

LAPORAN TUGAS BESAR

IF2110/Algoritma dan Struktur Data

MOBILITA


Dipersiapkan oleh:

Kelompok 3

Suryanto	13520059
Ziyad Dhia Rafi	13520064
Farnas Rozaan Iraquee	13520067
Lyora Felicya	13520073
Rava Naufal Attar	13520077

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2110-TB-02-03</i>		<i>18</i>
		<i>Revisi</i>	<i>1</i>	<i>21/10/2021</i>

Daftar Isi

1 Ringkasan.....	3
2 Penjelasan Tambahan Spesifikasi Tugas	3
2.1 Daftar Pesanan.....	3
2.2 Save dan Load	3
3 Struktur Data (ADT)	4
3.1 ADT Point	4
3.2 ADT List Statis.....	4
3.3 ADT List Dinamis	4
3.4 ADT Matriks	4
3.5 ADT Mesin Karakter dan Mesin Kata	4
3.6 ADT Queue	4
3.7 ADT Stack.....	5
3.8 ADT Linked List	5
4 Program Utama	6
5 Data Test	6
5.1 Data Test 1	6
5.2 Data Test 2	11
6 Test Script	13
7 Pembagian Kerja dalam Kelompok	15
8 Lampiran	15
8.1 Deskripsi Tugas Besar.....	15
8.2 Notulen Rapat.....	16
8.3 Log Activity Anggota Kelompok.....	18

1 Ringkasan

Pandemi COVID-19 sangat berdampak pada sektor usaha dengan berbagai kelas, salah satunya usaha orang tua Mobita yang ikut mengalami penurunan pendapatan. Dengan peringanan tingkat PPKM, Mobita ingin membantu orang tuanya dengan menjadi kurir. Kami diminta untuk membuat aplikasi yang dapat membantu Mobita melacak pesanan, navigasi peta, mengambil dan menurunkan barang.

Kami membuat sebuah permainan berbasis CLI (*command-line interface*) tentang pengantaran barang. Permainan yang dibuat dalam bahasa C dengan menggunakan struktur data yang telah kami pelajari di kelas IF2110.

Pada laporan ini berisi terkait, penjelasan spesifikasi tugas, struktur data yang digunakan pada program, alur program utama, algoritma yang menarik bagi kontributor, data test program, test script program, pembagian kerja dalam kelompok, dan beberapa lampiran.

Pada tugas besar kali ini, digunakan berbagai struktur data. Struktur data tersebut sangat berkaitan dan harus digunakan secara tepat karena memiliki kegunaan yang berbeda-beda. Melalui tugas besar ini kami dapat belajar bagaimana caranya mengimplementasikan ADT-ADT yang telah dipelajari di kelas untuk suatu program permainan sederhana. Selain itu, tugas ini juga mendorong kami untuk bekerja sama dan saling membantu dalam proses pengerjaan tugas.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Daftar Pesanan

Daftar pesanan berkaitan dengan *To Do List*, *In Progress List*, dan Tas. Tidak semua daftar pesanan dapat masuk ke *To Do List*, bergantung pada waktu pesanan masuk, jika waktu pesanan masuk lebih besar atau sama dengan waktu sekarang, maka pesanan dapat masuk ke *To Do List*. Berikutnya apabila Mobita *Pick Up* suatu barang maka pesanan tersebut dihapus dari *To Do List* dan masuk ke *In Progress List* dan Tas Mobita. Sedangkan, bila Mobita *Drop Off* suatu barang, maka pesanan tersebut dihapus dari *In Progress List* dan Tas Mobita.

2.2 Save dan Load

Fitur ini memungkinkan player untuk melakukan penyimpanan state-state yang ada ke dalam file konfigurasi agar dapat dilanjutkan lagi di waktu lain. State-state yang disimpan di antaranya adalah jumlah pesanan yang telah diantar, posisi Mobita, uang, waktu, kondisi *ability speedboost*, informasi *heavy items* dalam tas, kapasitas beserta isi dari tas, data progress pesanan yang diantarkan, daftar todo, inventory dari Mobita dan informasi dari file semula. Semua informasi ini kemudian dituliskan ke dalam suatu file berformat txt, baik file konfigurasi sebelumnya, atau pada file baru.

