

Kata Pengantar

Puji dan syukur yang tak terhingga saya panjatkan ke hadirat Allah SWT, yang telah memberikan kemudahan dan kesempatan kepada saya untuk dapat menyelesaikan proyek aplikasi game Ulo Asem. Saya merasa sangat bersyukur karena telah diberikan kemampuan dan kekuatan untuk menyelesaikan proyek ini tepat pada waktunya.

Tidak lupa saya sampaikan terima kasih yang tak terhingga kepada Bapak Ali Tarmuji, S.T., M.Cs. Teknik Informatika, selaku dosen pembimbing yang telah memberikan bimbingan dan arahan yang sangat membantu saya dalam mengerjakan pembuatan aplikasi ini. Saya merasa sangat beruntung telah diberikan kesempatan untuk belajar dan mengikuti bimbingan dari Bapak.

Terimakasih juga kepada teman-teman saya yang mensupport saya dalam pembuatan aplikasi ini, serta tidak lupa juga kepada orang tua saya yang selalu memberikan dukungan dan semangat kepada saya sehingga saya dapat menyelesaikan tugas ini.

Meskipun saya berusaha semaksimal mungkin dalam menyusun portofolio ini, saya sadar bahwa masih banyak kekurangan dan kelemahan yang terdapat pada aplikasi ini. Oleh karena itu, saya mengajak pembaca untuk memberikan kritik dan saran yang membangun kepada saya, agar aplikasi yang akan datang dapat menjadi lebih baik lagi. Kritik konstruktif sangat saya harapkan demi kesempurnaan aplikasi selanjutnya.

Akhir kata, saya berharap portofolio ini dapat memberikan gambaran yang baik tentang aplikasi game Ulo Asem yang saya buat, dan dapat menjadi referkensi yang bermanfaat bagi pembaca. Saya juga berharap agar pembuatan aplikasi game dapat berjalan dengan lebih lancar dan sukses di masa yang akan datang.

Yogyakarta, Januari 2023

Penyusun

Mohammad Farid Hendianto

**Surat Pengesahan**

Yogyakarta, 25 Januari 2023

**Nomor :** 01/SP/Universitas Ahmad Dahlan/FTI/Jurusan Informatika/2022

Kepada Yth,

Dekan Fakultas Teknologi Industri

Universitas Ahmad Dahlan

Dengan Hormat,

Saya yang bertanda tangan di bawah ini :

**Nama :** Mohammad Farid Hendianto

**NIM :** 2200018401

**Jurusan :** Informatika

**Fakultas :** Fakultas Teknologi Industri

**Universitas :** Universitas Ahmad Dahlan

Dengan ini menyatakan bahwa aplikasi yang berjudul "Ulo Asem" merupakan hasil karya saya sendiri dan tidak melanggar hak cipta atau hak kekayaan intelektual milik pihak lain. Aplikasi ini dibuat untuk memenuhi tugas dasar sistem komputer dan sebagai portofolio aplikasi saya.

Demikian surat pengesahan ini dibuat dengan sebenarnya dan atas perhatiannya saya ucapkan terima kasih.



**Pembuat Aplikasi.**

Mohammad Farid Hendianto

Dosen Pembimbing

Ali Tarmuji, S.T., M.Cs

**Mengetahui.**

Dekan Fakultas Teknologi Industri

Universitas Ahmad Dahlan

Sunardi, S.T., M.T., Ph.D.

**DAFTAR ISI**

[**DAFTAR ISI** 4](#_Toc125670901)

[**A.** **Deskripsi dan Ruang Lingkup Proyek** 5](#_Toc125670902)

[**1)** **Deskripsi Proyek** 5](#_Toc125670903)

[**2)** **Ruang Lingkup Proyek** 8](#_Toc125670904)

[**B.** **Daftar seluruh spesifikasi aplikasi** 9](#_Toc125670905)

[**C.** **Rancangan Antarmuka** 11](#_Toc125670906)

[**D.** **Diagram proses aplikasi** 15](#_Toc125670907)

[**E.** **Penjelasan koding/ skrip program** 19](#_Toc125670908)

[**1)** **Inisialisasi Variabel** 19](#_Toc125670909)

[**2)** **Prosedur Proses Utama** 25](#_Toc125670910)

[**F.** **Tampilan yang dihasilkan Aplikasi** 49](#_Toc125670911)

[**H.** **Analisis Pengerjaan Projek** 55](#_Toc125670912)

[**1)** **Tinjauan dari Sisi Waktu** 55](#_Toc125670913)

