

Rekomendasi Penentuan Jadwal Ibadah Sehari-hari Dengan Algoritma Greedy

Muhammad Davis Adhipramana – 13522157

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13522157@std.stei.itb.ac.id

Abstract- Dalam kehidupan sehari-hari, khususnya bagi umat Islam, seringkali terdapat tantangan dalam menjaga keseimbangan antara ibadah dan aktivitas duniawi. Makalah ini bertujuan untuk menghasilkan sebuah algoritma yang dapat membantu dalam menyusun jadwal ibadah yang seimbang di tengah kesibukan harian, dengan menggunakan pendekatan algoritma Greedy. Algoritma ini dirancang untuk mengoptimalkan waktu dan memastikan bahwa setiap ibadah terjadwal dengan baik tanpa mengganggu aktivitas lainnya. Selain itu, algoritma ini akan mengalokasikan waktu berdasarkan Batasan seperti kewajiban dan waktu yang tepat. Hasil dari penelitian ini diharapkan dapat memberikan solusi praktis bagi umat Islam untuk memanfaatkan waktu mereka secara lebih efektif dan meningkatkan kualitas ibadah sehari-hari.

Keywords : Ibadah Sehari-hari, Penjadwalan Ibadah, Algoritma Greedy, Optimisasi Waktu, Keseimbangan Ibadah dan Aktivitas Duniawi

I. PENDAHULUAN

Dalam kehidupan sehari-hari, khususnya bagi umat Islam, menjaga keseimbangan antara ibadah dan aktivitas duniawi merupakan suatu tantangan yang sulit. Ibadah adalah salah satu kewajiban utama bagi umat Muslim, yang meliputi shalat, dzikir, membaca Al-Quran, dan berbagai bentuk ibadah lainnya. Namun, seiring bertambahnya usia dan semakin meningkatnya kesibukan. Seringkali kita lalai dan semakin meningkatnya kesibukan, seringkali kita lalai dan tidak melaksanakan ibadah dengan optimal. Padahal, sejak dini kita telah diajarkan bahwa kehidupan beragama dan duniawi haruslah seimbang, seperti yang tercantum dalam Al-Quran pada Q.S Al-Ahsr ayat 1-3 yang menyatakan bahwa manusia yang menyianyikan waktunya termasuk orang yang merugi. Selain itu, pada Q.S al-Hasyr ayat 19 juga dijelaskan akan kewajiban muslimin untuk selalu mengatur jadwal ibadahnya.

Pada kondisi seperti ini, diperlukan sebuah metode bagi umat muslim agar dapat mengatur waktu ibadahnya dengan lebih teratur dan seimbang ditengah kesibukan yang ada. Makalah ini memiliki tujuan untuk mengembangkan sebuah algoritma yang dapat membantu dalam penyusunan jadwal ibadah sehari-hari yang efisien dan efektif dengan menggunakan pendekatan Algoritma Greedy. Dengan algoritma ini, diharapkan dapat membantu mengoptimalkan

penggunaan waktu, memastikan ibadah terjadwal dengan baik dan tidak mengganggu aktivitas lainnya.

Melalui penelitian ini, diharapkan dapat memberikan solusi praktis bagi umat muslim yang taat agar dapat melakukan penjadwalan waktu dengan lebih efektif.

II. LANDASAN TEORI

A. Algoritma

Algoritma adalah sekumpulan aturan-aturan berhingga yang memberikan sederetan operasi-operasi untuk menyelesaikan masalah tertentu.

B. Algoritma Greedy

Algoritma Greedy adalah sebuah metode yang populer dan sederhana dalam memecahkan persoalan optimisasi. Algoritma ini bekerja dengan cara mengambil keputusan yang terbaik pada setiap langkah dengan harapan bahwa keputusan tersebut akan menghasilkan solusi optimal secara keseluruhan. Dalam konteks algoritma Greedy, terdapat dua macam persoalan optimisasi, yaitu

1. Maksimasi (Maximization) : bertujuan untuk memaksimalkan suatu nilai atau keuntungan tertentu
2. Minimasi (Minimization) : Bertujuan untuk meminimalkan suatu biaya atau kerugian tertentu

Pada makalah ini, algoritma Greedy yang diterapkan adalah maksimasi, yaitu dengan melakukan pemaksimalan penggunaan waktu dan ganjaran yang mungkin didapatkan dalam pelaksanaan ibadah dalam waktu tertentu.