3 Struktur Data (ADT)

3.1 ADT Point

ADT Point terdiri dari 3 buah variabel sebagai penanda posisi bangunan pada peta. Variabel pertama bertipe character untuk menyimpan nama bangunan, dan dua variabel lainnya bertipe integer untuk menyimpan koordinat, yaitu Absis dan Ordinat dari bangunan tersebut.

3.2 ADT List Statis

ADT List Statis digunakan untuk *List Inventory* dan *List Gadget*. Hal ini dilakukan karena jumlah elemen dari Inventory sudah diketahui jumlahnya, dan operasi yang dilakukan juga cukup sederhana yaitu menghapus dan memasukkan elemen dari List Gadget ke List Inventory tanpa memerlukan keterurutan elemennya. List Gadget menyimpan DataGadget berupa id dan harga dari Gadget, sedangkan List Inventory menyimpan id Gadget setelah sebuah Gadget dibeli.

Pada program kami, ADT List Statis diimplementasikan pada ListGadget.h dan ListGadget.c, merupakan sebuah list statis yang menyimpan elemen bertipe DataGadget yang terdiri dari idname dan price, yaitu id dan harga dari gadget yang dapat dibeli. Kemudian diimplementasikan juga pada ListInventory.h dan ListInventory.c, yang merupakan sebuah list statis yang menyimpan data Gadget bertipe integer, yaitu idname dari gadget yang telah dibeli.

3.3 ADT List Dinamis

ADT List Dinamis digunakan untuk menyimpan daftar bangunan yang tersedia pada peta. Alasan kami memilih ADT List Dinamis untuk menyimpan daftar bangunan adalah karena jumlah elemen yang akan disimpan di dalam list ini tidak menentu, dan akan berubah-ubah sesuai dengan titik/bangunan yang dapat dicapai pemain, dan menyesuaikan dengan *input* dari pengguna.

Pada program, ADT List Dinamis diimplementasikan pada listpoint.h dan listpoint.c. List ini terdiri dari 3 variabel yaitu capacity, jumlah elemen efektif nEff, serta buffer sebagai memori tempat penyimpanan, dalam hal ini yaitu Point (titik dari bangunan-bangunan).

3.4 ADT Matriks

ADT Matriks digunakan untuk representasi tampilan peta permainan, serta merepresentasikan hubungan antar bangunan menggunakan Adjacency Matrix.

3.5 ADT Mesin Karakter dan Mesin Kata

ADT Mesin Karakter dan Mesin Kata merupakan ADT yang digunakan untuk membaca masukan berupa teks. Pada program kami, kedua ADT tersebut digunakan untuk membaca file eksternal berupa konfigurasi untuk memulai permainan, serta untuk membaca input pemain baik berupa karakter maupun angka.

3.6 ADT Queue

ADT Queue merupakan sebuah struktur data dimana penghapusan dan penambahan elemen pada queue sudah pasti dilakukan pada dua ujung yang berbeda. Penambahan pasti dilakukan pada

STEI- ITB	IF2110-TB-03-02	Halaman 4 dari 18 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

akhir queue dan penghapusan elemen dilakukan pada elemen pertama. Dengan kata lain, elemennya tersusun secara FIFO (*First In First Out*).

Pada program kami, implementasi ADT Queue dilakukan dengan sedikit modifikasi menjadi PrioQueuePesanan untuk menyelesaikan persoalan daftar pesanan. Hal ini dilakukan karena daftar pesanan membutuhkan suatu keterurutan, dimana elemennya diurutkan berdasarkan waktu masuk dari pesanan. Pesanan yang pertama masuk adalah pesanan yang diantar terlebih dahulu. Implementasinya dilakukan pada file daftarPesanan.h dan daftarPesanan.c, PrioQueuePesanan memiliki atribut buffer yang menyimpan data bertipe Pesanan, idxHead untuk indeks elemen pertama dan idxTail untuk indeks elemen terakhir. Pesanan sendiri terdiri dari 4 atribut yaitu waktuMasuk, pickUp, dropOff, jenisItem, serta waktuHangus untuk item bertipe Persihable.

