	02/11/20	DD3631BN	085342594056	Loket 1	No KTP	Email
14	02/11/20	DD3346YQ	081341324032	Loket 1	No KTP	Email
15	02/11/20	DD1570LR	085242499907	Loket 1	No KTP	Email
16	02/11/20	DD1124BY	082250633182	Loket 1	No KTP	Email
17	02/11/20	DD1020LI	081355859729	Loket 1	No KTP	Email
18	02/11/20	DD6338LL	081288943259	Loket 1	No KTP	Email
19	02/11/20	DD6368BI	081229823998	Loket 1	No KTP	Email
20	02/11/20	DD5078LL	082349887477	Loket 1	No KTP	Email
21	02/11/20	DD1812YN	085243194635	Loket 1	No KTP	Email
22	02/11/20	DD4905BR	085397286070	Loket 1	No KTP	Email
23	02/11/20	DD3381BU	085342317717	Loket 1	No KTP	Email

Figure 3. Display of Conventional Queue Application Reporting

Figure 4. Display of the Conventional Queue Application Input Form

Building Prototypes: Based on application requirements, the prototype of the proposed application can be seen in the use case diagram and UML (Unified Modeling Language) diagram. UML diagrams are used in the analysis, construction and maintenance of software systems (Bergström et al., 2022).

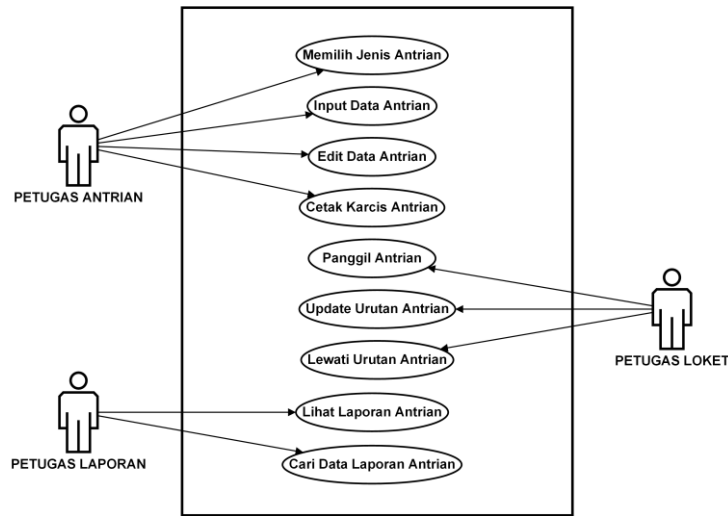


Figure 5. Use Case Diagram

The use case diagram of the proposed application is shown in figure 5. First, the queuing officer performs activities to select the type of queue needed by the taxpayer. Then input and edit data and print the taxpayer queue receipt. Second, the counter clerk calls the queue, updates the queue data, and skips the queue. Third, the report officer looks at the report and registers the queue data. Taxpayers can see the status of the queue on the monitor, as shown in figure 13. Queuing officer activity can be seen in figure 6, counter officer activity in figure 7, and reporting officer activity in figure 8

