The Implementation of Genetic Algorithm in The Vehicle KIR Test Scheduling System

Suprihadi[*1], Rama Aufadha Risqullah[2]
[1,2]Informatics Engineering Study Program, Faculty of Information Technology, Satya Wacana Christian University
E-mail: [*1]**suprihadi@uksw.edu**, [2]aufadhar@gmail.com

Abstract

Based on the results of observations and data collection, the vehicle parking test services at the Salatiga City Transportation Agency have not implemented a structured scheduling system in managing their service schedules. This has resulted in an accumulation of the number of vehicles that exceed the service capacity on a working day. Therefore, a scheduling system is needed to assist the agency in arranging the due time schedule for the KIR test of a vehicle, and helping the vehicle owner community to find out the due time for the next KIR test. The scheduling method used is a genetic algorithm that is applied to a web-based application. The application is built using the PHP language and MySql database. With this website-based scheduling system application, it has made it easier for service officers to manage the KIR test service schedule, especially if there is a change in the number of vehicles.

Keywords — *Genetic Algorithm, Scheduling, KIR Test*

1. INTRODUCTION

Scheduling is an activity that must be owned by everyone to assist activities in everyday life. Especially an agency or institution that has important activities that must be carried out regularly and neatly. The importance of scheduling can run as planned. With the existence of a process called scheduling, the scheduling process is a difficult process because it requires accuracy in managing sufficient time so that an activity does not clash with other activities.

In this case, the case study taken to design and create a vehicle delivery scheduling system at the Salatiga City Transportation Agency greatly affects the course of the vehicle parking test activities that are scheduled for maturity. However, there is an obstacle, namely the absence of a computerized system to help inform vehicle owners and the scheduling of a vehicle with a maximum capacity of vehicle steering test services which can only serve 30 vehicles per day. Because there are problems regarding the vehicle parking test schedule in service every day, an application is needed to optimize the scheduling system in vehicle steering test service activities. Based on these problems, it will be discussed how to solve these problems using a method, namely genetic algorithms as a scheduling system optimization system in carrying out activities in an optimal and structured manner.

Several research topics that have been carried out in utilizing genetic algorithms to solve scheduling problems are used as references for this study. This Genetic Algorithm is good enough to be used to solve big problems with a good solution, even though this problem requires a long execution time when done manually [1]. There are three advantages of the application of genetic algorithms in the optimization process, namely genetic algorithms do not require too many mathematical requirements in completing the optimization process. The evolutionary operation of the Genetic Algorithm is very effective for observing random global positions. Genetic Algorithms have the flexibility to be implemented efficiently for certain problems [2].

In a study entitled Lecture Scheduling System Using Gentika Algorithm Method (Case Study of the Faculty of Medicine and Health, Muhammadiyah University of Jakarta),

explained that the lecture scheduling system relates to the optimization of limited space with limited lecturer readiness. The results of this study are a website-based system that functions to facilitate academics in making and presenting lecture schedule information for lecturers and students [3]. Another study entitled Building a Scheduling Information System with the Genetic Algorithm Method at the Informatics Engineering Laboratory, Muhammadiyah University of North Maluku. This study makes a Web application to optimize computerized scheduling that can replace manual methods and produce a more accurate schedule with the existing rules in a shorter time using the PHP programming language [4]. In principle, both studies are that genetic algorithms are applied in the process of making a scheduling so that it is easy, precise, and effective. A similar study was also conducted by [5] [6].

To find out what the needs of the Salatiga City Transportation Office in conducting a vehicle driving test, an interview was conducted. Interviews were conducted at the Salatiga City Transportation Agency through the head of the vehicle feasibility test section regarding the vehicle steering test mechanism, starting from the registration process to the vehicle registration process in accordance with the procedures and test standards set by the Ministry of Transportation. The Salatiga City Transportation Agency provides services to the public, especially those who have public vehicles or cargo vehicles, to test their vehicles as roadworthy or not by providing a vehicle registration book.

However, there are several problems regarding this vehicle test where vehicle owners often forget the validity period of their vehicle test. At this time, the Salatiga City Transportation Agency does not yet have a scheduling system that provides information to vehicle owners when the validity test period will expire and the vehicle delivery test service is limited, which only serves 25-30 vehicles per day. A genetic algorithm is a method commonly used in this scheduling problem. With a scheduling system that uses genetic algorithms, it will provide a solution to the problems that occur.