3.7 ADT Stack

ADT Stack ialah suatu list linier yang elemen puncaknya dikenal sebagai (TOP). Aturan penyisipan dan penghapusan selalu dilakukan di atas (TOP). TOP adalah satu-satunya alamat tempat terjadi, element stack tersusun secara LIFO (*Last In First Out*).

ADT stack digunakan pada persoalan Tas Mobita, karena pada Tas Mobita diterapkan aturan masuk item secara LIFO dengan kapasitas awal tiga item, juga item yang dapat di-drop off adalah hanya item di tumpukan teratas tas yang mana hal ini sangat cocok dengan ADT Stack. Pada program diimplementasikan pada file tas.h dan tas.c, tas memiliki atribut buffer (menyimpan elemen tas), idxTop (indeks top tas), dan capacity (kapasitas tas). Elemen tas itu sendiri memiliki 4 atribut, yaitu pickUp (tempat item diambil), dropOff (tempat tujuan item dikirim), itemType (jenis item), dan perishTime (waktu pesanan hangus bila jenisnya perishable item).

3.8 ADT Linked List

Linked list adalah list yang elemen-elemennya direpresentasikan dengan suatu Node yang terdiri dari info dan next. List diacu melalui address, untuk alamat elemen berikutnya (suksesor) diakses dengan next, sedangkan elemen terakhir ditandai dengan Next-nya menunjuk ke NIL.

ADT linked list dipakai pada persoalan To Do List dan In Progress List, karena apabila terdapat penyisipan atau penghapusan elemen cukup mudah untuk dilakukan yang mana hal tersebut seringkali dilakukan pada To Do List dan In Progress List. Info dari To Do List diterapkan dengan lima atribut yaitu, timeIn (waktu pesanan masuk), pickUp (tempat item diambil), dropOff (tempat tujuan item dikirim), itemType (jenis item), dan perishTime (waktu pesanan hangus bila jenisnya perishable item). Sedangkan untuk info In Progress List memiliki empat atribut yaitu, pickUp, dropOff, itemType, perishTime. Pada program To Do List diimplementasikan pada file toDo.h dan toDo.c, untuk In Progress List diimplementasikan pada file inProg_rev.h dan inProg_rev.c.

4 Program Utama

Program utama diawali dengan meminta input file konfigurasi dari pemain. Saat baru dimulai, posisi pemain akan berada di *Headquarters*, yaitu bangunan bersimbol angka 8. Pemain akan mendapatkan jumlah pesanan yang harus diselesaikan. Setelah itu, pemain dapat memasukkan *command* sesuai dengan yang diinginkan, dan akan mulai mendapatkan pesanan yang masuk ke *To Do List*. *Game* akan berakhir saat pemain telah berhasil mengantarkan seluruh pesanan dan kembali ke *Headquarters*. Akan ditampilkan pesan bahwa game sudah selesai, serta pencapaian pemain yaitu berapa jumlah item yang berhasil diantar dan berapa lama waktu yang telah dilampaui.

