

# Order Fulfillment pada Taksi Online dengan Mempertimbangkan Prioritas Penumpang Menggunakan Metode *Recency, Frequency* dan *Monetary*

Viona Angelica, Andreas Handoyo, Tanti Octavia

Program Studi Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-Mail: vionaang24@gmail.com, handoyo@petra.ac.id, tanti@petra.ac.id

## ABSTRAK

Seiring dengan berkembangnya teknologi di Indonesia, perusahaan taksi *online* menjadi salah satu bidang yang mulai dikembangkan. Untuk menjaga hubungan baik dengan penumpang dan meningkatkan pelayanan pada *loyal* penumpang, perusahaan taksi *online* tentu perlu membuat pelayanan khusus pada penumpang yang lebih *loyal*. Dalam penelitian ini, akan diterapkan faktor-faktor untuk meningkatkan pelayanan pada *loyal* penumpang seperti faktor *rating* pengemudi, jumlah *trip*, serta skor RFM pengemudi dan penumpang. Metode yang digunakan untuk memberikan *ranking* pada pengemudi dan penumpang menggunakan metode RFM dan *filtered* RFM. Metode yang digunakan untuk memasangkan pengemudi dan penumpang dalam proyek ini menggunakan metode Hungaria. Simulasi akan dijalankan dengan dua kali percobaan, yaitu secara *online* (menggunakan Google Maps API dan *database online*) dan secara *offline*. Penelitian ini menunjukkan bahwa dengan menambahkan faktor-faktor tambahan seperti skor RFM pengemudi dan penumpang, *rating* pengemudi, dan jumlah perjalanan pengemudi disertai dengan *limit* waktu penjemputan penumpang tidak memberikan perbedaan pada waktu *assign* time, waktu tunggu penumpang, dan waktu jemput penumpang tetapi lebih mendahulukan penumpang dan pengemudi dengan faktor prioritas tersebut di atas. Selain itu, kecepatan internet juga memiliki pengaruh yang cukup besar pada simulasi *order fulfillment* berbasis *website*.

**Kata Kunci:** Pemasangan Hungaria, prioritas RFM, simulasi pemasangan,

## ABSTRACT

Along with the development of technology in Indonesia, online taxi companies are one of the fields that are starting to be developed. Just like other companies, online taxi companies are looking for profits, to achieve it, they need to maintain good relations with their passengers. That can be achieved by improving service to loyal passengers. In this study, factors will be applied to improve service to loyal passengers and drivers such as rating, number of trips, driver's RFM score and passenger's RFM score. The method used to segment drivers and passengers is RFM prioritization and Filtered RFM prioritization. The method used to pair the driver and passengers is the Hungarian method. This study shows that by adding additional factors such as driver and passenger RFM scores, driver ratings, and the number of trip drivers accompanied by a passenger pick-up time limit, don't change the assign time, waiting time, and pickup time of passenger but can prioritize passengers and

drivers according to those factors. In addition, internet speed also has a huge influence on website-based order fulfillment simulations.

**Keywords:** Hungarian Assignment, RFM Prioritization, assignment simulation .

## 1. PENDAHULUAN

Perkembangan teknologi membuat segala bidang kehidupan menjadi lebih maju, termasuk di bidang transportasi. Hal ini ditunjukkan dengan munculnya beberapa perusahaan taksi *online* di Indonesia. Layaknya sebuah perusahaan, perusahaan taksi *online* pasti juga berusaha untuk memperoleh keuntungan. Memfokuskan pelayanan lebih pada penumpang *loyal* adalah cara terbaik untuk memperoleh keuntungan [10]. Penumpang *loyal* di sini berarti penumpang yang sering menggunakan layanan dan menghabiskan banyak biaya untuk itu. Sayangnya, belum banyak penelitian tentang sistem *order fulfillment* dengan mempertimbangkan penumpang *loyal* di dalamnya. Sistem *order fulfillment* yang mampu memberikan pelayanan lebih baik pada penumpang *loyal* dapat membuat mereka lebih senang menggunakan pelayanan perusahaan, sehingga dapat meningkatkan keuntungan bagi perusahaan. Namun masih sedikit sistem simulasi *order fulfillment* yang menggunakan faktor prioritas pelanggan dalam sistem mereka. Oleh karena itu, penelitian ini akan membuat simulasi *online* berbasis *website* untuk memasangkan pengemudi dan penumpang menggunakan metode Hungaria dengan mempertimbangkan beberapa faktor: durasi, jarak, *rating* pengemudi, jumlah *trip*, skor RFM penumpang, dan skor RFM pengemudi. Metode Hungaria digunakan karena merupakan metode yang paling banyak diterapkan untuk *assignment problem* karena waktu kalkulasinya cepat dan akurat [5]. Metode RFM digunakan untuk membuat segmentasi pelanggan karena merupakan metode yang dapat mensegmentasikan pelanggan dengan sederhana namun efektif [3, 7].