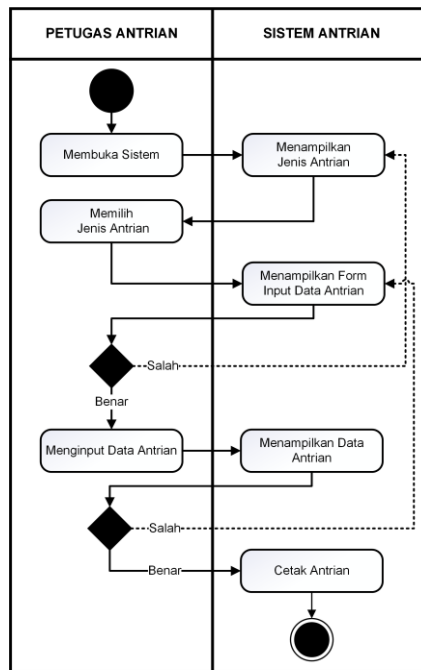


Figure 6. Queue Officer Activity

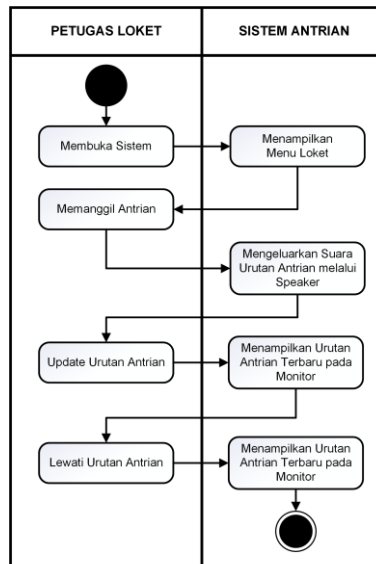


Figure 7. Counter Officer Activity

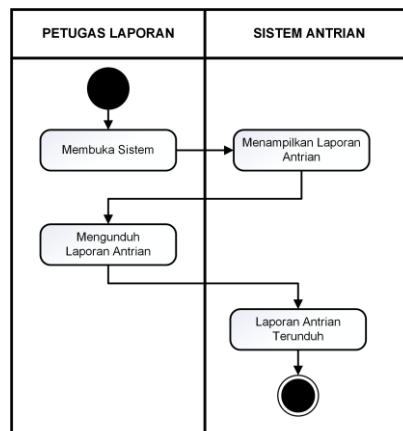


Figure 8. Queue Officer Activity

Prototype Evaluation: Evaluation of the prototype follows the use case diagrams and UML diagrams that have been built based on the needs of the proposed system.

Encoding System: The design of conventional queuing applications uses a desktop-based programming language with a Microsoft Access database. While the development of the proposed queuing application uses a web-based programming language, namely PHP version 7.3.21, Bootstrap version 4.4.1, JQuery version 3.1.1, and MySQL database version 10.1.31. CodeIgniter is an MVC (Model View Controller) based framework used for application development. Apart from that, CodeIgniter can also build dynamic PHP applications on the CRUD (Create Read Update Delete) project model (Muqorobin & Rozaq Rais, 2022).

Testing the System: The system testing method used is Gray Box Testing by combining Black Box Testing and White Box Testing methods. Testing using gray box testing shows better test results compared to black box testing (Sachtleben & Peleska, 2022). Where is black box testing for external system structure functions such as from each page button on the application interface display. While white box testing is for internal system structure functions such as calling queue functions and monitoring queue status.

System Evaluation: System evaluation to check the system as a whole according to the test results using gray box testing.

Using the System: Implementation of the system in government agencies providing motorized vehicle tax, namely the Gowa Samsat Office.

Results and Discussion

Result

Based on the results of the analysis on the table structure of the conventional queuing application database, it is necessary to develop a table structure to suit the needs of the proposed system model. Some of the items added are: (1) ID code design, (2) field selection, and (3) additional data design. Changes to the database structure can be seen in figure 9. The database used has 2 tables, namely data and status. The data table is for storing queue data while the status table is for storing vehicle queue status information. The structure of the data table and status table can be seen in tables 1 and 2.



Figure 9. Database of Queing Application

Table 1. Structure of Data Tabel

Name	Type	Size	Primary Key / Auto Increment
id	INT	11	YES
no_antrian	VARCHAR	10	NO
nopol	VARCHAR	10	NO
no_hp	VARCHAR	15	NO
nik	BIGINT	20	NO
proses	VARCHAR	50	NO
datetime	DATETIME		NO