5 Data Test

5.1 Data Test 1

Pada data test pertama akan dites berbagai fitur sebagai berikut:

- Move dengan membawa Normal Item, Heavy Item, dan Perishable Item
- Pick up Normal Item, Heavy Item, dan Perishable Item
- Drop Off Heavy Item dengan kondisi masih terdapat Heavy Item lain pada tas (efek *speedboost* tidak aktif) dan Drop Off Heavy Item dengan efek *speedboost* aktif
- Buy gadget Kain Pembungkus Waktu sekaligus memakai gadget tersebut dari Inventory untuk mengembalikan sisa waktu hangus perishable item seperti kondisi awal
- Menjalankan *command* TO_DO untuk melihat To Do List
- Menjalankan *command* IN_PROGRESS untuk melihat item yang sedang dibawa Mobita
- Menjalankan *command* MAP untuk melihat Map dari permainan
- Menjalankan *command* HELP untuk melihat *command-command* yang tersedia pada permainan

Pertama Mobita move ke 'G' kemudian pick up heavy item dengan tujuan 'L', lalu move ke 'H' untuk pick up heavy item yang kedua dengan tujuan 'J'. Mobita move ke 'J' mengantarkan heavy item teratas. Pada move tersebut waktu bertambah 3 unit dikarenakan pada Tas Mobita terdapat 2 heavy items. Setelah drop off heavy item di titik 'J', efek *speedboost* tidak aktif karena masih terdapat heavy item lainnya pada Tas Mobita.

```

Mobita sekarang berada di titik J (5,12)!
Time : 7
Posisi sekarang : J
Speedboost : 0

ENTER COMMAND: DROP_OFF
Pesanan Heavy Item berhasil diantarkan
Uang yang didapatkan: 400 Yen

ENTER COMMAND: MOVE
Posisi yang dapat dicapai:
1. G (3,8)
2. H (3,14)
3. I (4,5)
4. L (7,10)
5. O (8,15)
Posisi yang dipilih? (ketik 0 jika ingin kembali)

ENTER NUMBER: 4

Mobita sekarang berada di titik L (7,10)!
Time : 9
Posisi sekarang : L
Speedboost : 0

```

Selanjutnya Mobita move ke titik 'L' untuk mengambil perishable item dengan tujuan 'F', kemudian Mobita move ke Headquarters untuk membeli Kain Pembungkus Waktu sekaligus memakainya.

```

ENTER COMMAND: IN_PROGRESS
Pesanan yang sedang diantarkan:
1. Perishable Item (Tujuan: F, Sisa waktu: 19)
2. Heavy Item (Tujuan: L)

```

Setelah memakai Kain Pembungkus Waktu, sisa waktu hangus perishable item teratas kembali seperti kondisi awal.

```

ENTER NUMBER: 1
Kain Pembungkus Waktu berhasil digunakan!

ENTER COMMAND: IN_PROGRESS
Pesanan yang sedang diantarkan:
1. Perishable Item (Tujuan: F, Sisa waktu: 25)
2. Heavy Item (Tujuan: L)

```

Berikutnya Mobita mengantarkan perishable item ke titik 'F' dan setelahnya bergerak ke titik 'L' untuk mengantarkan heavy item, karena tidak terdapat heavy item lain pada Tas Mobita, maka efek *speedboost* aktif.

```
Mobita sekarang berada di titik L (7,10)!
Time                : 21
Posisi sekarang    : L
Speedboost         : 0

ENTER COMMAND: DROP_OFF
Pesanan Heavy Item berhasil diantarkan
Uang yang didapatkan: 400 Yen
```

```
ENTER COMMAND: MOVE
Posisi yang dapat dicapai:
1. I (4,5)
2. J (5,12)
3. K (6,3)
4. N (8,6)
5. P (9,13)
Posisi yang dipilih? (ketik 0 jika ingin kembali)

ENTER NUMBER: 3

Mobita sekarang berada di titik K (6,3)!
Speedboost activated, waktu tidak bertambah
Time                : 21
Posisi sekarang    : K
Speedboost         : 9
```

Dapat dilihat, setelah Mobita berpindah tempat pada hitungan pertama dari titik 'L', waktu permainan tidak bertambah karena adanya efek *speedboost*. Setelahnya mobita bergerak ke titik 'F' dengan efek *speedboost* yang masih aktif.