## 2. LANDASAN TEORI

### 2.1 HUNGARIAN PROGRAMMING

*Hungarian assignment* adalah salah satu metode yang sering digunakan dalam *assignment problem* [8]. Ide dasarnya adalah dengan memasangkan satu obyek ke obyek lain dan mencari skor yang paling optimal dari tiap pemasangan. Diasumsikan terdapat satu set 'n' jobs dimana  $J = \{J_1, J_2, \dots, J_n\}$  akan ditugaskan ke mesin 'm'  $M = \{M_1, M_2, \dots, M_m\}$  untuk eksekusi. Biaya eksekusi Cij,  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$ . Langkah perhitungannya adalah sebagai berikut:

- a. Input nilai m dan n.
- b. Mencari biaya minimum di setiap baris dan mengurangi masing-masing baris tersebut dengan nilai minimum yang didapatkan (Operasi ini akan membuat minimal salah satu angka dalam baris menjadi nol).
- c. Mencari biaya minimum di setiap kolom mengurangi masing-masing kolom tersebut dengan nilai minimum yang didapatkan (Operasi ini akan membuat minimal salah satu angka dalam kolom menjadi nol).
- d. Uji apakah tugas ideal telah dicapai. Hal ini dilakukan dengan memutuskan jumlah minimum garis yang diharapkan menutupi setiap nol. Jika jumlah garis sama dengan jumlah baris, maka masuk ke Langkah 7 jika tidak, ke Langkah 5.
- e. Jika jumlah garis tidak sama dengan jumlah baris, maka pilih biaya minimum yang tidak ditutupi garis yang telah ditentukan:
  - Kurangi angka yang tidak tertutup dengan garis dengan nilai minimum tersebut.
  - Tambahkan biaya minimum tersebut pada biaya yang terletak di setiap persimpangan garis.
- f. Untuk menetapkan pekerjaan, temukan baris yang hanya memiliki satu nol. Pilih nilai nol tersebut dan keluarkan nol yang tersisa yang sesuai dengan kolom masing-masing (pekerjaan yang sama tidak dapat ditetapkan ke lebih dari satu mesin tetapi mesin yang sama dapat melakukan lebih dari satu pekerjaan).
- g. Jika terjadi kesamaan skor, yaitu baris yang memiliki dua atau lebih nol, tentukan nilai yang memiliki biaya paling minimum pada masalah asli.
- h. Ulangi langkah 7 dan 8 hingga semua mesin sudah dipasangkan.

## 2.2 RFM SEGMENTATION

RFM pertama kali diperkenalkan oleh Stone dan Bob pada tahun 1989, merupakan metode yang efektif dan sederhana yang dapat diterapkan dalam segmentasi pasar [2]. Analisis RFM banyak digunakan untuk menentukan peringkat pelanggan dengan berdasarkan informasi riwayat pembelian pelanggan yang telah tercatat di masa lalu. Metode ini digunakan terutama dalam berbagai aplikasi yang melibatkan pelanggan dalam jumlah besar, seperti *online purchase*, *retailing*, dan lain-lain [3].

Menurut Christy et al., dalam metode ini, pelanggan akan dikelompokkan berdasarkan 3 dimensi, yaitu *Recency*(R), *Frequency*(F), *Monetary*(M). Nilai *recency* adalah jumlah hari/periode antara transaksi terakhir yang dilakukan pelanggan dan hari ini. Nilai *frequency* adalah jumlah transaksi yang dilakukan pelanggan. Sedangkan nilai *monetary* adalah total uang yang dihabiskan oleh pelanggan pada periode tertentu. Dalam pengelompokannya, setiap pelanggan diberi skor yang berbeda-beda untuk variabel *recency*, *frequency*, dan *monetary*. Skor yang diberikan menggunakan skala angka berdasarkan nilai kuintilnya. Misalnya dapat diaplikasikan angka dari 5 sampai 1, dimana kuintil teratas diberikan skor 5, dan kuintil terendah akan diberikan skor 1.

Kemudian semua pelanggan yang telah diberi skor untuk masing-masing *recency*, *frequency*, dan *monetary* akan dikelompokkan berdasarkan total skornya. Contohnya apabila seorang penumpang memiliki skor *recency* 4, skor *frequency* 5, dan skor *monetary* 4, maka skor RFMnya adalah 4 ditambah 5 ditambah 4 yaitu 13. Penumpang dengan skor RFM 15 adalah penumpang yang paling diprioritaskan di dalam skripsi ini.

Dalam penerapannya, apabila pengkategorian *recency* dipaksakan menjadi 5 kategori berdasarkan kuintilnya, maka ada kemungkinan *recency* yang bernilai sama akan dimasukkan ke dalam kategori berbeda. Hal ini tentu membuat perhitungan skor RFM menjadi berbeda dan tidak benar. Untuk menyelesaikan masalah ini John Wirth mengusulkan membuat perhitungan baru untuk nilai *recency* dengan hard coding. Di mana nilai *recency* dikategorikan secara manual dengan menggunakan interval 0 sampai 3 bulan, 4 sampai 6 bulan, 7 sampai 12 bulan, 13 sampai 24 bulan, dan lebih dari 25 bulan yang secara berturut-turut diberikan skor 5, 4, 3, 2, 1 [7].