2. RESEARCH METHOD

At the research stage, namely the identification of problems that match the needs or problems that occur at the Salatiga City Transportation Agency, namely the problem of vehicle driving tests. Next is the collection of data needed to conduct research or research, namely data on vehicles that have been tested. Next is the system design stage, namely the system design process using UML, including use case diagrams, activity diagrams. At the stage of making the system, the system is built on a web basis using the MVC architecture, the PHP programming language with the Codeigniter framework, and MySQL as a database. The next stage is the implementation and testing of the system, that is, the system always makes adjustments when there are deficiencies in the validation results during the system test. This is done so that the system is developed according to user needs, namely the Salatiga City Transportation Agency.


Figure 1. Research Flow

Use Case diagrams are used to explain how actors or users relate to subsystems in the application system. The use of this diagram can describe a system by entering functions that exist in the subsystems that work in a flow so that it can clarify the relationship of actors in running the application system. In this system, there is an admin actor who has the right to access the system in full which includes data processing on the system dashboard, admin management, vehicle management, delivery management, completion of delivery, scheduling. The admin has full access rights such as adding data, editing data, looking for data, submitting delivery, and sending notifications to vehicle owners who will carry out tail tests and manage automated scheduling processes in the system with a genetic algorithm.
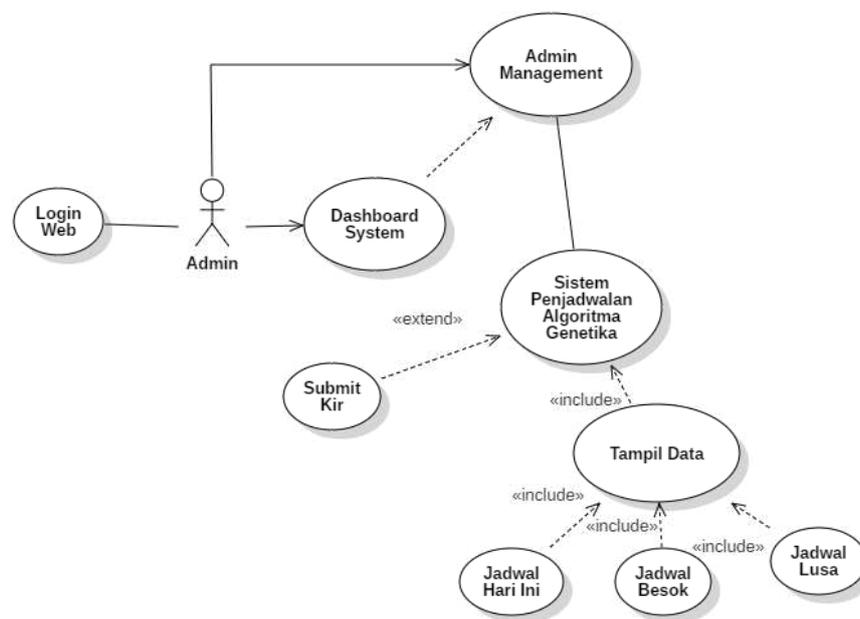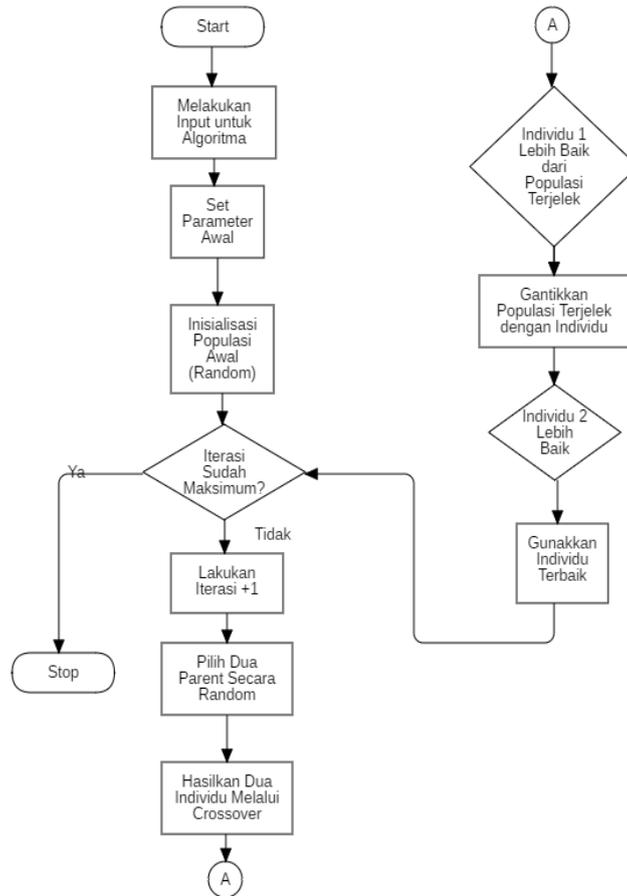