Kemudian Mobita melihat To Do List dan Inventory yang seharusnya kondisi Inventory kosong.

```
ENTER COMMAND: TO_DO
Pesanan pada To Do List:
1. P -> E (Normal Item)

ENTER COMMAND: INVENTORY
1. -
2. -
3. -
4. -
5. -
Gadget mana yang ingin digunakan? (ketik 0 jika ingin kembali)
```

Setelah itu dijalankan command MAP dan HELP, dengan hasil sesuai yang diharapkan.

```
ENTER COMMAND: MAP
*****
*8      C  D B*
*  E      *
*F      G  H  *
*    I      *
*          J  *
*  K      *
*          L  *
* M    N      O*
*          P  *
*A Q      *
*****
```

```
ENTER COMMAND: HELP
1. MOVE      -> Untuk berpindah ke lokasi selanjutnya
2. PICK_UP   -> Untuk mengambil item di lokasi terkini
3. DROP_OFF  -> Untuk mengantarkan item ke lokasi terkini
4. MAP       -> Untuk menampilkan peta permainan
5. TO_DO     -> Untuk menampilkan pesanan yang masuk ke To Do List
6. IN_PROGRESS -> Untuk menampilkan pesanan yang sedang dikerjakan
7. BUY       -> Untuk menampilkan gadget yang dapat dibeli lalu membelinya
8. INVENTORY -> Untuk melihat gadget yang dimiliki dan menggunakannya
9. SAVE_GAME -> Untuk menyimpan state dari permainan yang sedang dijalankan
10. HELP     -> Untuk mengeluarkan list command dan kegunaannya
```

Kemudian Mobita bergerak ke Headquarters dengan efek *speedboost* yang masih ada. Sesampainya di Headquarters, Mobita membeli Pintu Kemana Saja dengan jumlah 2.

```
ENTER COMMAND: BUY
Uang Anda sekarang: 3400 Yen
Gadget yang tersedia:
1. Kain Pembungkus Waktu (800 Yen)
2. Senter Pembesar (1200 Yen)
3. Pintu Kemana Saja (1500 Yen)
4. Mesin Waktu (3000 Yen)
Gadget mana yang ingin kau beli? (ketik 0 jika ingin kembali)

ENTER NUMBER: 3
Pintu Kemana Saja berhasil dibeli!
Uang Anda sekarang: 1900 Yen.
```

Kemudian Pintu Kemana Saja dipakai untuk pick up normal item dengan tujuan 'E'. Untuk drop off normal item tersebut, digunakan Pintu Kemana Saja yang kedua kalinya ke titik 'E'.

Karena semua barang telah diantarkan, maka permainan akan diakhiri dengan kembalinya Mobita ke Headquarters, dan hasil akhir permainan akan ditampilkan.

```
Mobita sekarang berada di titik 8 (1,1)!
Time : 23
Posisi sekarang : 8
Speedboost : 6

Permainan selesai
Semua pesanan telah diantarkan
Jumlah pesanan diantar : 4
Waktu : 23
```

5.2 Data Test 2

Pada test kedua ini diinisialisasikan Uang 10.000 Yen, serta waktu awal 50 satuan untuk keperluan testing. Selanjutnya user melakukan pick up 4 item sekaligus pada titik E. Pada pick up ke-4 tas penuh karena sudah berisi 3 pesanan (sejumlah capacity yaitu 3).

```
Mobita sekarang berada di titik E (2,3)!
Time           : 51
Posisi sekarang : E
Speedboost      : 0

ENTER COMMAND: PICK_UP
Pesanan berupa Normal item berhasil diambil!
Tujuan pesanan: A

ENTER COMMAND: PICK_UP
Pesanan berupa Heavy item berhasil diambil!
Tujuan pesanan: B

ENTER COMMAND: PICK_UP
Pesanan berupa Perishable item berhasil diambil!
Tujuan pesanan      : C
Sisa waktu hangus    : 20

ENTER COMMAND: PICK_UP
Tas penuh!
```