Pada penelitian ini, akan dilakukan percobaan untuk memberikan filter pada nilai *frequency* dan *monetary* RFM untuk membuat data transaksi yang dibaca program menjadi lebih aktual. Perhitungan *frequency* dan *monetary* dibagi kembali menjadi 3 kategori, misalnya 120 hari, 60 hari, dan 30 hari. *Frequency* dan *monetary* kategori 1 akan dikalikan dengan 10%, kategori 2 dikalikan dengan 30%, dan kategori 3 akan dikalikan dengan 60%. *Frequency* dan *monetary* total didapatkan dari penjumlahan *frequency* dan *monetary* kategori 1 hingga 3. Di mana kemudian *frequency total* dan *monetary total* akan diolah menjadi skor, dengan membaginya ke dalam *quantile* masing-masing.

## 2.3 UJI TES T

Tes T adalah salah satu uji statistik untuk membandingkan dua buah rata-rata dari dua kelompok. Tujuannya adalah untuk mengetahui adanya perbedaan rata-rata antar kelompok setelah diberikan perlakuan tertentu. Tes T biasanya digunakan dalam kasus dimana subjek eksperimen dibagi menjadi dua kelompok independen, dengan satu kelompok diberi perlakuan A dan kelompok lain diberi perlakuan B. Peneliti dapat memperoleh dua jenis hasil untuk setiap kelompok (yaitu, sebelum perlakuan dan setelah perlakuan). pengobatan): preA dan postA, dan preB dan postB. Uji t independen dapat digunakan untuk perbandingan antar kelompok postA dan postB atau untuk perbandingan antar kelompok perubahan preA ke postA (postA-preA) dan perubahan preB ke postB (postB-preB) [6]. Pada skripsi ini, akan digunakan Uji T dua arah dengan varian yang tidak sama. Langkah-langkah Uji T yang digunakan yaitu:

1. Menentukan hipotesis 0 (H0) dan hipotesis 1 (H1)
 
$$H_0: \mu_1 - \mu_2 = d_0 \quad (1)$$

$$H_1: \mu_1 - \mu_2 \neq d_0 \quad (2)$$
2. Menentukan tingkat kepercayaan  
Tingkat kepercayaan yang sering digunakan dalam uji statistik adalah 95 persen, yaitu  $1 - \alpha = 0.95$ . Maka, tingkat signifikansinya adalah 5 persen, yaitu  $\alpha = 0.05$
3. Statistik uji untuk varian tidak sama
 
$$t = \frac{(\bar{x}_1 - \bar{x}_2) - d_0}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad (3)$$
4. Menentukan titik kritis  
Cara untuk menentukan titik kritis untuk uji dua arah adalah  $-T_{\alpha/2, v}$  dan  $T_{\alpha/2, v}$
5. Keputusan  
Keputusan untuk uji dua arah adalah tolak H0 apabila:  
 $t < -T_{\alpha/2, v}$  atau  $t > T_{\alpha/2, v}$

## 2.4 PENELITIAN SEBELUMNYA

Penelitian mengenai pemasangan pengemudi dan penumpang sudah pernah dibuat beberapa kali sebelumnya. Terdapat penelitian oleh Diamant & Baron yang membuat pemasangan dua buah antrian dengan mempertimbangkan kesabaran *customer*nya. Dalam penelitian ini diperoleh formula yang tepat untuk distribusi panjang antrian stasioner dan beberapa tindakan kinerja *steady-state* [4]. Selain itu, terdapat penelitian sebuah sistem *order fulfillment* dengan metode *combinatorial optimization* untuk memaksimalkan *driver acceptance rate* untuk tiap *order* tanpa mengurangi kepuasan pelanggan. Kelebihan penelitian ini adalah pengemudi dapat menerima atau menolak tugas yang telah dipasangkan mesin, sehingga untuk berjaga-jaga, mesin akan memasangkan pekerjaan pada lebih dari satu pengemudi yang sesuai. Sistem dibuat untuk memprediksi apakah pengemudi akan menerima atau menolak pesanan, dan memprediksi lokasi tujuan pengguna, guna menaikkan *global success rate*[11]. Juga, terdapat sistem *order fulfillment* dengan menggabungkan metode Hungarian dan Greedy. Penelitian ini berfokus pada penggabungan kedua metode tersebut untuk menghasilkan sistem *order fulfillment* yang lebih efektif. Simulasi dilakukan dengan statis di mana penumpang akan menunggu di stasiun yang sudah ada, dan faktor yang dihitung adalah jarak dan *waiting time* penumpangnya. [1]. Selain itu, terdapat sebuah sistem *driver assignment* pada suatu simulasi taksi *online* berbasis *web* menggunakan metode *Tabu Search*, *Branch and Bound* dan *Hungarian*. Hasilnya Hungarian Algorithm memiliki waktu kalkulasi yang paling kecil [5,9]. Seiring bertambahnya data, waktu kalkulasi metode Hungarian tetap stabil dengan kenaikan yang kecil, begitu juga dengan metode Tabu Search, sedangkan algoritma Branch and Bound mengalami peningkatan yang signifikan [5]. Sedangkan untuk segmentasi RFM, terdapat sebuah penelitian untuk membuat sebuah penelitian untuk memprioritaskan pelanggan dari data transaksi suatu perusahaan. Penelitian ini bertujuan untuk membandingkan metode K-Means, dan Fuzzy-C-Means untuk membuat segmentasi prioritas pelanggan yang telah dihitung menggunakan metode RFM sebelumnya. Hasil penelitian menunjukkan metode pengelompokan K-Means menghasilkan waktu yang lebih cepat daripada metode Fuzzy C-Means dan mengurangi jumlah iterasi datanya [3]. Dari penelitian-penelitian di atas, belum ada penelitian yang menerapkan metode segmentasi pelanggan dengan RFM untuk memasangkan pengemudi dan penumpang. Selain itu, penelitian serupa untuk membuat simulasi *online* berbasis *web* belum menggunakan faktor selain durasi dan jarak, sehingga dalam penelitian ini ditambahkan faktor lain untuk menjadi pertimbangan. Faktor-faktor yang digunakan dalam penelitian ini antara lain: jarak, waktu, *rating* pengemudi, skor RFM pengemudi, skor RFM penumpang, dan jumlah *trip* pengemudi.

