

LAPORAN TUGAS BESAR

IF2110/Algoritma dan Struktur Data

Let's Finish This


Dipersiapkan oleh:

Kelompok 08

1. Muhammad Farhan Kemal - 13513085
2. Fanda Yuliana Putri - 13514023
3. Bervianto Leo P. - 13514047
4. Diastuti Utami - 13514071
5. Muhammad Az-zahid A. S - 13514095

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-08-01</i>		12
		<i>Revisi</i>	<i>00</i>	<i>29 November 2015</i>

Daftar Isi

1 Ringkasan.....	3
2 Penjelasan Tambahan Spesifikasi Tugas	3
2.1 Landmark	3
2.2 Keluar dari Penjara.....	3
3 Struktur Data (ADT)	4
3.1 List Berkait.....	4
3.2 Tabel Kontigu.....	4
3.3 Lain-lain	5
4 Program Utama	5
5 Algoritma-Algoritma Menarik.....	5
5.1 Traversal dalam Board (List Sirkuler).....	5
5.2 Algoritma Validasi dalam Command.....	5
5.3 Algoritma Menukar Informasi dalam List	6
6 Data Test	6
6.1 Test Jual-Beli kepada Bank.....	6
6.2 Test Jual-Beli antar Pemain.....	7
6.3 Test Kartu Chance	7
6.4 Test Baca-Tulis File	8
6.5 Test Perintah Ditempat Tertentu	8
6.6 Test Output Status Terbaru	8
6.7 Test Perintah Otomatis dan Menjalankan Pemain	9
7 Test Script	9
8 Pembagian Kerja dalam Kelompok	10
9 Lampiran	10
9.1 Deskripsi Tugas Besar 2.....	10
9.2 Notulen Rapat.....	11
9.3 Log Activity Anggota Kelompok.....	11

1 Ringkasan

Permainan *board* seperti monopoli merupakan permainan yang populer bahkan permainan ini memiliki versi virtual. Permainan LINE Let's Get Rich sebagai contohnya, permainan ini sangat populer dan banyak yang memainkan *game* ini. Oleh karena itu untuk mencoba membuat *game board* sederhana, *game* tersebut diimplementasikan menggunakan bahasa C dengan tampilan berbasis teks dan interaksi pemain dengan permainan menggunakan *keyboard*. Masing-masing player akan memiliki suatu identitas yang unik.

Secara sederhana permainan monopoli atau seperti LINE Let's Get Rich ini dapat diimplementasikan dengan struktur data yang unik. Struktur data seperti giliran pemain maupun data petak dapat disimpan menggunakan struktur data list sirkuler. Sedangkan untuk mengacak sebuah kartu dapat digunakan sebuah tabel kontigu dengan akses elemen secara acak. Oleh karena itu permainan ini dapat diimplementasikan dengan baik.

Program utama dalam game yang telah dibuat akan menjalankan suatu perintah membaca “perintah” dari pengguna. Sebelum memasuki permainan terlebih dahulu memilih ingin melanjutkan permainan sebelumnya “jika ada” atau memulai dari awal. Setelah itu akan masuk ke dalam permainan dengan membaca perintah dari pengguna dan seterusnya hingga permainan berakhir. Permainan akan berakhir jika ada pemain menggunakan perintah “exit” atau tersisa satu pemain.

Secara umum hasil dari *game board* yang telah diimplementasikan dengan baik. Penggunaan struktur data list sirkuler maupun tabel kontigu tersebut sudah sesuai. Namun sedikit sulit untuk mengintegrasikan suatu perintah otomatis yang pada dasarnya bukan suatu masukan pengguna dan juga sedikit sulit mengatur data yang akan disimpan. Namun dapat diambil simpulan bahwa permainan ini menarik dan sudah diimplementasikan dengan baik.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Landmark