Algoritma greedy sendiri memiliki beberapa karakteristik utama, yaitu

1. Membentuk Solusi Step by Step: Solusi yang dibentuk secara bertahap, dengan setiap langkah yang berusaha mendekati solusi optimal
2. Menghasilkan Solusi Optimal Lokal: Pada setiap langkah, algoritma Greedy membuat keputusan optimal secara lokal tanpa mempertimbangkan dampak jangka panjangnya.

Dalam algoritma greedy, terdapat beberapa elemen yang perlu diperhatikan diantaranya:

1. Himpunan Kandidat (C) : berisi semua kandidat yang akan dipilih pada tiap langkah.
2. Himpunan Solusi (S) : berisi kandidat yang sudah dipilih untuk membentuk solusi

3. Fungsi Solusi : Menentukan apakah himpunan kandidat yang dipilih sudah memberikan solusi yang valid.
4. Fungsi Seleksi : Memilih kandidat berdasarkan strategy greedy tertentu. Strategy ini bersifat heuristik dan bertujuan memilih kandidat terbaik pada tiap langkah.
5. Fungsi Kelayakan (*feasible*) : Memeriksa apakah kandidat yang dipilih dapat dimasukkan ke dalam himpunan solusi (layak atau tidak) .
6. Fungsi Obyektif : Bertujuan untuk memaksimalkan atau meminimalkan nilai tertentu.

Algoritma Greedy sendiri tidak akan selalu menghasilkan hasil yang optimal dan efisien. Namun demikian, karena heuristik pada kasus ini sepenuhnya akurat maka hasil optimal bukanlah sebuah patokan yang penting.

C. Ibadah

Kata “ibadah” berasal dari bahasa Arab yaitu “al-‘ibadah”. Kata ini merupakan bentuk mashdar dari kata kerja “abadaya” yang artinya “ketaatan”. Ibadah dalam Islam mencakup segala bentuk ketaatan dan penghambaan kepada Allah SWT. Ibadah memiliki makna tunduk kepada segala perintah dan larangan Allah SWT.

Terdapat beberapa jenis ibadah berdasarkan perbuatannya, yaitu :

1. Ibadah Qolbiyah : Ibadah yang mencakup aspek i’tiqad atau keyakinan adanya Allah SWT. Ibadah ini bersifat internal dan spiritual sehingga sulit untuk diukur secara kuantitatif
2. Ibadah Qauliyah : Ibadah yang dilakukan secara lisan, seperti membaca Al-Quran, dzikir, dan doa. Ibadah ini dapat diukur berdasarkan frekuensi dan durasinya
3. Ibadah Amaliyyah : Ibadah ini dilakukan melalui aktivitas fisik, seperti shalat, puasa, dan haji. Ibadah ini dapat diukur karena perbuatannya dapat dijadwal dengan mudah.
4. Ibadah Maaliyyah : Ibadah ini dilakukan dengan menyumbangkan harta, seperti zakat, sedekah, dan infaq. Ibadah ini bergantung pada kemampuan finansial individu sehingga sulit diukur dalam konteks penjadwalan waktu harian

Ibadah dalam Islam memiliki berbagai tingkatan berdasarkan kewajibannya, yang dapat memengaruhi prioritas dalam pelaksanaan:

1. Wajib : Ibadah yang harus dilakukan pada waktu yang sudah ditentukan, seperti shalat lima waktu. Ibadah wajib memiliki prioritas tertinggi dan tidak boleh ditinggalkan
2. Sunah : Ibadah yang dianjurkan tapi tidak wajib dilaksanakan, seperti dzikir dan shalat sunnah. Ibadah sunnah dilakukan untuk mendapat pahala tambahan dan mendekatkan diri pada Allah SWT.
3. Mubah : Aktivitas yang tidak menghasilkan pahala maupun dosa, seperti tidur atau aktivitas

harian lainnya. Meskipun tidak menghasilkan pahala, aktivitas mubah tetap penting untuk keseimbangan hidup.