## 3. DESAIN SISTEM

### 3.1 ALUR PROSES SISTEM

*Flowchart* pada Gambar 1 menggambarkan proses yang harus dilakukan saat melakukan pemasangan pengemudi dan penumpang. Proses yang dapat dilakukan berturut-turut antara lain: membaca data yang diperlukan dari *database*, menghitung faktor yang digunakan menjadi matriks biaya, melakukan pemasangan dengan metode Hungarian, dan menyimpan hasilnya ke dalam *database*.

Pada mulanya, *admin* akan mengisi *form* untuk membuat simulasi baru. Konfigurasi yang *diinput* berupa nama simulasi, tanggal mulai dan selesai, waktu mulai dan selesai, faktor-faktor yang akan digunakan dalam perhitungan, serta area yang digunakan

dalam perhitungan. Apabila simulasi yang dijalankan menggunakan RFM, maka *admin* perlu memasukkan pilihan untuk menjalankan *Filtered RFM* dan kategorinya, serta pengkategorian *recency* yang terdiri dari 5 kategori. Setelah mengklik tombol *submit*, akan muncul halaman simulasi *online*. Program simulasi untuk membangkitkan data penumpang dan pengemudi serta memasangkannya dapat dijalankan. Apabila telah ada pengemudi dan penumpang yang dibangkitkan atau dipasangkan, sistem akan membaca data yang diperlukan dari *database*. Data berupa data pengemudi dan penumpang tersebut kemudian digambarkan ke dalam *interface* Google Maps API.

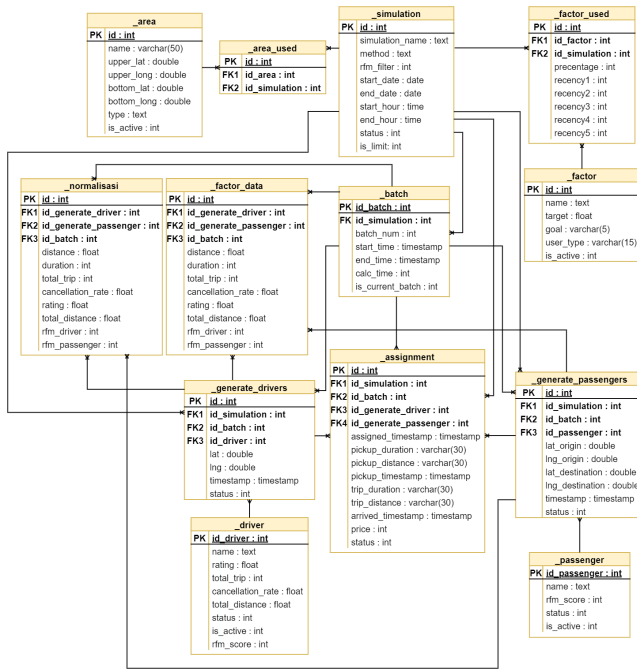


Gambar 1. Flowchart sistem

Apabila *admin* menghentikan simulasi yang dijalankan, program akan menampilkan hasil pemasangan yang telah disimpan dalam *database*. Halaman ini memuat berbagai informasi detail tentang simulasi yang baru saja dijalankan, seperti hasil *assignment* per *batch* dan waktunya, jumlah *batch*, jumlah *assignment*, jumlah *cancelled order*, *total assign time*, *waiting time*, *pick-up time*, dan *pick-up distance*. Data disajikan dalam bentuk grafik dan tabel untuk memudahkan proses analisa. Apabila simulasi yang dijalankan memiliki simulasi lain sebagai simulasi perbandingan, maka data simulasi perbandingan juga akan otomatis dimunculkan.

### 3.2 DATABASE

Struktur *database* yang digunakan dalam percobaan ini akan ditampilkan dalam bentuk *entity relationship diagram* (ERD). Model ERD yang digambarkan pada Gambar 2 menunjukkan terdapat 11 *entity* dalam simulasi ini, yaitu *\_area*, *\_area\_used*, *\_assignment*, *\_batch*, *\_driver*, *\_factor*, *\_factor\_data*, *\_generate\_drivers*, *\_generate\_passengers*, *\_passenger*, *\_simulation*, dan *\_transaksi*.