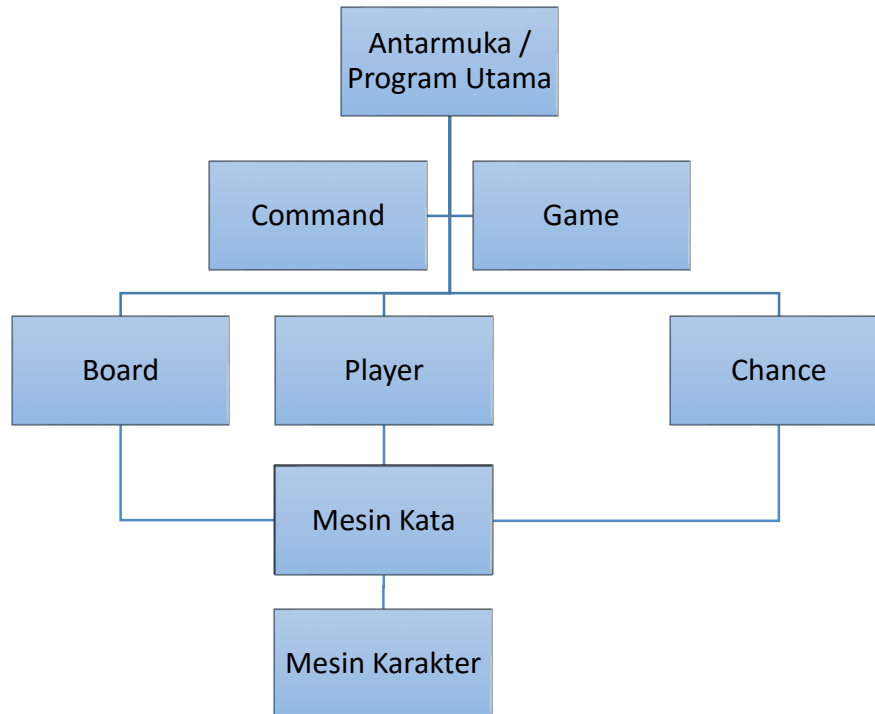
Landmark memberikan batasan level lebih dari biasanya. Perlu menambahkan spesifikasi pada fitur buy secara khusus. Pada fitur buy perlu mengecek keadaan level bangunan, jika level bangunan sudah mencapai landmark (level 5) tidak dapat dibeli secara langsung (merebut petak) kecuali pemain telah sell offered.

2.2 Keluar dari Penjara

Untuk keluar dari penjara memiliki dua fitur, pertama memakai kartu *chance* sedangkan kedua dengan membayar. Pada bagian membayar kurang jelas dengan menggunakan perintah atau otomatis. Sehingga fitur tersebut dibuat dengan dua keadaan. Pertama, pemain memasukan perintah untuk membayar. Kedua, pemain tidak membayar hingga tiga kali kesempatan lalu akan membayar secara otomatis. Biaya keluar dari penjara yaitu 10 persen dari uangnya saat itu. Sehingga selalu dapat membayar keluar dari penjara.

3 Struktur Data (ADT)

Struktur data yang digunakan pada persoalan yang ditemukan dalam permainan ini yaitu list berkait dan tabel kontigu, dan lain-lain. Berikut ini diagram keterhubungan setiap modul yang terdapat dalam *game* ini.



Gambar 1 Hirarki modul

Modul tersebut posisi atas memiliki kebutuhan modul yang dibawah. Misalkan mesin kata yang membutuhkan mesin karakter dan seterusnya. Adapun detil dari struktur data yang digunakan sebagai berikut.

3.1 List Berkait

List berkait yang digunakan merupakan list berkait sirkuler. List sirkuler digunakan agar memiliki perputaran giliran list yang akan dieksekusi serta memiliki suatu alamat yang dapat diputari. List berkait sirkuler ini sangat diperlukan untuk giliran pemain. List berkait sirkuler digunakan dalam permainan dalam modul *player.h* dan *board.h*. *ListOfPlayer* menunjukan sebuah giliran pemain sehingga memudahkan perputaran giliran dengan menggunakan list sirkuler. Sedangkan *Board* merepresentasikan “board” yang sebenarnya.

3.2 Tabel Kontigu

Tabel kontigu yang digunakan eksplisit statis. Tabel kontigu ini digunakan untuk menyimpan informasi kartu *chance* yang akan didapatkan. Tabel kontigu eksplisit statis dipilih dikarenakan mudah mendapatkan informasi jumlah kartu *chance* dan mudah untuk mengakses kartu *chance* karena cukup menggunakan *index*. Implementasi dari tabel kontigu digunakan dalam *chance.h*.