Figure 2. Use case diagram of Genetic Algorithm Scheduling

Activity diagrams are used to define the process flow of the user's relationship with the system in the application. In the Vehicle Delivery Test Scheduling application, there are activities that can be explained from the user's relationship with the application, and the application with the database. The core process of the Kir test scheduling application is the process of retrieving data processed by genetic algorithms from the database to the display view scheduling data for today, tomorrow, and the next 2 days.
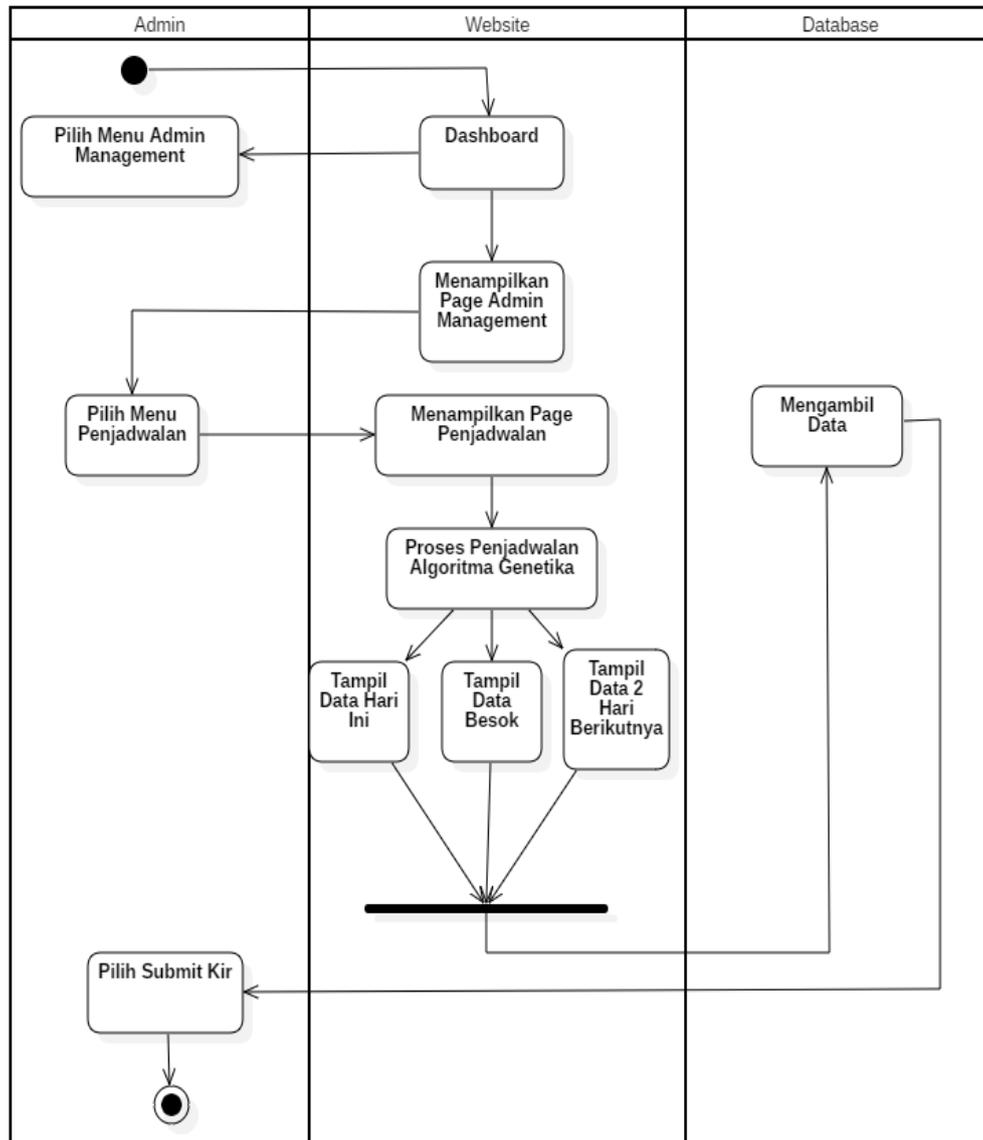
Figure 3. Activity diagram of Scheduling Genetic Algorithms