Table 2. Structure of Status Tabel

Name	Type	Size	Primary Key / Auto Increment
id	INT	11	YES
no_antrian	VARCHAR	10	NO
loket	VARCHAR	1	NO
t_antri	DATETIME		NO
1A	VARCHAR	20	NO
1B	VARCHAR	20	NO
1C	VARCHAR	20	NO
t_daftar	DATETIME		NO
1D	VARCHAR	20	NO
1E	VARCHAR	20	NO
1DE	VARCHAR	20	NO
t_tetap	DATETIME		NO
2A	VARCHAR	20	NO
2B	VARCHAR	20	NO
2C	VARCHAR	20	NO
t_bayar	DATETIME		NO
2D	VARCHAR	20	NO
2E	VARCHAR	20	NO
2DE	VARCHAR	20	NO
t_final	DATETIME		NO

Queuing application development using a client/server system. Module development in this application is divided into several module parts, namely home, reset, view, and call counters. The home and reset modules serve as server side. While the view and call counter modules serve as the client side. (1) The Home module is the dashboard of the application. This module can be seen in figure 10. (2) The Reset Module is a queuing data editing module in queuing applications. See figure 11. (3) The View module functions as monitoring information from queuing data. In this module there are several submodules namely data view, status view and report view. See figures 12, 13 and 14.