Waktu memiliki peran penting dalam pelaksanaan ibadah dalam Islam. Ganjaran atau pahala dari Ibadah dapat bervariasi tergantung waktu atau hari pelaksanaannya. Berikut beberapa aspek penting yang perlu dipertimbangkan :

1. Waktu spesifik ibadah : Terdapat beberapa ibadah yang hanya dapat dilakukan pada waktu tertentu. Ibadah tersebut diantaranya: Shalat 5 waktu, Sholat sunnah dhuha, sholat sunnah fajar, Puasa senin kamis, dan sebagainya.
2. Ibadah pada hari tertentu : Pada hari seperti hari jumat, idul fitri, idul adha, terdapat ibadah yang bersifat wajib dan tambahan yang hanya ada pada hari tersebut.
3. Ibadah pada bulan Ramadhan : pada bulan penuh keberkahan ini, tingkat ganjaran yang diberikan akan berbeda dengan ganjaran pada hari biasanya.

Memahami ibadah ini sangat diperlukan dalam penerapan algoritma greedy nantinya karena seluruh elemen greedy yang ada akan didasarkan pada kalkulasi dan batasan yang ada pada ibadah nyata.

D. Integer Knapsack Problem

Integer Knapsack Problem adalah sebuah permasalahan optimasi yang bertujuan memaksimalkan total keuntungan yang bisa didapatkan dengan batasan kapasitas tertentu. Dalam knapsack, setiap item memiliki berat (weight) dan nilai (profit) dan tujuan utamanya adalah memilih sejumlah item sehingga total berat tidak melebihi kapasitas dengan memaksimalkan total keuntungan yang didapat.

III. IMPLEMENTASI

A. Penentuan Elemen Greedy

Untuk menjadwalkan ibadah dengan pendekatan Greedy, terdapat beberapa batasan yang perlu diperhatikan, yaitu waktu yang digunakan untuk beribadah dan batasan seperti kewajiban dan kesunahan dari ibadah itu sendiri.

Sebelumnya, interval waktu yang akan digunakan pada scheduling ini adalah :

- Shubuh : 04:00 – 06:00
- Pagi : 06:00 – 12:00
- Siang : 12:00 – 15:00
- Sore : 15:00 – 18:00
- Maghrib : 18:00 – 19:00
- Malam : 20:00 – 04:00

Elemen greedy pada penerapan scheduling ini dapat dibagi menjadi berikut :

- Himpunan Kandidat (C) : Himpunan kandidat untuk masing-masing waktu memiliki Himpunan kandidatnya tersendiri. Misalnya pada waktu shubuh maka kandidat dari elemen tersebut adalah {sholat

shubuh, membaca Al-Quran, murojaah Al-Quran, Dzikir}.

- Himpunan Solusi (S) : Seluruh kandidat yang memenuhi untuk durasi tertentu. Contohnya: pada waktu shubuh, durasi kosongnya adalah 1 jam maka waktu tersebut cukup untuk sholat shubuh dan membaca alquran
- Fungsi Solusi : Solusi dengan density paling kuat density tersebut didapatkan dari π/w_i dimana π adalah profit atau keuntungan dari ibadah yang dikerjakan dan w_i adalah durasi dari ibadah tersebut.
- Fungsi Kelayakan (Feasible) : solusi yang didapat memenuhi perhitungan berikut :
 $\sum w_i \leq k$, dimana $\sum w_i$ adalah akumulasi durasi dari ibadah yang dipilih dan k adalah durasi waktu kosong yang ada.
- Fungsi Obyektif : Pada penerapan scheduling ini, ibadah dengan type wajib akan diutamakan terlebih dahulu lalu akan disorting dan didapatkan kombinasi maximal ibadah yang dapat dikerjakan.

Dengan elemen-elemen tersebut diharapkan dapat menghasilkan Algoritma greedy yang cukup mangkus dalam menghasilkan jadwal yang dibutuhkan.