In Figure 3, it explains that the admin part takes an action selecting the Admin Management menu which leads to a scheduling system that has been designed using a genetic algorithm, where the data will appear on the vehicle scheduling data display page.

Figure 4. Flowchart diagram of Genetic algorithm KIR Test Scheduling

Figure 4 is a flowchart diagram of the genetic algorithm in the vehicle steering test scheduling. There are several calculation stages in carrying out the process of scheduling the test car, namely to input, vehicle data such as vehicle numbers, vehicle test numbers, and the validity period of the test. The initial parameters used in the genetic algorithm are population size, maximum iteration, and the probability of a crossover or crossbreeding process. Then this initial population initialization is filled randomly according to the needs of the problem as in the case of this test scheduling, which is taken with values of 30, 50, and 100. Each population that has been determined will be iterated as many times as we want, each population will repeat as many iterations as entered. The function of the iteration is to find the best population. After all these processes are completed, the application will display the scheduling result data closer to the due date of the Kir test.

Pseudocode of Genetic Algorithms

1. Initialize the N chromosome population
2. Loop
3.   Loop for the N chromosome
4.   Chromosome decode
5.   Chromosome evaluation
6.   End
7.   Make a copy or two of the best chromosomes
8.   Loop until you get the best N chromosome
9.   Select two chromosomes
10.   *Crossover*
11.   Mutation
12.   End
13. End

There are several main components in genetic algorithms, namely initialization procedures, coding techniques, evaluation functions, selection, genetic operators, parameter determination, and population replacement.

3. RESULTS AND DISCUSSION

3.1. Experimental Parameters

This parameter combination experiment uses the Design Of Experiment (DOE) method. To determine the right combination, it is necessary to study the parameters for all three parameters (population size, maximum iteration, and probability of crossover) as experimental factors, by comparing the mean of the final results (standard deviation) [7]. DOE for parameters was tested using the Minitab17 software after 270 trials were carried out, the best parameters were generated to be implemented into the vehicle steering test scheduling system.

Table 1. Best Parameter Combinations

No	Factor	Score
1.	Population Size (Ukpop)	100
2.	Maximum interaction (MaxG)	5.000
3.	Probability of crossover (Pc)	0,95

In the analysis of variance (ANOVA), the probability value of each factor serves to determine whether the level change of these factors has a significant effect on the output, namely the standard deviation. If the probability value is less than 0.05, this factor has a significant effect on output. The probability value of the Ukpop factor is 0.00, MaxG 0.111, Pc 0.00. This shows that the Ukpop (population) and Pc factors have an effect on the settlement algorithm.

The P-value for the interaction between the Ukpop and MaxG factors was 0.296, the Ukpop and PC factors were 0.001, the MaxG and Pc factors were 0.762, and the interaction between the three factors was 0.001. The probability value above 0.05 indicates that there is no interaction at each level of factors that can have a significant effect on output. So the factors that indicate an interaction are the Ukpop factor with Pc and these three factors.