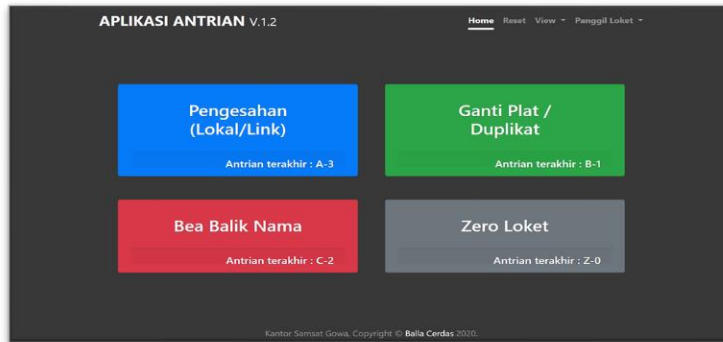


Figure 10. Display of the Home Module Interface

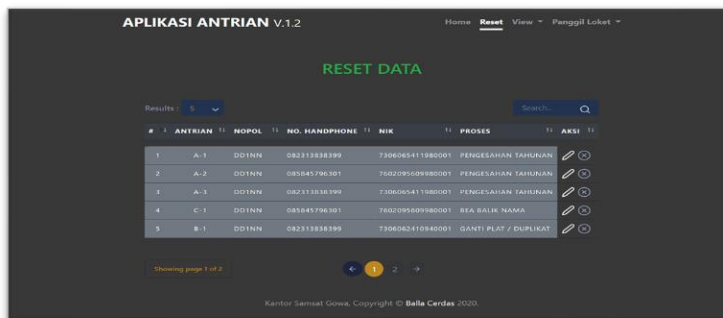


Figure 11. Reset Module Interface Display



Figure 12. Data View Module Interface Display

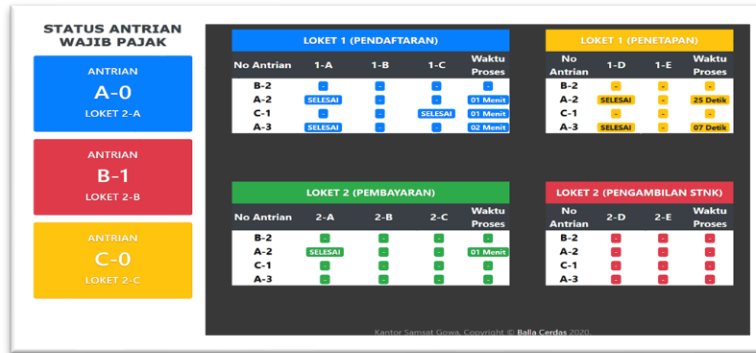


Figure 13. Display Interface Status View Module

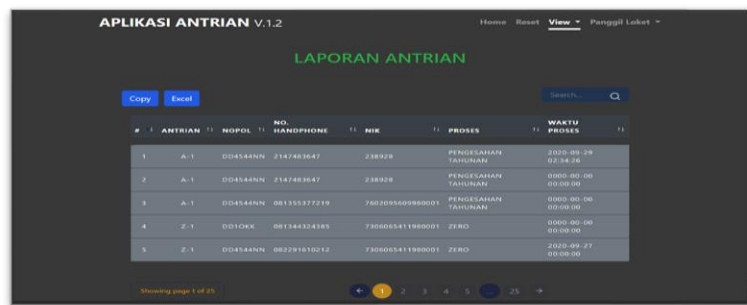


Figure 14. Display of the Report View Module Interface

The results of the development of the input form interface display in figure 4 can be seen in figure 15. In figure 4, the queue service still uses one type of queue. Figure 15 provides a solution for increasing queue services by developing four types of queue counters, namely validation counters (local/link), plate change/duplicate, new vehicles (BBN) and incomplete file requirements (zero).

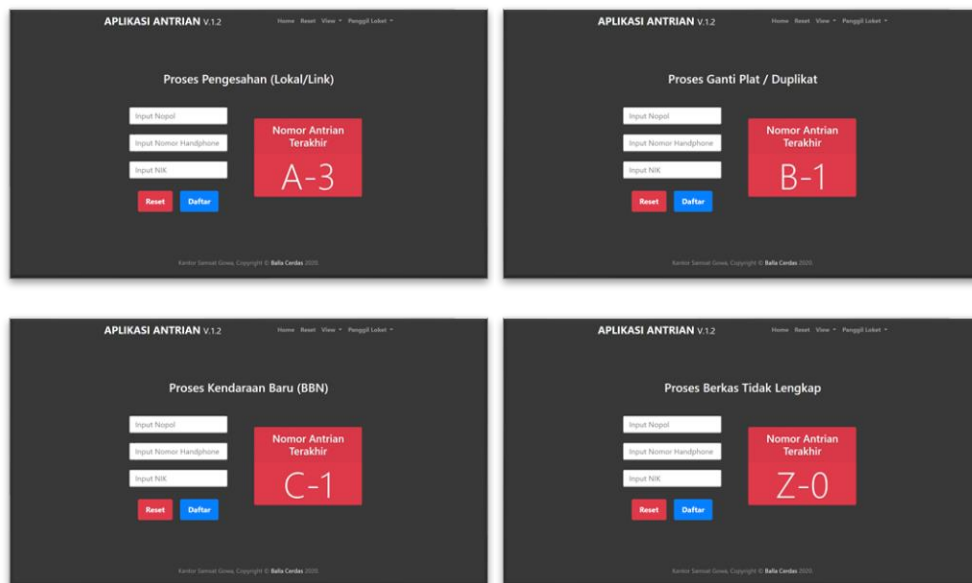


Figure 15. Display of the Input Form

In addition to the development of the queue counter type feature, this application has a call queue feature compared to the previous feature which only printed queue tickets. In the queue calling counter module, there are several calling counter modules for each tax official. (1) Registration Module with three calling counters namely counters 1-A, 1-B and 1-C. (2) Designation Modules, namely counters 1-D and 1-E. (3) Payment Modules, namely counters 2-A, 2-B and 2-C. (4) STNK/Plat Submission Modules, namely 2D and 2E counters. The following is the interface display of the queue counter calling module in figure 16.