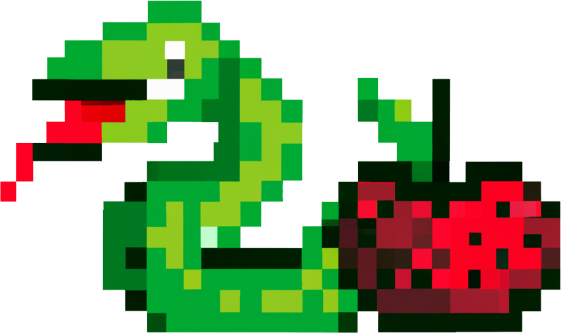
[**2)** **Ketercapaian Spesifikasi** 57](#_Toc125670914)

[**3)** **Biaya yang dibutuhkan** 58](#_Toc125670915)

[**4)** **Kendala yang dihadapi** 58](#_Toc125670916)

[**5)** **Tantangan Masa Depan** 59](#_Toc125670917)

1. **Deskripsi dan Ruang Lingkup Proyek**
2. **Deskripsi Proyek**



Gambar 1Logo Ulo Asem. **(Sumber: Penulis)**

Aplikasi game Ulo Asem adalah sebuah aplikasi yang dibuat menggunakan bahasa pemrograman Assembly, yaitu MASM. Aplikasi ini ditujukan untuk membantu para pemula dalam mempelajari bahasa Assembly dengan cara yang menyenangkan.

Aplikasi game Ulo Asem ini diberi nama dengan mengambil inspirasi dari bahasa Jawa yang merupakan salah satu bahasa daerah di Indonesia. Kata 'ulo' dalam bahasa Jawa merupakan salah satu sinonim untuk kata 'ular', sedangkan 'asem' merupakan kepanjangan dari bahasa Assembly yang merupakan bahasa pemrograman yang digunakan dalam pembuatan aplikasi ini. Sehingga, penamaan aplikasi game Ulo Asem ini merupakan gabungan dari kata 'ular' dan 'Assembly', yang diharapkan dapat menjadi salah satu sarana belajar bahasa Assembly yang menyenangkan bagi para pemula.

Dalam aplikasi game Ulo Asem ini, pengguna dapat memainkan game tersebut dengan menggunakan tombol panah pada keyboard untuk mengatur arah gerakan ular. Ular tersebut akan bergerak mengikuti arah yang ditunjukkan oleh pengguna, dan akan terus bergerak hingga menabrak tembok atau tubuh sendiri. Selain itu, setiap kali ular berhasil memakan sebuah apel, maka skor akan bertambah dan panjang ular akan bertambah.

Aplikasi ini dilengkapi dengan fitur-fitur seperti tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Selain itu, saat game berakhir, aplikasi game Ulo Asem akan menampilkan skor akhir yang telah diraih oleh pengguna. Skor tersebut akan ditampilkan sesuai dengan jumlah apel yang telah dimakan oleh ular. Selain itu, aplikasi game ini juga menyediakan fitur reset yang dapat digunakan oleh pengguna untuk mengulang permainan dari awal.

Dalam proyek pembuatan aplikasi ini, saya telah berusaha untuk menyertakan dokumentasi yang lengkap dan jelas sebagai referensi bagi pengguna aplikasi ini. Dokumentasi tersebut berisi informasi mengenai cara penggunaan aplikasi, kode program yang digunakan, serta gambaran dari alur proses pembuatan aplikasi. Hal ini dilakukan agar pengguna dapat dengan mudah memahami dan menggunakan aplikasi ini dengan baik. Selain itu, dokumentasi ini juga dapat digunakan sebagai bahan referensi bagi para mahasiswa/i Fakultas Teknologi Industri Universitas Ahmad Dahlan Jurusan Informatika yang ingin mempelajari bahasa Assembly melalui aplikasi game Ulo Asem ini.

Dokumentasi tersebut juga mencakup informasi mengenai arsitektur aplikasi, serta detail tentang masing-masing fitur yang terdapat pada aplikasi ini. Seperti fitur tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Selain itu, dokumentasi ini juga menjelaskan proses pembuatan aplikasi ini, mulai dari perancangan, implementasi, hingga pengujian aplikasi.

Selain itu, dalam proyek pembuatan aplikasi ini, saya juga berusaha untuk menyertakan dokumentasi yang lengkap dan jelas sebagai referensi bagi pengguna aplikasi ini. Dokumentasi tersebut berisi informasi mengenai cara penggunaan aplikasi, kode program yang digunakan, serta gambaran dari algoritma yang digunakan dalam pembuatan aplikasi ini.