Results for: Worksheet 2

General Linear Model: std.deviasi versus Ukpop; MaxG; Pc

```
Factor    Type    Levels  Values
Ukpop     fixed      3    1; 2; 3
MaxG      fixed      3    1; 2; 3
Pc        fixed      3    1; 2; 3

Source          DF    Seq SS      Adj SS      Adj MS       F       P
Ukpop            2  0,0070650   0,0070650   0,0035325   24,52   0,000
MaxG             2  0,0006393   0,0006393   0,0003196    2,22   0,111
Pc               2  0,0133567   0,0133567   0,0066784   46,36   0,000
Ukpop*MaxG       4  0,0007121   0,0007121   0,0001780    1,24   0,296
Ukpop*Pc         4  0,0028930   0,0028930   0,0007233    5,02   0,001
MaxG*Pc          4  0,0002678   0,0002678   0,0000669    0,46   0,762
Ukpop*MaxG*Pc    8  0,0038566   0,0038566   0,0004821    3,35   0,001
Error          243  0,0350051   0,0350051   0,0001441
Total          269  0,0637956
```

Figure 5. Results of Experimental Analysis of Standard Deviation Parameters

3.2. Application Implementation

After several design stages are carried out in the development of this system, the results of the system that are made will then be discussed. The following are some of the website-based system views that were made based on the design that was done previously.

Figure 6. Display Login System KIR Test Scheduling

In Figure 6 this page can be accessed by admin actors and user actors. The admin actor is an employee of the Salatiga City Transportation Agency, while the user actor is the owner of the vehicle. The start page displays a login page that requires the correct username and password to be able to access the actor's dashboard page.

Figure 7. Admin Actor and User Actor Dashboard Page Display.

In Figure 7 this page contains all the data that has been input by the admin that has entered the database, namely vehicle number, test number, vehicle type, year of manufacture, vehicle owner, due date.

Figure 8. Admin Management Page Views

Figure 8 on this page contains submenus such as vehicle management, delivery management, delivery completion, scheduling in which the admin actor can perform (create, update, delete) data.

Figure 9. Scheduling Page Views

Figure 9 on this page shows the results of scheduling using a genetic algorithm that has been implemented in a scheduling system application that performs a Kirir test on a predetermined day. This scheduling system has automatically adjusted the date on the computer system (current time) which will update automatically according to the day and date. If there is no vehicle delivery test schedule that day, the system will automatically enter what the admin entered at that time. In one day the system has limited what has been determined by the Salatiga Transportation Agency, which only serves 20-30 vehicles per day. If the vehicles accumulate or cross the limits set by the system, the vehicle data will be thrown into the tomorrow's menu, and so on.

Figure 10. Scheduling Tomorrow Page Views

In Figure 10, this page displays the results of scheduling using a genetic algorithm that has been implemented in an application that performs a test tomorrow. In this scheduling menu, there are two actions that the admin can take, namely submitting delivery and sending notifications to vehicle owners.

Program Code 1. Initialization Procedure

```
1   public function Inisialisasi()
2     {
3         $jumlah_pegawai = count($this->pegawai);
4         $jumlah_jam = count($this->jam);
5         $jumlah_hari = count($this->hari);
6         $jumlah_ruang_reguler = count($this->ruangReguler);
7         for ($i = 0; $i < $this->populasi; $i++) {
8           for ($j = 0; $j < $jumlah_pegawai; $j++) {
9             $kendaraan = $this->kendaraan[$j];
10            $this->individu[$i][$j][0] = $j;
11            if ($kendaraan == 1) {
12              $this->individu[$i][$j][1] = mt_rand(0,  $jumlah_jam - 1);
13            }
14            if ($kendaraan == 2) {
15              $this->individu[$i][$j][1] = mt_rand(0, ($jumlah_jam - 1) - 1);
16            }
17            if ($kendaraan == 3) {
18              $this->individu[$i][$j][1] = mt_rand(0, ($jumlah_jam - 1) - 2);
19            }
20            if ($kendaraan == 4) {
21              $this->individu[$i][$j][1] = mt_rand(0, ($jumlah_jam - 1) - 3);
22            }
23            $this->individu[$i][$j][2] = mt_rand(0, $jumlah_hari - 1);
24            if ($this->jenis_kendaraan[$j] === $this->KIR) {
                    $this->individu[$i][$j][3] = intval($this->ruangKir[mt_rand(0,
25   $jumlah_ruang_kir 1)]);
```

```
26            }
27          }
28       }
```

Program code 1 is an initialization procedure, which determines what will be the input for calculating the genetic algorithm. There are a number of parameters for the initialization procedure, namely employees, hours of work, days, and working room.