B. Constraint

Pada penerapan scheduling ibadah ini, terdapat beberapa constraint yang harus dilakukan sebelum menggunakan algoritma greedy yang ada. Constraint tersebut diantaranya:

- Ibadah dengan tipe wajib harus dilaksanakan paling awal
- Terdapat beberapa ibadah yang hanya dapat dilakukan sekali sehari seperti sholat fardhu /wajib, sholat dhuha, dan lainnya
- Sholat memiliki waktunya tersendiri
- Waktu dimulai pukul 04.00 – 03.00 hal ini untuk mempermudah kalkulasi jadwal nantinya

Dengan constraint ini, diharapkan sudah dapat merepresentasikan jadwal sesungguhnya

C. Datatype Ibadah

Data perhitungan ibadah pada scheduling ini disimpan pada sebuah json file. Hal ini dilakukan untuk memudahkan perubahan data ibadah yang ada jika diperlukan

```
1 "ibadah": [  
2   {  
3     "name": "Shalat Subuh",  
4     "weight": 30,  
5     "profit": 100,  
6     "time_of_day": "shubuh",  
7     "days": ["everyday"],  
8     "type": "wajib",  
9     "allowed": "once"  
10  },
```

Gambar 1: Bentuk data yang digunakan pada perhitungan

Terdapat beberapa variable yang terdapat pada tiap potongan json tersebut :

- name : nama yang digunakan saat ditampilkan pada scheduling
- weight : bobot ibadah, pada konteks ini, bobot merupakan representasi dari durasi
- profit : perkiraan pahala yang didapat saat melaksanakan ibadah tersebut. Perkiraan ini masih ambigu karena tidak dapat diketahui pastinya
- time_of_day : waktu khusus ibadah dapat dilakukan
- days : hari apa saja sebuah ibadah dapat dilakukan, ini berkaitan dengan hari jumat
- type : tingkat kewajiban sebuah ibadah (ada wajib dan sunah)
- allowed : berapa kali suatu ibadah dapat dilakukan dalam sehari (ada once atau sekali dan multiple atau dapat lebih dari sekali)

Mengetahui data tersebut penting karena akan banyak digunakan pada implementasi scheduling nantinya. Diharapkan, data tersebut cukup merepresentasikan suatu ibadah jika akan dikomputasikan.

D. Implementasi Knapsack pada Scheduling

Greedy yang diterapkan pada scheduling ini memiliki mekanisme knapsack. Greedy ini akan menghasilkan list of ibadah yang akan dilaksanakan pada waktu tertentu.

```
1 def calculate_knapsack(self, ibadah_list, duration):  
2     selected_ibadah = []  
3     remaining_duration = duration  
4  
5  
6     # Day Constraint Implementation  
7     if self.day == 'JUMAT': # Check if it is a Jumat  
8         wajib_activities = [ibadah for ibadah in ibadah_list if ibadah['type'] == 'wajib' and ibadah['name'] != 'Sholat Dhuha']  
9         sunah_activities = [ibadah for ibadah in ibadah_list if ibadah['type'] == 'sunah']  
10    else: # If it is not  
11        wajib_activities = [ibadah for ibadah in ibadah_list if ibadah['type'] == 'wajib' and ibadah['name'] != 'Shalat Jumat']  
12        sunah_activities = [ibadah for ibadah in ibadah_list if ibadah['type'] == 'sunah']  
13  
14    # Sort the density of the wajib and sunah activities  
15    wajib_activities.sort(key=lambda x: x['profit']/x['weight'], reverse=True)  
16    sunah_activities.sort(key=lambda x: x['profit']/x['weight'], reverse=True)  
17  
18    # Clear wajib activities first  
19    for ibadah in wajib_activities:  
20  
21        weight = ibadah['weight'] / self.interval_duration  
22        if weight <= remaining_duration:  
23            selected_ibadah.append(ibadah)  
24            # Multiple pick constraint (# Remove from list if only allowed once)  
25            if (ibadah['allowed'] == 'once'):  
26                self.removeOnceAllowedIbadah(ibadah)  
27            remaining_duration -= weight  
28  
29    # Then Sunah followed respectively  
30    for ibadah in sunah_activities:  
31  
32        weight = ibadah['weight'] / self.interval_duration  
33  
34        if weight <= remaining_duration:  
35            selected_ibadah.append(ibadah)  
36            # Multiple pick constraint (# Remove from list if only allowed once)  
37            if (ibadah['allowed'] == 'once'):  
38                self.removeOnceAllowedIbadah(ibadah)  
39            remaining_duration -= weight  
40    return selected_ibadah
```