3.3 Lain-lain

Selain dari kedua struktur data utama. Terdapat beberapa file header lain, seperti *global.h* yang berisi mengambil beberapa *library* lain seperti *stdio.h* yang dipakai pada permainan ini seperti perintah dasar mencetak ke layar, membaca input pengguna, dan lain-lain serta sebagai konstanta global dan selektor global untuk list sirkuler yang digunakan. Selain itu terdapat *stdlib.h* yang digunakan mengenai list. *Mesinkata.h* mengatur pembacaan suatu *file* kata per kata yang dipakai permainan untuk membaca suatu *file* permainan yang sudah disimpan. Serta digunakan juga *string.h* yang berisi mengenai struktur *string*, yang digunakan dalam permainan ini untuk menyalin *string* masukan pengguna dan juga *concat* untuk membuat suatu masukan sesuai format nama *file*. Lalu *time.h* untuk *generate* suatu *random* karena jika tanpa *time* tersebut, *random* akan sangat terlihat terpola. Dalam *game.h* terdapat berbagai fungsi dan prosedur penyusun game ini. Sedangkan pada *command.h* memiliki suatu prosedur validasi suatu perintah dari pengguna. Terakhir *boolean.h* yang berisi deklarasi definisi suatu *boolean* yang ditetapkan.

4 Program Utama

Program utama sebelum memulai akan memberi pilihan kepada pengguna untuk memulai dari awal (*new game*) atau load dari sebuah file yang sudah disimpan (jika ada). Selanjutnya akan segera *load file* sesuai yang diminta, jika memulai dari awal akan me-*load default* dan sebagainya. Setelah berhasil, pemain pertama atau pemain pada giliran saat disimpan datanya akan dapat memulai permainan dengan memasukkan sebuah perintah. Perintah akan divalidasi menggunakan suatu prosedur validasi *command* atau perintah. Jika perintah tersedia akan dieksekusi perintah tersebut. Program akan berhenti ketika salah satu player melakukan exit atau 3 pemain kalah dalam permainan. Saat permainan berlangsung dapat melakukan *save* atau menyimpan data permainan. Setelah selesai atau berhenti dari permainan, data yang sudah dialokasi dihapus dari memori.

5 Algoritma-Algoritma Menarik

5.1 Traversal dalam Board (List Sirkuler)

Algoritma yang menarik yaitu dalam mencetak *board* karena memiliki tiga bagian algoritma. Algoritma bagian atas atau baris 1 merupakan traversal biasa sehingga masih mudah dilakukan. Namun yang menarik yaitu setelah baris 1, baris 2 hingga baris $n-1$ (n sebagai akhir) sedikit sulit karena perlu menjaga dua identitas atau dua traversal hingga $n-1$. Dalam baris perlu mencetak elemen baris per baris, sehingga perlu mengulang traversal dari awal. Sedangkan baris 2 dan seterusnya mencetak dari elemen per elemen setelah selesai mengganti elemen selanjutnya. Namun dalam kolom kiri perlu mundur sedangkan kanan tetap maju. Sehingga mempersulit dan tidak ada fitur mundur karena hanya menggunakan list sirkuler. Setelah baris terakhir, algoritma mirip dengan baris pertama namun, berbeda karena traversal mundur.

5.2 Algoritma Validasi dalam Command

Algoritma ini menarik karena dalam memvalidasi suatu command akan semakin panjang jika semakin banyak suatu perintah yang dapat digunakan. Terkadang suatu command akan terlewat jika tidak memperhatikan dengan baik. Namun validasi command ini sangat sederhana sehingga cukup menarik.

5.3 Algoritma Menukar Informasi dalam List

Algoritma ini cukup menarik, karena kasus yang ditangani mengenai suatu petak ditempati oleh pemain. Dengan struktur data yang dibuat menggunakan list dengan infotype letak pemain disiapkan empat array yang akan diisi oleh pemain. Sedikit menyulitkan untuk memindahkan pemain dari satu petak ke petak lain. Sehingga digunakan prosedur untuk menghapus informasi letak pemain sebelumnya dan juga menaruh informasi letak pemain setelahnya. Penukaran ini menarik karena informasi mengenai player akan tetap ada hanya mengatur tempat sebelumnya dan tempat yang dituju.

6 Data Test

Berikut ini data test yang digunakan beserta fitur yang di test dan beberapa penjelasan mengenai test tersebut.

6.1 Test Jual-Beli kepada Bank

Fitur yang dites yaitu *buy*, *upgrade* dan *sell bank*. Hasil yang seharusnya diberikan sebagai berikut.

> **buy**

- Keluaran berhasil membeli

> Selamat, kota Seoul kini menjadi milikmu!

> Level bangunan 1

> Sisa uangmu : 1850K

> Player A mendapatkan hasil penjualan sebesar 150K

- Keluaran saat mencoba membeli petak yang tidak dijual

> Petak yang Anda tempati tidak dapat dibeli

- Keluaran saat mencoba membeli petak milik sendiri

> Lokasi ini sudah milik Anda.