Program Code 2. The Process of Calculating Fitness

```
1   public function HitungFitness()
2     {
3        for ($indv = 0; $indv < $this->populasi; $indv++)
4        {
5           $fitness[$indv] = $this->CekFitness($indv);
6        }
7
8        return $fitness;
9     }
```

Program code 2 is the process of calculating the fitness value in which this process uses the looping function. In this process, individuals will be checked one by one for their maximum or best value, if the best value has been found, there will be a parent (parent), then cross-bred (crossover) to find the best solution from the mutation results. After that there is a selection to produce a new individual.

Program Code 3. Crossover Process

```
1    public function StartCrossOver()
2      {
3         $individu_baru = array(array(array()));
4         $jumlah_pegawai = count($this->pegawai);
5         for ($i = 0; $i < $this->populasi; $i += 2)
6         {
7           $b = 0;
8           $cr = mt_rand(0, mt_getrandmax() - 1) / mt_getrandmax();
9           if (floatval($cr) < floatval($this->crossOver)) {
10             $a = mt_rand(0, $jumlah_pegawai - 2);
11             while ($b <= $a) {
12                $b = mt_rand(0, $jumlah_pegawai - 1);
13             }
14             for ($j = 0; $j < $a; $j++) {
15               for ($k = 0; $k < 4; $k++) {
16                  $individu_baru[$i][$j][$k]     = $this->individu[$this->induk[$i]][$j][$k];
17                  $individu_baru[$i + 1][$j][$k] = $this->individu[$this->induk[$i + 1]][$j][$k];
18               }
19             }
```

```
20          for ($j = $a; $j < $b; $j++) {
22              $individu_baru[$i][$j][$k]    = $this->individu[$this->induk[$i + 1]][$j][$k];
23              $individu_baru[$i + 1][$j][$k] = $this->individu[$this->induk[$i]][$j][$k];
24          }
25          }
26          for ($j = $b; $j < $jumlah_pegawai; $j++) {
27            for ($k = 0; $k < 4; $k++) {
28              $individu_baru[$i][$j][$k]    = $this->individu[$this->induk[$i]][$j][$k];
29              $individu_baru[$i + 1][$j][$k] = $this->individu[$this->induk[$i + 1]][$j][$k];
30            }
31          }
32        } else {
33          for ($j = 0; $j < $jumlah_pegawai; $j++) {
34            for ($k = 0; $k < 4; $k++) {
35              $individu_baru[$i][$j][$k]    = $this->individu[$this->induk[$i]][$j][$k];
36              $individu_baru[$i + 1][$j][$k] = $this->individu[$this->induk[$i + 1]][$j][$k];
37            }
38          }
39        }
40      }
```

Program code 3 is a crossover process. In this process, the best group of individuals will be crossover between the parent and the newborn from the fitness calculation process to replace the parent with a bad value in the selection process, so that a population with the best fitness value can be produced. When the random value is more than the probability value, the new schedule is the same as the selected schedule.

In this vehicle delivery test scheduling system, the genetic algorithm works in accordance with the flowchart that has been designed, where the initialization procedure is to determine the parameters. In this scheduling system application, the genetic algorithm plays a very important role, where after the data is entered, the system will immediately calculate or process the data automatically and the data will be immediately displayed on the scheduling page. If the calculation is not in accordance with the specified procedure, then on the scheduling page, such as there is a stocking schedule, the data that occurs will be transferred to tomorrow's scheduling menu with a limit of 30 vehicle data. If there is no vehicle parking test schedule on that day, the system defaults to retrieve the data entered on that day.

System test is a stage in research that aims to determine application requirements. Application testing is done by testing the features and functions of the application (black box method), it can be seen in table 2 that the application has been running well. This test can also find out whether there is an error in the application system that has been made [8].