User kemudian menggunakan senter pembesar yang sudah dibeli sebelumnya. Kapasitas bertambah 2 kali lipat menjadi 6 sehingga pick up selanjutnya dapat dilakukan.

```
ENTER COMMAND: INVENTORY
1. Senter Pembesar
2. Mesin Waktu
3. -
4. -
5. -
Gadget mana yang ingin digunakan? (ketik 0 jika ingin kembali)

ENTER NUMBER: 1
Senter Pembesar berhasil digunakan!

ENTER COMMAND: PICK_UP
Pesanan berupa Normal item berhasil diambil!
Tujuan pesanan: D
```

User Kembali ke HQ, saat ini waktu 53.

```
Mobita sekarang berada di titik 8 (1,1)!
Time           : 53
Posisi sekarang : 8
Speedboost      : 0
```

Setelah menggunakan mesin waktu yang telah dibeli sebelumnya, waktu berkurang 50 satuan menjadi 3. Setelah itu dilakukan move dengan kondisi ada 1 heavy item sehingga bertambah 2 lagi. Sehingga pada keadaan akhir waktu yang dicapai adalah 5 satuan waktu.

```
ENTER NUMBER: 2
Mesin Waktu berhasil digunakan!

ENTER COMMAND: MOVE
Posisi yang dapat dicapai:
1. C (1,9)
2. E (2,3)
3. F (3,1)
Posisi yang dipilih? (ketik 0 jika ingin kembali)

ENTER NUMBER: 1

Mobita sekarang berada di titik C (1,9)!
Time           : 5
Posisi sekarang : C
Speedboost      : 0
```

Dilakukan save game, kemudian load game. Permainan Kembali dalam kondisi yang sama dengan uang terakhir adalah 5800 Yen.

```
ENTER COMMAND: BUY
Uang Anda sekarang: 5800 Yen
Gadget yang tersedia:
1. Kain Pembungkus Waktu (800 Yen)
2. Senter Pembesar (1200 Yen)
3. Pintu Kemana Saja (1500 Yen)
4. Mesin Waktu (3000 Yen)
Gadget mana yang ingin kau beli? (ketik 0 jika ingin kembali)
```

6 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Start Game	Menguji file reader dan mulai game	Ketikkan “START_GAME” lalu ketikkan nama file config yang ada.	Data Test 1	File terbuka dan berhasil di baca	File terbuka dan berhasil di baca
2	Move	Berpindah tempat dalam keadaan tidak membawa heavy item	Input command “MOVE”, pilih titik tujuan.	Data Test 1	Berhasil pindah ke titik yang diinginkan dan waktu bertambah 1 unit	Pemain berpindah posisi, unit waktu bertambah 1.
3	Pick Up	Mengambil item pesanan di bangunan yang ditentukan	Move ke bangunan yang sudah menyediakan pesanan (time > waktu pesanan masuk)	Data Test 1	Berhasil mengambil pesanan dan dimasukkan ke tas	Berhasil mengambil pesanan dan dimasukkan ke tas
4	Pick Up dan Tas	Menguji pengambilan item pesanan saat tas penuh	Melakukan pickup saat kapasitas tas penuh	Data Test 2	Ditampilkan Pesan “Tas Penuh” jika player mencoba mengambil pesanan	Ditampilkan Pesan “Tas Penuh” jika player mencoba mengambil pesanan
5	Drop off	Mengantarkan item pesanan ke bangunan yang ditentukan	Pickup item dan pastikan merupakan item terakhir yang diambil (top tas) lalu move ke bangunan yang ditentukan	Data Test 1	item pesanan di keluarkan dari tas dan player mendapat gold / effect	item pesanan di keluarkan dari tas dan player mendapat gold / effect
6	Move dengan 2 heavy items	Melihat penambahan unit waktu saat berpindah dengan membawa heavy item	Sebelumnya telah dipick up 2 heavy item. Lalu move.	Data Test 1	Unit waktu bertambah 3	Pemain berpindah posisi, unit waktu bertambah 3 dari sebelum move.
7	Menggunakan Mesin Waktu	Menguji pengurangan waktu dengan mesin waktu	Sebelumnya, beli mesin waktu di Headquarters	Data Test 2	Waktu global berkurang 50 satuan	Waktu global berkurang 50 satuan
8.	Menggunakan Kain Pembungkus waktu	Mengembalikan sisa waktu perishable item ke semula menggunakan	Membeli kain pembungkus waktu di Headquarters. Telah pick up perishable item, dan	Data Test 1	Sisa waktu perishable item kembali ke semula (di daftar pesanan)	Sisa waktu perishable item yang tinggal 19