Gambar 2. Entity Relationship Diagram

## 4. PENGUJIAN SISTEM SIMULASI

### 4.1 UJI COBA 1

Tujuan percobaan ini untuk membandingkan hasil pemasangan metode Hungarian menggunakan faktor jarak, durasi, skor RFM pengemudi dan skor RFM penumpang dengan berbagai bobot persen, tanpa memberikan *limit* waktu penjemputan. Bobot yang digunakan antara lain:

- Simulasi 1: Jarak 50%, durasi 50%
- Simulasi 2: Jarak 10%, durasi 10%, skor RFM pengemudi 40%, skor RFM penumpang 40%
- Simulasi 3: Jarak 15%, durasi 15%, skor RFM pengemudi 35%, skor RFM penumpang 35%
- Simulasi 4: Jarak 20%, durasi 20%, skor RFM pengemudi 30%, skor RFM penumpang 30%

Pengategorian *recency* yang digunakan dalam uji coba ini adalah 120, 90, 60, 30, dan 14 hari. Keempat simulasi masing-masing dijalankan secara otomatis selama 15 menit. Simulasi yang dijalankan menggunakan database dan API *online*, sehingga data yang dihasilkan dalam uji coba ini dipengaruhi oleh jaringan internet. Detail hasil pengujian Uji Coba 1 dapat dilihat pada Tabel 1.

Tabel 1. Perbandingan Hasil Uji Coba 1

	Simulasi 1 Durasi	Simulasi 2 -RFM 80%	Simulasi 3 -RFM 70%	Simulasi 4 -RFM 60%
Jumlah Batch	15	15	15	15
Jumlah Assignment	34	34	34	34
Total Assign Time	236.189 s	261.422 s	281.004 s	268.425 s

Avg Waiting Time	11.547 menit	14.490 menit	13.030 menit	12.797 menit
Total Waiting Time	392.583 menit	492.650 menit	443.017 menit	435.100 menit
Min Pick Up Time	1.000 menit	1.000 menit	1.000 menit	1.000 menit
Max Pick Up Time	8.000 menit	9.000 menit	8.000 menit	9.000 menit
Avg Pick Up Time	10.000 menit	13.912 menit	12.176 menit	11.882 menit
Total Pick Up Time	340.000 menit	473.000 menit	414.000 menit	404.000 menit

Kesimpulan dari uji coba 1 adalah :

- Penambahan faktor RFM tanpa menggunakan *limit* waktu penjemputan membuat waktu *assign time* menjadi lebih lama dengan selisih waktu maksimal 45 detik
- Penambahan faktor RFM tanpa menggunakan *limit* waktu penjemputan membuat rata-rata waktu tunggu penumpang menjadi bertambah seiring dengan bertambahnya bobot faktor RFM yang digunakan pada simulasi. Seiring dengan bertambahnya bobot faktor RFM dengan total sebanyak 20%, menambah lama waktu tunggu penumpang rata-rata selama 80 detik.

### 4.2 UJI COBA 2

Uji coba ini dilakukan untuk membandingkan hasil pemasangan dengan metode Hungarian dengan berbagai faktor: jarak, durasi, *rating* pengemudi, skor RFM pengemudi, skor RFM penumpang, dan jumlah *trip* pengemudi. Bobot faktor masing-masing simulasi adalah:

- Simulasi 1: Durasi 50%, jarak 50%
- Simulasi 2: Durasi 10%, jarak 10%, skor RFM penumpang 40%, dan skor RFM pengemudi 40%
- Simulasi 3: Durasi 30%, jarak 30%, dan *rating* 40%
- Simulasi 4: Durasi 30%, jarak 30%, dan jumlah *trip* 40%
- Simulasi 5: Durasi 20%, jarak 20%, *rating* 30% dan jumlah *trip* 30%
- Simulasi 6: Durasi 10%, jarak 10%, *rating* 20%, skor RFM pengemudi 20%, jumlah *trip* 20%, dan skor RFM penumpang 20%

Pengategorian *recency* yang digunakan dalam simulasi kedua dan ketiga berturut-turut adalah 120, 90, 60, 30, dan 14 hari. Setiap pemasangan yang dilakukan dibatasi dengan jarak, yaitu jika jarak penjemputan lebih besar dari 4.300 meter, pemasangan akan dibatalkan agar penumpang tidak menunggu penjemputan terlalu lama. Simulasi ini dilakukan dua kali dengan menggunakan *database* dan API *online* (sehingga hasil yang diperoleh bergantung pada koneksi internet) dan dilakukan secara *offline* agar hasil tidak terpengaruh dengan internet. Seluruh simulasi dijalankan secara otomatis selama 15 menit, di mana jumlah penumpang dan *pengemudi* tiap batch yang dibangkitkan diacak dengan *range* 2-5 orang. Hasil uji coba dapat dilihat pada Tabel 2 dan Tabel 3.