Gambar 2 : Pemodelan Knapsack dalam menyelesaikan permasalahan Scheduling

Pada awalnya, function akan meminta parameter berupa list of ibadah dan duration. List of ibadah ini adalah list dari ibadah yang dilaksanakan pada waktu tertentu. Misalnya pada pagi hari atau Morning, ibadah yang dilakukan adalah ibadah yang hanya dapat dilakukan pada pagi hari atau ibadah yang dapat dilakukan pada waktu kapan saja. Parameter kedua adalah durasi. Pada konteks permasalahan knapsack, durasi ini adalah k atau batas yang tidak boleh dilewati atau dilebihi dari weight yang ada.

Setelah itu akan dilakukan pengecekan berdasarkan hari yang ada. Pada konteks ibadah islam, pada hari jumat, sholat dhuhur akan digantikan dengan sholat jumat dan perlu dilakukan handling untuk case tersebut.

Setelah constraint akan hari, kita perlu mengetahui bahwa ibadah yang dilakukan akan dibedakan menjadi ibadah wajib dan ibadah sunah. Hal ini mengingat batasan bahwa ibadah wajib harus dilaksanakan terlebih dahulu daripada ibadah sunah.

Penerapan algoritma greedy dilakukan berdasarkan density dari tiap-tiap ibadah yang ada.

$$\text{Density} = \text{Profit} / \text{Weight}$$

Pada akhirnya akan dihasilkan sebuah list yang berisikan density dari masing-masing ibadah dan diurutkan terurut mengecil.

Untuk masing-masing list of density dari tiap type yang ada, akan dilakukan pengecekan apakah weightnya ketika ditambahkan akan melebihi duration yang ada. Selain itu, untuk tiap ibadah jika ibadah tersebut hanya boleh sekali sehari, maka ibadah tersebut akan di remove dari list of ibadah yang bersifat global. Untuk ibadah yang tidak melanggar constraint yang diberikan, akan ditambahkan pada selected_ibadah dan akan di return sebagai ibadah yang dipilih.

E. Implementasi Scheduling

Knapsack yang dikalkulasi hanya merupakan implementasi dari salah satu duration yang didapatkan. Pada dasarnya fungsi perhitungan knapsack tersebut digunakan pada bagian lain dalam program. Bagian ini menggunakan konsep design pattern observer dalam pembuatannya

```
1 def calculate_scheduling(self):
2
3     scheduled_activities = {
4         'SHUBUH' : [],
5         'MORNING' : [],
6         'AFTERNOON' : [],
7         'EVENING' : [],
8         'MAGHRIB' : [],
9         'NIGHT' : []
10    }
11
12    for freetime in self.knapsack_list:
13        duration = freetime[0]
14        types = freetime[1]
15        selected_activities = self.calculate_one_type_of_knapsack(duration, types)
16        if(selected_activities):
17            for activity in selected_activities:
18                scheduled_activities[types].append(activity)
19
20    return scheduled_activities
```

Gambar 3 : Program untuk process tiap waktu kosong

Pada awal program untuk masing-masing ibadah yang akan dicatat, akan dibuat list yang dimasukkan dalam dictionary. Dictionary ini akan digunakan nantinya saat menampilkan jadwal ibadah yang dibuat pada tampilan program.