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		kain pembungkus waktu	merupakan item teratas dalam in progress list. Input command "INVENTORY"			kembali menjadi 25.
9	In Progress List	Melihat isi in progress list	Input command "IN_PROGRESS"	Data Test 1	Menampilkan list pesanan yang telah di pick up.	Terlihat list pesanan yang sedang diantarkan, termasuk sisa waktu untuk perishable item.
10	To Do List	Melihat isi to do list	Input command "TO_DO"	Data Test 1	Menampilkan pesanan yang sudah masuk namun belum di pick up.	Terlihat pesanan yang belum di pick up.
11.	Inventory	Melihat isi list inventory setelah menggunakan semua gadget yang dibeli.	Gadget yang dibeli sebelumnya telah digunakan semua. Lalu input command "INVENTORY"	Data Test 1	Isi Inventory gadget kosong.	Isi Inventory gadget kosong.
12.	Map	Menampilkan Map sesuai file konfigurasi	Input command "MAP"	Data Test 1	Map ditampilkan dengan nama-nama bangunan + warna yang sesuai	Map ditampilkan dengan nama-nama bangunan + warna yang sesuai
13.	Help	Menampilkan isi help	Input command "HELP"	Data Test 1	Menampilkan seluruhh command game dan penjelasannya	Tertulis seluruh command game beserta penjelasannya.
14.	Save game	Menyimpan game ke dalam sebuah file	Input command "SAVE_GAME" kemudian memasukkan input nama file	Data Test 2	File yang berisi config dan state permainan disimpan dalam sebuah file .txt	Seluruh config dan state tertulis lengkap dalam file .txt
15.	Load game	Melanjutkan game dengan state yang telah di save sebelumnya.	Telah terdapat file .txt yang telah disave dari sebelumnya. Input command "LOAD_GAME" dan input nama file.	Data Test 2	Game dimulai dengan kondisi yang sebelumnya telah di save.	Game dimulai dengan kondisi yang sebelumnya telah di save.
16.	End game	Menyelesaikan sebuah game dan menampilkan score akhir	Selesaikan semua Todo dan In Progress, kemudian	Data Test 1	Permainan selesai, Waktu akhir serta jumlah pesanan	Permainan selesai serta waktu dan jumlah

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			kembali ke headquarters		diantarkan ditampilkan	pesanan diantarkan muncul pada layar

7 Pembagian Kerja dalam Kelompok

No	Nama	NIM	Tugas
1.	Suryanto	13520059	<ul style="list-style-type: none"> - Baca file konfigurasi - Daftar Pesanan - Save dan Load
2.	Ziyad Dhia Rafi	13520064	<ul style="list-style-type: none"> - Pick Up - Drop Off
3.	Farnas Rozaan Iraquee	13520067	<ul style="list-style-type: none"> - Move - Map
4.	Lyora Felicya	13520073	<ul style="list-style-type: none"> - Buy - Inventory
5.	Rava Naufal Attar	13520077	<ul style="list-style-type: none"> - To Do List - In Progress List - Help






8 Lampiran

8.1 Deskripsi Tugas Besar







Dengan adanya pandemi COVID-19, usaha orang tua Mobita mengalami penurunan pendapatan. Setelah diringankannya PPKM, Mobita ingin membantu orang tuanya mendapat penghasilan menjadi seorang kurir. Tugas kalian adalah membantu Mobita membuatkan aplikasi yang akan membantunya melacak pesanan, navigasi peta, mengambil dan menurunkan barang.