Aplikasi ini dikembangkan dengan menggunakan bahasa pemrograman Assembly yaitu MASM. Aplikasi ini ditujukan untuk membantu para pemula dalam mempelajari bahasa Assembly dengan cara yang menyenangkan. Dalam aplikasi game Ulo Asem ini, pengguna dapat memainkan game tersebut dengan menggunakan tombol panah pada keyboard untuk mengatur arah gerakan ular. Ular tersebut akan bergerak mengikuti arah yang ditunjukkan oleh pengguna, dan akan terus bergerak hingga menabrak tembok atau tubuh sendiri. Selain itu, setiap kali ular berhasil memakan sebuah apel, maka skor akan bertambah dan panjang ular akan bertambah.

Aplikasi ini juga dilengkapi dengan beberapa fitur yang dapat membantu dalam proses belajar Assembly. Fitur-fitur tersebut seperti tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Selain itu, saat game berakhir, aplikasi game Ulo Asem akan menampilkan skor akhir yang telah diraih oleh pengguna. Skor tersebut akan ditampilkan sesuai dengan jumlah apel yang telah dimakan oleh ular. Selain itu, aplikasi game ini juga menyediakan fitur reset yang dapat digunakan oleh pengguna untuk mengulang permainan dari awal.

Dalam proyek pembuatan aplikasi ini, saya telah berusaha sebaik mungkin untuk menyelesaikan proyek ini sesuai dengan tujuan dan ruang lingkup yang telah ditentukan. Proses pembuatan aplikasi ini meliputi tahap-tahap seperti perencanaan, pengembangan, pengujian dan evaluasi. Pada tahap perencanaan, saya melakukan riset dan analisis terhadap game-game serupa yang sudah ada sebagai acuan dalam pembuatan aplikasi ini. Pada tahap pengembangan, saya mulai menulis kode program dengan menggunakan bahasa Assembly dan menggunakan software MASM sebagai compiler. Pada tahap pengujian, saya melakukan uji coba terhadap aplikasi yang telah dibuat untuk mengecek apakah aplikasi tersebut dapat berjalan dengan baik dan sesuai dengan yang diharapkan. Pada tahap evaluasi, saya melakukan evaluasi terhadap aplikasi yang telah dibuat dan melakukan perbaikan-perbaikan yang diperlukan sesuai dengan hasil pengujian.

Dalam proyek pembuatan aplikasi ini, saya juga berusaha untuk menyertakan fitur-fitur yang bermanfaat bagi pengguna. Salah satu fitur yang disertakan adalah pemilihan kecepatan Ulo. Pengguna dapat memilih kecepatan gerakan ular sesuai dengan preferensi mereka. Fitur ini dapat membantu pengguna dalam menyesuaikan tingkat kesulitan game sesuai dengan kemampuan mereka. Selain itu, saya juga menyertakan fitur pembuatan acak apel. Fitur ini dapat membuat permainan lebih menantang karena apel yang ditampilkan tidak selalu muncul di tempat yang sama.

Selain itu, saya juga menyertakan fitur tampilan dinding. Fitur ini dapat membuat permainan lebih menarik karena dinding yang ditampilkan dapat diubah sesuai dengan preferensi pengguna. Pengguna dapat memilih dari berbagai macam warna dinding yang tersedia. Selain itu, saya juga menyertakan fitur papan skor. Fitur ini dapat membuat pengguna dapat melihat skor yang telah diraih serta high score yang pernah dicapai.

Dalam proyek pembuatan aplikasi ini, saya juga berusaha untuk menyertakan dokumentasi yang lengkap dan jelas sebagai referensi bagi pengguna aplikasi ini. Dokumentasi tersebut berisi informasi mengenai cara penggunaan aplikasi, kode program yang digunakan, serta gambaran dari algoritma yang digunakan dalam pembuatan aplikasi. Dokumentasi ini diharapkan dapat membantu pengguna dalam mengerti cara kerja aplikasi serta membantu dalam proses troubleshooting jika terjadi masalah.

Secara keseluruhan, aplikasi game Ulo Asem ini diharapkan dapat menjadi media belajar yang menyenangkan bagi para pemula yang ingin mempelajari bahasa Assembly. Aplikasi ini diharapkan dapat membantu para pemula dalam memahami cara kerja bahasa Assembly serta dapat meningkatkan kemampuan logika dan problem solving mereka. Saya berharap aplikasi ini dapat diterima dengan baik oleh pengguna dan dapat bermanfaat bagi mereka.