- Keluaran saat mencoba membeli petak milik orang lain

> Maaf Anda tidak bisa membeli langsung. Silahkan hubungi pemilik petak ini.

- Keluaran saat uang tidak cukup

> Uangmu tidak cukup untuk membeli kota ini.

> **upgrade**

- Keluaran saat cukup membeli petak

> Selamat kini bangunanmu di Seoul memiliki level 2.

> Sisa uangmu : 1800K

- Keluaran saat tidak memiliki uang yang cukup

> Uangmu tidak cukup untuk upgrade kota ini.

> **sell bank Seoul**

> Seoul dijual ke bank dengan harga 800K

STEI- ITB

IF2110-TB-08-01

Halaman 6 dari 12 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.

> *Uangmu sekarang 1800K*

Buy telah mengeluarkan instruksi dengan baik. Belum ada masalah dalam tiga fitur tersebut. Adapun data test yang digunakan sebagai berikut :

1. Kondisi kota belum ada pemilik.
2. Pemain sedang di kotak selain kota dan tempat parawisata.
3. Kondisi kota sudah ada pemilik.
4. Kota masih belum menjadi milik pemain.
5. Kota sudah menjadi milik pemain.
6. Uang tidak cukup membeli atau upgrade.

6.2 Test Jual-Beli antar Pemain

Fitur yang dites yaitu *sell offered*, *show offered* dan *buy offered*. Hasil yang seharusnya diberikan sebagai berikut.

> *sell Seoul*

> *Seoul dimasukan ke list penjualan. Jika Seoul terjual, kamu akan mendapat uang hasil penjualan.*

> *show offered*

> *Seoul, harga 1800K.*

> *end turn*

> *buy offered Seoul*

> *Seoul menjadi milikmu.*

> *Sisa uangmu : 2200K*

Percobaan tersebut dengan kondisi uang pemain pembeli memiliki uang yang cukup untuk membeli petak yang ditawarkan. Data test yang digunakan, uang pemilik petak 1000K, uang pembeli 4000K, harga petak 1800K. Mendapatkan hasil uang pemilik petak telah bertambah menjadi 2800K, uang pembeli menjadi 2200K, serta petak tersebut telah menjadi milik pembeli.

6.3 Test Kartu Chance

Fitur yang dites yaitu *off*, *protect*, *free me* dan *free tax*. Hasil yang seharusnya diberikan sebagai berikut.

> *off Jakarta*

> *Kota Jakarta milik Player B listriknya telah dimatikan. Biaya sewa di kota Jakarta = 0.*

> *Player B memiliki kartu perlindungan*

> *end turn*

> *protect Jakarta*

> *Player B menggunakan kartu perlindungan. Biaya sewa di kota Jakarta kembali normal.*

> *end turn*

> *roll dice*

> *free tax*

> *Anda sedang di petak pajak.*

```

> Anda telah menggunakan kartu bebas pajak.
> end turn
> roll dice
> Anda mendapatkan kartu masuk penjara. Maaf Anda dipenjara.
> end turn
> end turn
> free me
> Anda bebas dari penjara.

```

Kartu chance telah berfungsi dengan baik. Beberapa alur telah dilakukan, perintah-perintah di atas merupakan perintah alur yang penting. Data yang digunakan, *player A* memiliki kartu mati lampu, *player B* memiliki kartu perlindungan, *player C* memiliki kartu bebas pajak, *player D* memiliki kartu bebas dari penjara.

6.4 Test Baca-Tulis File

Fitur yang dites yaitu *load* dan *save*. Hasil yang seharusnya diberikan sebagai berikut.

```

> load test
> Permainan di-load dari file test-map.txt dan test-player.txt
...
> save new
> Permainan telah disimpan ke file new-map.txt dan new-player.txt

```

Perintah tersebut telah baik dilakukan. Jika *file* tidak ditemukan saat *load* sudah ditangani dengan baik. *Save* akan melakukan penulisan *file* baru jika *file* tersebut belum ada. Sehingga dapat membuat banyak *saved game*. Data yang digunakan yaitu *file text test-map.txt* dan *test-player.txt* sudah ada dan *file new-*.txt* belum ada.