Untuk masing-masing waktu kosong yang ada, akan didapatkan list of activity dari perhitungan knapsack dimana tiap activity tersebut akan dimasukan kedalam dictionary sesuai dengan tipe ibadah tersebut.

```
1 def schedules(self, activities : tuple[str, str], day, interval_duration: float = 60):
2     self.reset_ibadah_schedule() # Reset the Ibadah of the Day
3     self.day = day # Day when the schedules button is clicked
4     self.interval_duration = interval_duration # Duration of the User
5     start_time = None
6
7     for counter, act in enumerate(activities):
8         if(act[1] == "" and not start_time):
9             start_time = act[0]
10        if(start_time):
11            isSame = (self.get_type_of_time(start_time) == self.get_type_of_time(act[0]))
12            if(not isSame or act[1] != ""):
13                end_time = act[0]
14                self.add_ibadah(start_time, end_time)
15                start_time = None
16            elif(counter == 23 and act[1] == ""):
17                self.add_ibadah(start_time, act[0])
18    return self.calculate_scheduling()
```

Gambar 4 : Program utama scheduling

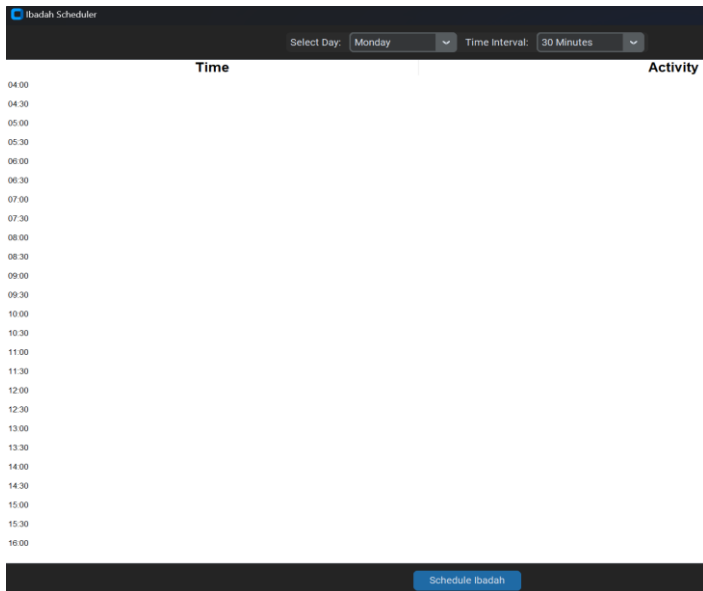
Program ini merupakan program utama yang memanggil seluruh deretan instruksi yang lainnya seperti calculate_scheduling. Program ini meminta dua parameter berupa activities yang berisikan seluruh aktivitas user pada hari itu, interval_duration berupa interval durasi yang dipilih oleh user pada program dapat berupa 1 jam atau 60, 2 jam atau 120, dan 0.5 jam atau 30. Interval_duration ini akan digunakan nantinya untuk mengkalkulasi weight dari suatu ibadah.

Process kalkulasi waktu kosong dilakukan dengan pada awalnya menyimpan waktu ketika ditemukan aktivitas kosong pada list yaitu ketika activity == "". Pada implementasi ini, waktu akhir akan di hitung jika ditemukan activity yang tidak kosong atau sudah melebihi suatu jentang waktu. Misalnya jika waktu awal adalah saat pagi dan ketika activity masih kosong hingga sudah siang.

Program akan memanggil instruksi add_ibadah untuk menambahkan pada self.knapsack_list berupa tuple yang berisikan durasi dan tipe waktu saat itu. Pada akhirnya program akan mereturn self.calculate_scheduling() untuk diprocess pada bagian UI nantinya.

F. Pembuatan perangkat lunak untuk Scheduling

Pembuatan GUI atau Graphical User Interface dalam memberikan gambaran untuk user, tentunya akan sangat membantu user dalam melakukan Scheduling