Tabel 2. Perbandingan Hasil Uji Coba 2 online

	Durasi	RFM	Rating	Total Trip	All	All RFM

Jumlah Batch	15	15	15	15	15	15
Jumlah Assignment	48	47	48	48	48	46
Total Assign Time	420.794 s	482.588 s	411.455 s	423.864 s	421.151 s	487.962 s
Avg Waiting Time	7.072 menit	7.892 menit	7.456 menit	7.269 menit	7.47 menit	7.496 menit
Total Waiting Time	339.450 menit	370.917 menit	357.867 menit	348.900 menit	358.567 menit	344.833 menit
Min Pick Up Time	0.000 menit	1.133 menit	0 menit	0 menit	0 menit	0.883 menit
Max Pick Up Time	9.867 menit	9.133 menit	9.867 menit	9.867 menit	9.867 menit	9.833 menit
Avg Pick Up Time	5.495 menit	5.401 menit	5.807 menit	5.682 menit	5.808 menit	5.399 menit
Total Pick Up Time	263.767 menit	253.867 menit	278.750 menit	272.717 menit	278.767 menit	248.333 menit

**Tabel 3. Perbandingan Hasil Uji Coba 2 online**

	Durasi	RFM	Rating	Total Trip	All	All RFM
Jumlah Batch	15	15	15	15	15	15
Jumlah Assignment	48	47	48	48	48	46
Total Assign Time	115.779 s	123.297 s	119.549 s	118.584 s	121.868 s	128.786 s
Avg Waiting Time	6.204 menit	6.550 menit	6.547 menit	6.456 menit	6.514 menit	6.319 menit
Total Waiting Time	297.800 menit	307.867 menit	314.250 menit	309.867 menit	312.683 menit	290.667 menit
Min Pick Up Time	0.000 menit	1.133 menit	0 menit	0 menit	0 menit	0.883 menit
Max Pick Up Time	9.867 menit	9.133 menit	9.867 menit	9.867 menit	9.867 menit	9.833 menit
Avg Pick Up Time	5.495 menit	5.401 menit	5.807 menit	5.682 menit	5.808 menit	5.399 menit

Total Pick Up Time	263.767 menit	253.867 menit	278.750 menit	272.717 menit	278.767 menit	248.333 menit
--------------------	---------------	---------------	---------------	---------------	---------------	---------------

Kesimpulan uji coba 2 adalah:

- Menambahkan faktor RFM menambahkan waktu assign time program selama maksimal 73 detik, dan menambah waktu rata-rata tunggu penumpang selama maksimal 49 detik, mengurangi rata-rata waktu pickup selama 6 detik, dan mengurangi jumlah assignment simulasi sebanyak 1 pasang.
- Menambahkan faktor rating menambah waktu assign time program selama maksimal 4 detik, menambah waktu rata-rata tunggu penumpang selama maksimal 31 detik, menambah rata-rata waktu pickup selama 19 detik, dan tidak merubah jumlah assignment simulasi.
- Menambahkan faktor jumlah trip menambahkan waktu assign time program selama maksimal 12 detik, dan menambah waktu rata-rata tunggu penumpang selama maksimal 15 detik, menambah rata-rata waktu pickup selama 15 detik, dan tidak merubah jumlah assignment simulasi.
- Mengkombinasikan faktor jumlah trip dan rating pada program menambahkan waktu assign time program selama maksimal 10 detik, dan menambah waktu rata-rata tunggu penumpang selama maksimal 24 detik, menambah rata-rata waktu pickup selama 19 detik, dan tidak merubah jumlah assignment simulasi.
- Mengkombinasikan faktor jumlah trip, rating, dan RFM menambahkan waktu assign time program selama maksimal 69 detik, dan menambah waktu rata-rata tunggu penumpang selama maksimal 25 detik, mengurangi rata-rata waktu pickup selama 6 detik, dan mengurangi jumlah assignment simulasi sebanyak 2 pasang.

## 5. PENGUJIAN STATISTIK

Pengujian statistik dilakukan dengan menggunakan tes T pada sejumlah replikasi data pada masing-masing faktor. Nilai  $\alpha$  yang digunakan sebesar 5%. Masing-masing simulasi direplikasi sebanyak 10 kali di mana masing-masing simulasi berjumlah 15 batch. Data penumpang dan pengemudi yang digunakan dalam uji coba ini diacak dengan range 3-5 orang. Tabel 4 dan Tabel 5 menunjukkan hasil replikasi dari simulasi dengan faktor jarak dan durasi masing-masing sebesar 50%.

**Tabel 4. Replikasi Simulasi Durasi 1**

	Rep1	Rep2	Rep3	Rep4	Rep5
Average Assign time	28.121 detik	32.173 detik	29.236 detik	30.826 detik	32.559 detik
Average Waiting Time	6.299 menit	7.892 menit	5.786 menit	6.761 menit	5.958 menit
Average Pickup Time	4.972 menit	5.401 menit	4.414 menit	3.960 menit	4.752 menit

**Tabel 5. Replikasi Simulasi Durasi 2**

	Rep6	Rep7	Rep8	Rep9	Rep10
<i>Average Assign time</i>	41.007 detik	33.957 detik	37.428 detik	28.554 detik	28.556 detik
<i>Average Waiting Time</i>	6.446 menit	6.211 menit	5.490 menit	6.846 menit	6.673 menit
<i>Average Pickup Time</i>	4.928 menit	4.444 menit	4.912 menit	4.433 menit	4.167 menit

Data replikasi simulasi dengan faktor RFM dapat dilihat pada Tabel 6 dan Tabel 7. Seluruh simulasi dalam pengujian ini menggunakan faktor jarak 10%, durasi 10%, skor RFM pengemudi sebesar 40%, dan skor RFM penumpang sebesar 40%. Pengkategorian recency yang digunakan dalam seluruh replikasi berturut-turut adalah 120, 90, 60, 30, dan 14 hari.