6.5 Test Perintah Ditempat Tertentu

Fitur yang dites yaitu *host* dan *travel*. Hasil yang seharusnya diberikan sebagai berikut.

```

> host Seoul
> Seoul menjadi host World Cup. Uang sewa di Jakarta menjadi 1000K.
...
> travel Jakarta
> Player A sampai di Jakarta.

```

Perintah tersebut telah baik dilakukan hanya saja memiliki kelemahan, *host* dan *travel* hanya dapat melakukan perintahnya sendiri atau dengan kata lain saat sudah sampai *World Cup* atau *World Travel* akan membuat perintah lain tidak dapat digunakan sedikit menyulitkan pemain untuk melihat *board* dahulu sebelum pindah.

6.6 Test Output Status Terbaru

Fitur yang dites yaitu *leaderboard*, *board*, dan info petak. Hasil yang seharusnya diberikan sebagai berikut.

```

> leaderboard
> Player A 2000K
> Player C 1800K
> Player D 1500K

```


> *Player B 1000K*

> **board**

.....

> **info Jakarta**

> *Kota Jakarta*

> *Milik : A*

> *Harga sewa : 100K*

> *Harga ambil alih : 800K*

> *Harga upgrade : 1000K*

Status terbaru telah ditampilkan dengan baik. Pada info sudah memiliki tampilan yang menyesuaikan kondisi. Data yang dites Jakarta sudah dimiliki A, beberapa kota yang belum ada pemilik, *player A* memiliki uang dan aset lebih banyak. Posisi *player* yang sudah berubah.

6.7 Test Perintah Otomatis dan Menjalankan Pemain

Fitur yang dites yaitu beberapa perintah otomatis serta *roll dice*. Hasil yang seharusnya diberikan sebagai berikut.

- Setelah melewati atau menempati *start* menambahkan uang player tersebut
- Setelah melewati *world cup*, pemain yang telah mengaktifkan *world cup*, kota yang telah diberikan *tag world cup* telah hilang.
- Pemain dapat berpindah tempat setelah *roll dice*.
- Pemain mendapatkan kartu chance saat berhenti di petak kartu chance.
- Pemain membayar pajak jika tidak ada kartu bebas pajak dan telah sampai di tempat bayar pajak.
- Pemain mendapatkan bonus uang saat sampai petak bonus.
- Pemain otomatis masuk penjara jika sudah mendapat kartu masuk penjara.
- Petak yang sudah dimatikan menyala kembali saat player yang bersangkutan telah melewati petak *start*.

Data yang digunakan tidak ada, hanya menjalankan roll dice hingga tiba di petak-petak tersebut dan sudah .

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Buy, upgrade, sell to bank	Menyiapkan fitur jual-beli secara langsung.	Beli tanah – upgrade Rumah – jual tanah ke Bank	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai
2	Sell offered, show offered, dan buy offered	Menyiapkan fitur jual-beli untuk interaksi antar pemain.	Tanah sudah dibeli – sell offered – show offered – ganti pemain – buy offered	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai
3	Off, Protect, Free Me, Free Tax	Menyiapkan penggunaan kartu <i>chance</i> .	Mendapatkan kartu chance yang dimaksud – digunakan kartu chance tersebut	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
4	Load dan save	Menyiapkan seluruh data sehingga mudah untuk dicoba dan mulai dimainkan serta dalam inisialisasi.	Mencoba load file tertentu yang sudah tersedia – merubah data dengan bermain – menyimpan perubahan tersebut	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai
5	Host dan travel	Mempersiapkan perintah-perintah hanya di petak tertentu.	Pemain sudah di petak host atau travel – mencoba selain perintah yang tersedia dalam petak tersebut – mencoba perintah host atau travel	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai
6	Leaderboard, board, dan info petak	Memperlihatkan status saat ini	Setiap pemain mencoba perintah tersebut	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai
7	Roll dice dan fitur-fitur otomatis atau tanpa perintah	Mempersiapkan permainan hingga tahap akhir	Memainkan permainan hingga selesai dengan kondisi yang sudah ditentukan.	Tersedia dalam data test	Dijelaskan dalam data test.	Sudah sesuai

8 Pembagian Kerja dalam Kelompok

No	Nama	NIM	Tugas
1	Muhammad Farhan Kemal	13513085	Membuat fitur <i>show board, roll dice, show offered.</i>
2	Fanda Yuliana Putri	13514023	Membuat fitur info petak, <i>buy offered, upgrade, dan off(mati lampu).</i>
3	Bervianto Leo P.	13514047	Membuat fitur <i>load dan save, end turn, free me, free tax, travel</i> , dan membuat antarmuka serta mengintegrasikan semua prosedur.
4	Diastuti Utami	13514071	Membuat fitur <i>sell, sell offered, bayar pajak, dan host(world cup).</i>
5	Muhammad Az-zahid A. S	13514095	Membuat fitur <i>random chance, get bonus, buy, leaderboard</i> , dan masuk penjara.