Gambar 6 : User Interface

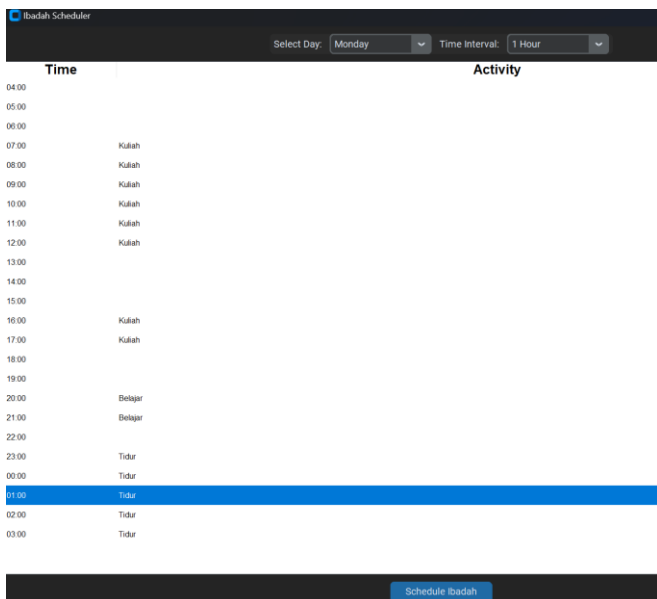
Pada aplikasi tersebut terdapat beberapa fitur yang dapat digunakan yaitu user dapat memilih hari yang di-inginkan serta time interval yang dapat dipilih user. User dapat menuliskan seluruh activity nya dalam sehari sesuai dengan time yang ada.

Setelah user mengisi keseluruhan, user dapat menekan tombol Schedule Ibadah untuk mulai melakukan mengisi bagian yang kosong dengan ibadah yang direkomendasikan secara otomatis.

IV. TESTING DAN ANALISIS

1. Pengujian Diluar Hari Jumat

Input:

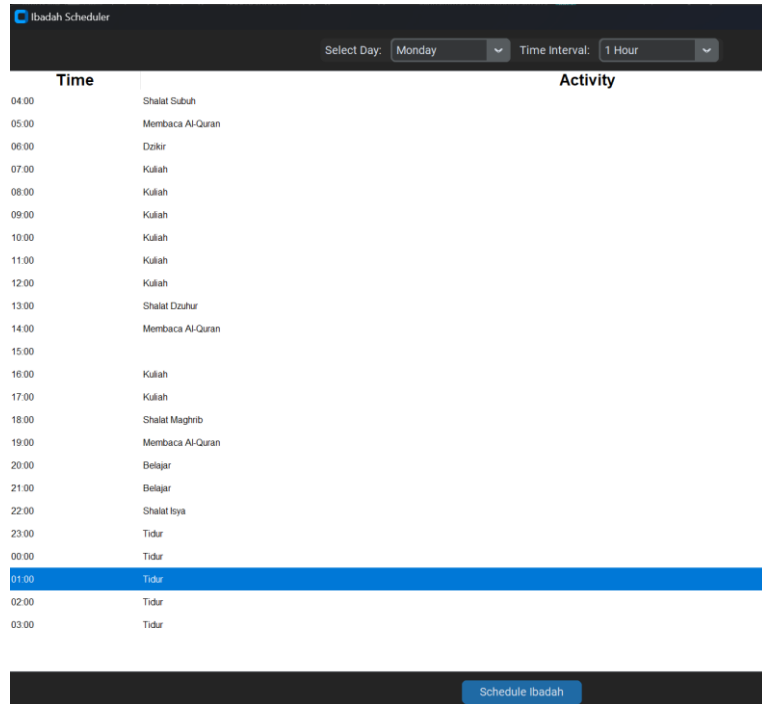


Gambar 7 : Test dengan Input Hari Senin

Spesifikasi :

- Hari : Senin
- Interval Waktu : 1 Jam
- Keadaan : Kuliah pukul 7 – 12 dan kegiatan kecil lainnya yang tersebar

Output :



Gambar 8 : Output Test 1

Analisis:

Pada pengujian pertama ini, algoritma berhasil melakukan scheduling pada mayoritas bagian kosong pada jadwal yang ada. Namun, terdapat 1 kesalahan atau error yang dihasilkan pada pukul 15:00 dimana seharusnya pada pukul tersebut masih dapat diisi dengan suatu aktivitas.

2. Pengujian pada Hari Jumat

Time	Activity
04:00	Tidur
05:00	Tidur
06:00	Tidur
07:00	Tidur
08:00	
09:00	
10:00	
11:00	
12:00	
13:00	Kuliah
14:00	Kuliah
15:00	Kuliah
16:00	Kuliah
17:00	
18:00	
19:00	
20:00	Nubes
21:00	Nubes
22:00	Nubes
23:00	Nubes
00:00	Nubes
01:00	Tidur
02:00	Tidur
03:00	Tidur

Gambar 9 : Test dengan Input hari Jumat

Spesifikasi :

- Hari : Jumat
- Interval Waktu : 1 Jam
- Keadaan: Padat dengan jumlah waktu kosong hanya pada pukul 8-12 dan 17-19.