8.2 Notulen Rapat

Asistensi I

Tanggal : 28/10/2021	Catatan Asistensi: Pedoman Tubes dan Spek penting untuk dibaca, harus dibaca dari sekarang. Tugas dibagi per-ADT, kalo udah ada ADT dari awal lebih enak jalan kedepannya. Perlu <i>rename</i> beberapa fungsi agar tidak konflik. Kategorikan penggunaan suatu ADT di fungsi tertentu. Pembagian folder dan isinya bagaimana? Contoh: Folder <i>stack</i> berisi, file .h, .c, dan drivernya. Apakah perlu ADT baru? ADT dasar sudah cukup, mungkin modifikasi nama fungsi jika diperlukan agar tidak konflik. Setiap ADT harus ada di setiap fitur. Pakai ADT biasa aja jika masih dapat terselesaikan. Harus pakai mesin karakter/kata. Untuk membaca komen dan program. Saran: Pedoman Tubes jangan lupa, jangan <i>deadliner</i> , pastikan buat ADT ter- <i>compile</i> , jangan meremehkan laporan, semangat!
Tempat : Google Meet	
Kehadiran Anggota Kelompok: No NIM Tanda tangan 1 13520059  2 13520064  3 - 4 13520073  5 13520077 	
	Tanda Tangan Asisten: 

Asistensi II

Tanggal : 10/11/2021	Catatan Asistensi: Jangan lupa perhatikan pedoman tubes. Cepat diselesaikan tiap bagiannya karena udah mau <i>deadline</i> , kalau misal masih kesulitan coba mulai nyicil laporan dulu aja. Untuk daftar pesanan, tidak semua dapat masuk <i>To Do List</i> . Untuk penyesuaian antar ADT bisa dilakukan membuat ADT <i>game console</i> yang dikumpulkan ataupun disesuaikan di <i>main</i> program, tinggal dipertimbangkan akan kerapihan kodenya. Segera diselesaikan tubesnya, karena akan ada <i>deadline-deadline</i> dan tubes lain yang bakal muncul kedepannya, semoga selesai tubesnya.
Tempat : Google Meet	
Kehadiran Anggota Kelompok: No NIM Tanda tangan 1 13520059  2 13520064  3 13520067  Farnas Rozaan .I 4 13520073  5 13520077 	
	Tanda Tangan Asisten: 

8.3 Log Activity Anggota Kelompok

Tanggal	Nama	NIM	Aktivitas
25/10/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Memahami spesifikasi tugas - Membuat repository github - Mengatur jadwal asistensi I
	Ziyad	13520064	
	Farnas Rozaan	13520067	
	Lyora	13520073	
	Rava	13520077	
28/10/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Pembagian tugas untuk rename dan menyesuaikan ADT agar tidak terjadi konflik
	Ziyad	13520064	
	Lyora	13520073	
	Rava	13520077	
5/11/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Pembagian tugas untuk setiap orang. Tugas dibagi per-fungsi dari program.
	Ziyad	13520064	
	Lyora	13520073	
	Rava	13520077	
15/11/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Meet untuk update progress dan menyesuaikan pekerjaan satu sama lain.
	Ziyad	13520064	
	Farnas Rozaan	13520067	
	Lyora	13520073	
	Rava	13520077	
16/11/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Menggabungkan main program dengan mesin karakter dan kata
	Ziyad	13520064	
	Rava	13520077	
17/11/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Merapihkan keseluruhan program - Debugging keseluruhan program - Menambahkan proses bila <i>game</i> berakhir
	Ziyad	13520064	
	Farnas Rozaan	13520067	
	Lyora	13520073	
	Rava	13520077	
18/11/2021	Suryanto	13520059	<ul style="list-style-type: none"> - Menambah fitur save game dan load game - Menyusun Laporan
	Ziyad	13520064	
	Lyora	13520073	
	Rava	13520077	