**Tabel 6. Replikasi Simulasi RFM 1**

	Rep1	Rep2	Rep3	Rep4	Rep5
<i>Average Assign time</i>	28.121 detik	32.173 detik	29.236 detik	30.826 detik	32.559 detik
<i>Average Waiting Time</i>	6.299 menit	7.892 menit	5.786 menit	6.761 menit	5.958 menit
<i>Average Pickup Time</i>	4.972 menit	5.401 menit	4.414 menit	3.960 menit	4.752 menit

**Tabel 7. Replikasi Simulasi RFM 2**

	Rep6	Rep7	Rep8	Rep9	Rep10
<i>Average Assign time</i>	41.007 detik	33.957 detik	37.428 detik	28.554 detik	28.556 detik
<i>Average Waiting Time</i>	6.446 menit	6.211 menit	5.490 menit	6.846 menit	6.673 menit
<i>Average Pickup Time</i>	4.928 menit	4.444 menit	4.912 menit	4.433 menit	4.167 menit

Dilakukan uji Tes T untuk membandingkan Tabel 4-5 dengan Tabel 6-7 untuk mengetahui apakah ada perbedaan antara *assign time*, *waiting time*, dan *pickup time* penumpang antara simulasi dengan RFM dan tanpa RFM.

a. Perhitungan *Average Assign Time*

H0 : “Tidak ada perbedaan rata-rata *assign time* antara simulasi tanpa faktor RFM dan simulasi dengan faktor RFM”

Pada simulasi ini, x1 menandakan data replikasi dengan faktor durasi pada Tabel 4 dan Tabel 5, dan x2 menandakan data replikasi pada Tabel 6 dan Tabel 7. Nilai  $T_{\alpha/2,v}$  yang digunakan dalam pengujian ini sebesar 2,360.

$$n1 = 10$$

$$n2 = 10$$

$$\bar{x1} = 29.093$$

$$\bar{x2} = 32.242$$

$$S1 = 2.813526771$$

$$S2 = 4.242352179$$

$$t = \frac{(29.093 - 32.242)}{\sqrt{\frac{2.813526771^2}{10} + \frac{4.242352179^2}{10}}}$$

$$t = -0.3061864798$$

Karena nilai t adalah -0.3061864798 maka tidak memenuhi rumus, maka gagal tolak H0. Artinya, tidak ada perbedaan rata-rata *assign time* antara simulasi dengan faktor RFM dan tanpa faktor RFM.

b. Perhitungan *Average Waiting Time*

H0 : “Tidak ada perbedaan rata-rata *waiting time* antara simulasi tanpa faktor RFM dan simulasi dengan faktor RFM”

Pada simulasi ini, x1 menandakan data replikasi dengan faktor durasi pada Tabel 4 dan Tabel 5, dan x2 menandakan data replikasi pada Tabel 6 dan Tabel 7. Nilai  $T_{\alpha/2,v}$  yang digunakan dalam pengujian ini sebesar 2,360.

$$n1 = 10$$

$$n2 = 10$$

$$\bar{x1} = 6.4968$$

$$\bar{x2} = 6.436$$

$$S1 = 0.5754555297$$

$$S2 = 0.6720799886$$

$$t = \frac{(6.4968 - 6.436)}{\sqrt{\frac{0.5754555297^2}{10} + \frac{0.6720799886^2}{10}}}$$

$$t = 0.02961339174$$

Karena nilai t adalah -0.02961339174 maka tidak memenuhi rumus, maka gagal tolak H0. Artinya, tidak ada perbedaan rata-rata *waiting time* penumpang antara simulasi dengan faktor RFM dan tanpa faktor RFM.

c. Perhitungan *Average Pickup Time*

H0 : “Tidak ada perbedaan rata-rata *pickup time* antara simulasi tanpa faktor RFM dan simulasi dengan faktor RFM”

Pada simulasi ini, x1 menandakan data replikasi dengan faktor durasi pada Tabel 4 dan Tabel 5, dan x2 menandakan data replikasi pada Tabel 6 dan Tabel 7. Nilai  $T_{\alpha/2,v}$  yang digunakan dalam pengujian ini sebesar 2,360.

$$n1 = 10$$

$$n2 = 10$$

$$\bar{x1} = 4.6579$$

$$\bar{x2} = 4.8871$$

$$S1 = 0.4733141898$$

$$S2 = 0.58603611$$

$$t = \frac{(4.6579 - 4.8871)}{\sqrt{\frac{0.4733141898^2}{10} + \frac{0.58603611^2}{10}}}$$

$$t = 0.01330528906$$

Karena nilai  $t$  adalah 0.01330528906 maka tidak memenuhi rumus, maka gagal tolak  $H_0$ . Artinya, tidak ada perbedaan rata-rata *pickup time* penumpang antara simulasi dengan faktor RFM dan tanpa faktor RFM.