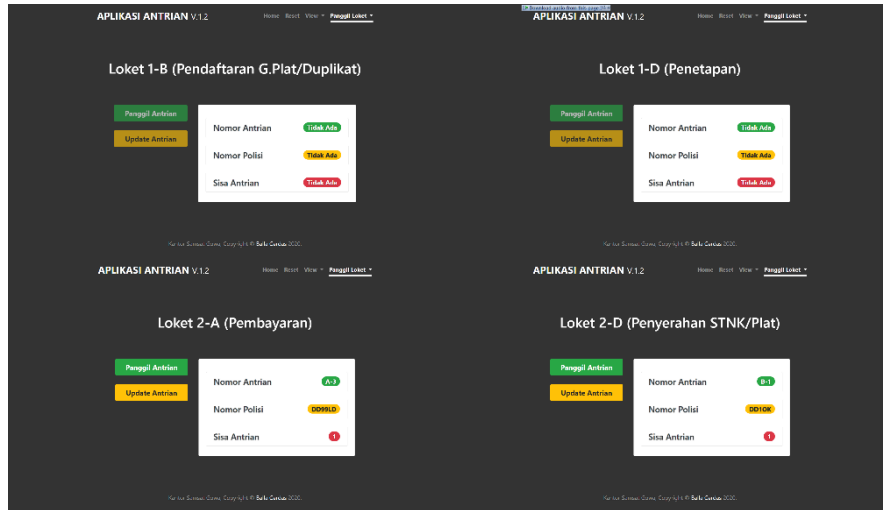


Figure 16. Queue Call Counter Interface Display

Discussion

The proposed queuing application system has been successfully evaluated and is running according to the expected test results. This application has several advantages over previous versions. Among them can monitor the estimated service processing time live on the waiting room monitor. Another advantage of this application is that it can call queues and is also able to manage the time for vehicle tax service work with different durations of work. This application can also change queue data if there is an input data error from the clerk. In addition, this application can be controlled on a smartphone or device that can connect to WiFi because it uses a client-server based network.

Conclusions and Suggestions

Conclusions

The proposed queue application is a queue application version 1.2 due to the development of the queue application that has been installed at the Samsat office in the Gowa region. This application provides several advantages in the form of features that can provide information to taxpayers because it provides information on the estimated service time for each service counter in a transparent manner. In addition, this application also increases the service duration time more quickly because the services are grouped based on the type of queue service.

Suggestions

This research is applied research related to application development from existing applications. Researchers hope that this research can become additional literature related to application development techniques using the SDLC research method with the prototype model and the gray box testing method. Improving application-based community services in the future will experience new challenges. Therefore, prepare analytical skills and problem solving by utilizing technology that is also in line with developments, especially in the digitalization era.

Contribution: Arsan1: Conceptualization, design, analysis, writing. Akbar2: Editing/reviewing, supervision. Sabirin3: design, analysis, supervision. Rizal4: Editing/reviewing, supervision. Randy5: Editing/reviewing, supervision. Annisa5: Editing/reviewing, supervision.

References

- Bergström, G., Hujainah, F., Ho-Quang, T., Jolak, R., Rukmono, S. A., Nurwidyantoro, A., & Chaudron, M. R. V. (2022). Evaluating the layout quality of UML class diagrams using machine learning. *Journal of Systems and Software*, 192. <https://doi.org/10.1016/J.JSS.2022.111413>
- Maulani, S., Firdaus, E. A., Syani, M., & Kirana, Y. (2022). PERANCANGAN DAN IMPLEMENTASI APLIKASI INFORMASI KEGIATAN MASJID (STUDI KASUS: MASJID AL BARAKAH). *NUANSA INFORMATIKA*, 16(2), 55–65. <https://doi.org/10.25134/NUANSA.V16I2.5798>
- Muqorobin, M., & Rozaq Rais, N. A. (2022). Comparison of PHP Programming Language with Codeigniter Framework in Project CRUD. *International Journal of Computer and Information System (IJCIS)*, 3(3), 94–98. <https://doi.org/10.29040/IJCIS.V3I3.77>
- Sachtleben, R., & Peleska, J. (2022). Effective grey-box testing with partial FSM models. *Software Testing Verification and Reliability*, 32(2). <https://doi.org/10.1002/STVR.1806>
- Wahidah. (2023). *Data Potensi Pajak Kendaraan Bermotor UPT Pendapatan Wilayah Gowa*.