Table 2. Black Box Testing of Vehicle Delivery Test Scheduling System

No.	Function tested	Input Data	Output Data	Test Results	Status
1	Test the Website application login page	Username and Password	Successfully Login	Success	Valid
2	Input vehicle data	A form filled correctly	Successfully insert data	Successfully added new data	Valid
3	Update vehicle data	A form filled correctly	Successfully update data	Successful data editing	Valid

4	Vehicle Scheduling using genetic algorithms	The test scheduling table is filled in	Displays the vehicle delivery test schedule	Applications can provide vehicle scheduling information along with the KIR validity period.	Valid
5	Submit Kir	The vehicle performs a delivery test and the data is stored in the database	Successfully sending a message the vehicle has carried out the delivery test	Successfully sending a message the vehicle has carried out the delivery test	Valid
6	Whatsapp notifications	The test schedule is out and is approaching the test day	Successfully send messages and notifications to vehicle owners	Successfully sending messages and notifications	Valid

4. CONCLUSION

Based on the research conducted, it can be concluded that scheduling through web-based applications can overcome the occurrence of clashes between one vehicle delivery test schedules. Thus, this application helps employees of the Salatiga City Transportation Service in making vehicle registration test schedules with limited service delivery, which can only serve 25 to 30 vehicles per day. This application can also display the validity period of the driver so that it helps vehicle owners in managing their vehicle's work schedule.

5. SUGGESTED

Suggestions for development are the addition of parameters in the vehicle parking test service, including parameters for the parking area and employees so that service efficiency can be increased as much as possible.

6. REFERENCES

[1] A. Jain, D. S. Jain, and D. P. K. Chande, "Formulation of Genetic Algorithm to Generate Good Quality Course Timetable," Int. J. Innov. Manag. Technol., vol. 1, no. 3, pp. 248–251, 2010, [Online]. Available: http://ijimt.org/papers/46-M431.pdf.

[2] W. A. Puspaningrum, A. Djunaidy, and R. A. Vinarti, "Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika di Jurusan Sistem Informasi ITS," vol. 2, no. 1, pp. 127–131, 2013.

[3] A. Laksono, M. Utami, and Y. Sugiarti, "Sistem Penjadwalan Kuliah Menggunakan Metode Algoritma Genetika (Studi Kasus: Fakultas Kedokteran Dan Kesehatan Universitas Muhammadiyah Jakarta)," Stud. Inform. J. Sist. Inf., vol. 9, no. 2, pp. 177–188, 2018, doi: 10.15408/sijsi.v9i2.7647.

[4] A. Assagaf, A. Ibrahim, and C. Suranto, "Membangun Sistem Informasi Penjadwalan Dengan Metode Algoritma Genetika Pada Laboratorium Teknik Informatika Universitas Muhammadiyah Maluku Utar," J u r n a l I l m i a h I L K O M I N F O, vol. Vol.1, no. 2, pp. 95–105, 2018.

[5] T. H. Nio, "Analisis dan Perancangan Sistem Informasi Penjadwalan Laboratorium Menggunakan Algoritma Genetika," Pros. Semin. Nas. Mat. IX 2015, pp. 391–402, 2013.

[6] Muliadi, "Pemodelan Algoritma Genetika," Kumpul. J. Ilmu Komput., vol. 01, no. 01, pp. 67–78, 2014, [Online]. Available: http://klik.ulm.ac.id/index.php/klik/article/view/8.

[7] S. Sukirman, "Aplikasi Design of Experiment Dalam Proses Optimasi Moisture Content Pada Pengeringan Kayu Menggunakan Kiln Drying," Teknoin, vol. 20, no. 1, pp. 1–12, 2014, doi: 10.20885/teknoin.vol20.iss1.art3.

[8] D. Febiharsa, I. M. Sudana, and N. Hudallah, "UJI FUNGSIONALITAS (BLACKBOX TESTING) SISTEM INFORMASI LEMBAGA SERTIFIKASI PROFESI (SILSP) BATIK DENGAN APPPERFECT WEB TEST DAN UJI PENGGUNA bahwa Black-Box Testing merupakan pengujian perangkat lunak yang merupakan eksternal Sedangkan Blackbox Testin," JOINED J., vol. 1, pp. 117–126, 2019, doi: 10.31331/joined.v1i2.752.