9 Lampiran

9.1 Deskripsi Tugas Besar 2

Dalam tugas besar ini, Anda diminta untuk membuat sebuah program yang mensimulasikan LINE Let's Get Rich, yaitu Let's Get Wealthy. Let's Get Wealthy diimplementasikan dalam bahasa C

STEI- ITB	IF2110-TB-08-01	Halaman 10 dari 12 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.		

dengan menggunakan command line dan input dari keyboard. Masing-masing player menggunakan huruf A-D sebagai identifikasi, dan untuk menandai posisinya dan kepemilikan asset.

9.2 Notulen Rapat

- Berikut ini hasil asistensi pertama :
 - a. Posisi pemain lebih baik di dalam petak
 - b. Beberapa alternatif pembuatan *board*
 - c. Penamaan kartu *chance* tetap menggunakan *identifier* boleh hanya karakter atau bilangan.
- Belum ada tatap muka kelompok secara resmi, komunikasi menggunakan grup chat dan pertemuan personal atau beberapa dari keseluruhan anggota yang kurang resmi. Beberapa hasil dari pembicaraan seperti penentuan struktur data yang digunakan dan pembagian tugas serta pembahasan mengenai beberapa fitur yang menjadi tugasnya masing-masing saat merasa membingungkan.

9.3 Log Activity Anggota Kelompok

No	Nama	Tanggal - Durasi	Kegiatan
1.	BLP	12 November 2015 – 3 jam	Menyelesaikan fitur <i>load</i> dan <i>save</i> .
2.	FYP	13 November 2015 – 2 jam	Menyelesaikan fitur info petak dan <i>upgrade</i> .
3.	BLP	14 November 2015 – 1 jam	Memperbaiki fitur <i>load</i> dan <i>save</i> serta menyelesaikan fitur <i>end turn</i> .
4.	DU	15 November 2015 – 2 jam	Menyelesaikan fitur <i>sell</i> dan <i>sell offered</i> .
5.	AAS	16 November 2015 – 2 jam	Menyelesaikan fitur <i>random chance</i> , <i>get bonus</i> , dan <i>buy</i> .
6.	BLP	18 November 2015 – 2 jam	Menyelesaikan fitur <i>free me</i> , <i>free tax</i> dan <i>travel</i> .
7.	BLP	20 November 2015 – 2 jam	Menyelesaikan fitur <i>roll dice</i> , menambahkan beberapa <i>autocommand</i> .
8.	AAS	20 November 2015 – 2jam	Menyelesaikan fitur <i>show leaderboard</i> , <i>masuk penjara</i> , .
9.	DU	22 November 2015 – 1 jam	Menyelesaikan fitur <i>bayar pajak</i> dan <i>host</i> .
10.	BLP	23 November 2015 – 2 jam	Menyelesaikan fitur <i>protect</i> , <i>show board</i> , dan <i>show offered</i> .
11.	BLP	24 November 2015 – 3 jam	Mengintegrasikan semua prosedur menjadi satu, menyelesaikan antarmuka permainan dan mempersiapkan untuk dipergunakan.
12.	BLP	25 November 2015 – 2 jam	Mengisi bagian 1, 2, 3 dan 4 pada laporan.

13.	AAS	26 November 2015 – 2 jam	Debug program fitur <i>host</i> , keadaan masuk dan keluar penjara, dan <i>buy offered</i> .
14.	BLP	27 November 2015 – 1,5 jam	Menambahkan isi dalam laporan, membenahi beberapa definisi harga dalam program.
15.	FYP, DU	28 November 2015 – 1 jam	Memperbaiki fitur buy.
16.	BLP	28 November 2015 – 1 jam	Mengisi semua bagian laporan.
17.	AAS	29 November 2015 – 2 jam	Debug program final
18.	DU	29 November 2015 – 2jam	Debug program final

Keterangan :

BLP : Bervianto Leo P

AAS : Muhammas Az-zahid A. S

DU : Diastuti Utami

FYP : Fanda Yuliana Putri

FK : Muhammad Farhan Kemal