Time	Activity
04:00	Tidur
05:00	Tidur
06:00	Tidur
07:00	Tidur
08:00	Membaca Al-Quran
09:00	Dzikir
10:00	Shalat Dhuha
11:00	Muroajaah Al-Quran
12:00	Shalat Jumat
13:00	Tidur
14:00	Tidur
15:00	Tidur
16:00	Tidur
17:00	Shalat Ashar
18:00	Shalat Maghrib
19:00	Membaca Al-Quran
20:00	Nubes
21:00	Nubes
22:00	Nubes
23:00	Nubes
00:00	Nubes
01:00	Tidur
02:00	Tidur
03:00	Tidur

Gambar 10: Output Test 2

Analisis:

Pada pengujian kedua ini, algoritma berhasil memenuhi keseluruhan jadwal dengan baik dan masih dapat melakukan handling untuk hari Jumat.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Algoritma greedy dapat dengan baik menghasilkan sebuah penjadwalan yang baik dimana algoritma dapat dengan mudah digunakan dalam batasan atau constraint yang cukup banyak. Hal ini membuat jadwal ibadah yang sesuai dapat dibuat dengan cepat.

B. Saran

- Perhitungan profit serta weight yang digunakan, dapat ditingkatkan lagi sesuai dengan keadaan ibadah sebenarnya
- Masih terdapat beberapa constraint yang dapat ditambahkan seperti aturan bahwa sholat tahajud dilaksanakan setelah tidur atau sholat khobdiyah dan badiyah hanya dapat dilaksanakan sebelum dan sesuai sholat.
- GUI yang dibuat dapat ditingkatkan guna pembacaan yang lebih baik.

UCAPAN TERIMA KASIH

Pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah mendukung dan berkontribusi dalam penyelesaian makalah ini. Ucapan terima kasih disampaikan kepada:

1. Tuhan Yang Maha Esa, atas rahmat dan hidayah-Nya sehingga makalah ini dapat diselesaikan dengan baik.
2. Ir. Rila Mandala, M.Eng., Ph.D. dan Monterico Adrian, S.T., M.T., selaku dosen mata kuliah IF2211 Strategi Algoritma yang telah memberikan ilmu dan materi dasar yang sangat membantu dalam penulisan makalah ini.

Ucapan terima kasih yang tulus juga disampaikan kepada semua pihak yang telah berkontribusi dalam perjalanan penulisan makalah ini.

REFERENSI

- [1] Wardani, "Manajemen Waktu Menurut Islam," UIN Antasari, [Online]. Available: <https://www.uin-antasari.ac.id/manajemen-waktu-menurut-islam/>
- [2] "Ini Dia Pengertian Algoritma Beserta Karakteristiknya," Universitas Bakrie. <https://bakrie.ac.id/articles/628-ini-dia-pengertian-algoritma-beserta-karakteristiknya.html>
- [3] R. Munir, "Algoritma Greedy (2021) Bag 1," Institut Teknologi Bandung, [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf)

- [4] "Pengertian dan Jenis Ibadah dalam Islam," Gramedia, :
https://www.gramedia.com/literasi/pengertian-dan-jenis-ibadah-dalam-islam/#google_vignette
- [5] "Pengertian Ibadah Agama Islam, Macam-macam, dan Syarat Diterimanya," Liputan6,
<https://www.liputan6.com/hot/read/4721230/pengertian-ibadah-agama-islam-macam-macam-dan-syarat-diterimanya>
- [6] "Apa Itu Manajemen Waktu dalam Islam? Ini Penjelasan," Kumparan, <https://kumparan.com/berita-hari-ini/apa-itu-manajemen-waktu-dalam-islam-ini-penjelasan-21ZmCH3Xrml/full>
- [7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms.

Bandung, 12 Juni 2024



Muhammad Davis Adhipramana, 13522157

Video Youtube : <https://youtu.be/bk1G7Urde3M>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.