## 6. KESIMPULAN DAN SARAN

### 6.1 KESIMPULAN

- Kecepatan internet dan kondisi perangkat yang digunakan dalam menjalankan simulasi mempengaruhi lama waktu assign time simulasi, sehingga juga mempengaruhi lama waktu penumpang saat menunggu dipasangkan dengan pengemudi.
- Pemberian limit waktu penjemputan dapat mengurangi lama waktu tunggu dan lama *pick up* penumpang.
- Berdasarkan tes T penambahan faktor RFM pada simulasi tidak memberikan perbedaan pada rata-rata *assign time*, rata-rata waktu tunggu penumpang, dan rata-rata waktu *pick up* penumpang, tetapi dapat memasangkan penumpang dan pengemudi yang memiliki skor RFM lebih tinggi.
- Berdasarkan tes T penambahan faktor *Filtered* RFM pada simulasi tidak memberikan perbedaan pada rata-rata *assign time* penumpang, rata-rata waktu tunggu penumpang, dan rata-rata waktu *pickup* penumpang, tetapi dapat memasangkan penumpang dan pengemudi yang memiliki skor RFM lebih tinggi.
- Berdasarkan tes T penambahan faktor *rating* pada simulasi tidak memberikan perbedaan pada rata-rata *assign time* penumpang, rata-rata waktu tunggu penumpang, dan rata-rata waktu *pick up* penumpang, tetapi dapat memasangkan pengemudi yang memiliki *rating* lebih tinggi.
- Berdasarkan tes T penambahan faktor *total trip* pada simulasi tidak memberikan perbedaan pada rata-rata *assign time* penumpang, rata-rata waktu tunggu penumpang, dan rata-rata waktu *pick up* penumpang, tetapi dapat memasangkan pengemudi yang memiliki *total trip* lebih rendah.
- Berdasarkan tes T penambahan faktor *rating*, *total trip*, dan RFM pada simulasi tidak memberikan perbedaan pada rata-rata assign time penumpang, rata-rata waktu tunggu penumpang, dan rata-rata waktu pick up penumpang, tetapi dapat memasangkan penumpang dan pengemudi yang memiliki skor prioritas yang lebih tinggi.

## 7. REFERENSI

- [1] Babicheva, T., Burghout, W., Andreasson, I., & Faul, N. 2018. The matching problem of empty vehicle redistribution in autonomous taxi systems. *Procedia Computer Science*, 130, 119–125. DOI=10.1016/j.procs.2018.04.020.
- [2] Birant, D. 2011. Data mining using RFM analysis. In *Tech*, 91–108. URI= <https://www.intechopen.com/books/knowledge-oriented-applications-in-data-mining/data-mining-using-rfm-analysis>.
- [3] Christy, A. J., Umamakeswari, A., Priyatharsini, L., & Neyaa, A. 2018. RFM Ranking – An Effective Approach to Customer Segmentation. *Journal of King Saud University -*

Computer and Information Sciences. DOI=10.1016/j.jksuci.2018.09.004.

- [4] Diamant, A., & Baron, O. 2019. Double-sided matching queues: Priority and impatient customers. *Operations Research Letters*. DOI=10.1016/j.orl.2019.03.003.
- [5] Eryn (2020). Perbandingan metode Tabu Search dan metode Hungarian Algorithm untuk penentuan Driver Assignment pada simulasi taksi online. URI= <https://dewey.petra.ac.id/catalog/digital/detail?id=46347>.
- [6] Kim, T. K. 2015. T test as a parametric statistic. *Korean Journal of Anesthesiology*, 68(6), 540. DOI=10.4097/kjae.2015.68.6.540.
- [7] McCarty, J. A., & Hastak, M. 2007. Segmentation approaches in data-mining: A comparison of RFM, CHAID, and logistic regression. *Journal of Business Research*, 60(6), 656–662. DOI=10.1016/j.jbusres.2006.06.015.
- [8] Munkres, J. 1957. Algorithms for the Assignment and Transportation Problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1), 32–38. DOI=10.1137/0105003.
- [9] Rabbani, Q., Khan, A., & Quddoos, A. 2019. Modified Hungarian method for unbalanced assignment problems with multiple jobs. *Applied Mathematics and Computation*, 361, 493–498. DOI=10.1016/j.amc.2019.05.041
- [10] Zeithaml, V. A., Rust, R. T., & Lemon, K. N. 2001. The Customer Pyramid: Creating and Serving Profitable Customers. *California Management Review*, 43(4), 118–142. DOI=10.2307/41166104.
- [11] Zhang, L., Hu, T., Min, Y., Wu, G., Zhang, J., Feng, P., ... Ye, J. 2017. A Taxi Order Dispatch Model based On Combinatorial Optimization. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '17*. DOI=10.1145/3097983.