1. **Ruang Lingkup Proyek**

Ruang lingkup proyek ini adalah pembuatan sebuah aplikasi game Ulo Asem yang ditujukan untuk semua pengguna yang ingin bermain game tersebut. Aplikasi ini dapat dijalankan di sistem operasi Windows dengan menggunakan bahasa Assembly. Tujuan dari pembuatan aplikasi ini adalah untuk menghasilkan sebuah game yang dapat menghibur pengguna serta dapat digunakan sebagai media belajar Assembly bagi mahasiswa/i Fakultas Teknologi Industri Universitas Ahmad Dahlan Jurusan Informatika. Aplikasi ini dilengkapi dengan fitur-fitur seperti tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Batasan dari aplikasi ini adalah hanya dapat digunakan di sistem operasi Windows dan ditulis dengan menggunakan bahasa Assembly.

1. **Daftar seluruh spesifikasi aplikasi**

Untuk dapat menjalankan aplikasi game Ulo Asem, diperlukan spesifikasi sistem yang minimal sebagai berikut:

|  |  |  |
| --- | --- | --- |
| **Sistem Operasi** | : | Windows XP 32 bit atau yang lebih tinggi |
| **Prosessor** | : | 512 Mhz atau lebih tinggi |
| **Memory** | : | 32 MB atau lebih tinggi |
| **Ukuran File** | : | 39 KB |
| **Storage** | : | 1 MB ruang hardisk yang tersedia |
| **Sound card** | : | Terdapat soundcard untuk menjalankan audio |
| **Graphics** | : | Graphic card yang mendukung penampilan warna |

Pastikan bahwa semua perangkat lunak yang diperlukan, seperti sistem operasi dan aplikasi pendukung, telah terinstal dengan benar pada komputer Anda agar game Ulo Asem dapat berjalan dengan lancar. Jika ada perangkat lunak yang kurang atau tidak terinstal dengan benar, maka kemungkinan game Ulo Asem akan mengalami error atau tidak dapat dijalankan sama sekali. Oleh karena itu, pastikan untuk melakukan instalasi semua perangkat lunak yang diperlukan dengan benar agar game dapat berjalan dengan lancar.

Selain spesifikasi sistem yang diperlukan untuk menjalankan aplikasi game Ulo Asem, aplikasi ini juga memiliki beberapa spesifikasi aplikasi yang perlu diperhatikan.

* Aplikasi ini ditulis menggunakan bahasa pemrograman Assembly yaitu MASM.
* Aplikasi ini hanya dapat digunakan pada sistem operasi Windows.
* Aplikasi ini dilengkapi dengan fitur-fitur seperti tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard.
* Aplikasi ini dapat digunakan untuk menghibur pengguna serta dapat digunakan sebagai media belajar Assembly bagi mahasiswa/i Fakultas Teknologi Industri Universitas Ahmad Dahlan Jurusan Informatika.
* Aplikasi ini memiliki batasan hanya dapat digunakan di sistem operasi Windows dan ditulis dengan menggunakan bahasa Assembly.

Semua spesifikasi aplikasi ini harus diperhatikan agar aplikasi dapat berjalan dengan baik dan sesuai dengan tujuan pembuatannya. Jika ada spesifikasi yang tidak sesuai, maka aplikasi mungkin tidak dapat dijalankan atau mengalami error. Namun, selalu ada kemungkinan untuk dapat meningkatkan atau menambah fitur-fitur yang dapat membuat pengalaman bermain game Ulo Asem semakin menyenangkan bagi pengguna. Selain itu, jika terdapat bug atau masalah dalam aplikasi, saya akan segera melakukan perbaikan untuk mengatasinya secepat mungkin agar aplikasi dapat digunakan dengan baik oleh pengguna.

Selain spesifikasi sistem yang diperlukan, aplikasi game Ulo Asem juga dilengkapi dengan beberapa fitur yang akan membuat pengalaman bermain game semakin menyenangkan bagi pengguna. Fitur-fitur tersebut antara lain:

* Tampilan dinding: Aplikasi ini menyediakan tampilan dinding yang akan membuat permainan semakin menyenangkan bagi pengguna.
* Papan skor: Fitur papan skor pada aplikasi game Ulo Asem menampilkan skor secara real-time dari skor yang telah didapatkan saat dimainkan. Skor tersebut ditampilkan dalam bentuk angka yang mudah dibaca dan dipahami oleh pengguna. Selain itu, fitur ini juga menampilkan skor tertinggi dari permainan sebelum-sebelumnya. Skor tertinggi ini diperoleh dari hasil permainan saat program dijalankan dan ditampilkan sebagai acuan bagi pengguna untuk dapat meningkatkan skor yang telah didapatkan. (Skor tertinggi yang ditampilkan hanya berlaku untuk permainan saat program) dijalankan.Pemilihan kecepatan Ulo: Pengguna dapat memilih kecepatan gerakan ular sesuai dengan keinginan mereka.
* Pembuatan acak apel: Aplikasi ini menyediakan pembuatan acak apel yang akan membuat permain
* Fitur reset: Untuk mengulang permainan dari awal

Dalam proyek pembuatan aplikasi ini, saya telah berusaha untuk menyertakan spesifikasi aplikasi yang sesuai dengan studi kasus yang ada. Namun, selalu ada kemungkinan untuk dapat meningkatkan atau menambah fitur-fitur yang ada pada aplikasi ini, tergantung dari hasil evaluasi dan umpan balik dari pengguna. Selain spesifikasi sistem yang diperlukan, aplikasi ini juga menyertakan fitur-fitur seperti tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Selain itu, aplikasi ini juga dilengkapi dengan fitur reset yang dapat digunakan oleh pengguna untuk mengulang permainan dari awal.

Dengan spesifikasi dan fitur yang disertakan dalam aplikasi ini, diharapkan dapat memberikan pengalaman yang menyenangkan bagi pengguna serta dapat digunakan sebagai media belajar bahasa Assembly bagi mahasiswa/i Fakultas Teknologi Industri Universitas Ahmad Dahlan Jurusan Informatika.

1. **Rancangan Antarmuka**

Dalam rancangan antarmuka aplikasi Ulo Asem, terdapat 4 tampilan utama yang akan ditampilkan kepada pengguna.

Gambar 2 Tampilan Awal Aplikasi Ulo Asem **(Sumber: Penulis)**



Logo Game

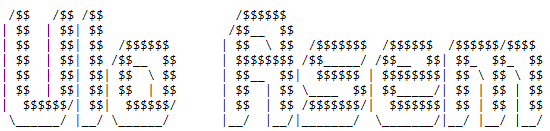
Nama dan NIM

Versi dan tahun pembuatan

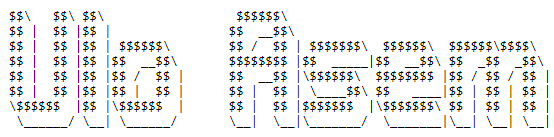
Nama dan NIM

Tampilan pertama adalah tampilan awal yang menampilkan logo game Ulo Asem dengan menggunakan ASCII Art. Logo Ulo Asem dalam berbentuk tulisan ini akan ditampilkan dalam bentuk animasi. Selain itu, tampilan ini juga akan menampilkan informasi mengenai versi dan tahun pembuatan program Ulo Asem serta nama dan NIM pembuat aplikasi.

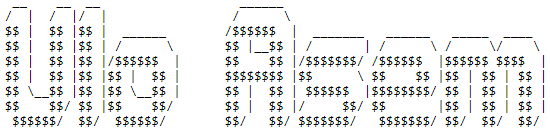
Berikut adalah rancangan tulisan Logo Ulo Asem yang ingin dianimasikan



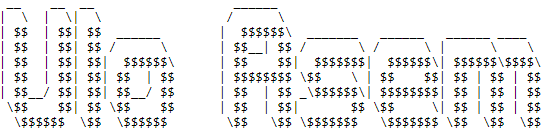
Gambar 3 Frame 1 logo Ulo Asem **(Sumber: Penulis)**



Gambar 4 Frame 2 logo Ulo Asem **(Sumber: Penulis)**



Gambar 5 Frame 3 logo Ulo Asem **(Sumber: Penulis)**



Gambar 6 Frame 3 logo Ulo Asem **(Sumber: Penulis)**

Gambar 7 Tampilan Game Aplikasi Ulo Asem **(Sumber: Penulis)**



Papan Skor

Papan Kecepatan Ulo

Apel

Papan Permainan

Ulo

Tampilan kedua adalah tampilan game, yang merupakan inti dari aplikasi ini. Tampilan ini akan menampilkan papan skor yang berisi skor yang sedang berlangsung di permainan, serta skor tertinggi dari permainan sebelumnya. Selain itu, tampilan ini juga akan menampilkan papan kecepatan ulo yang menunjukkan kecepatan ulo saat ini, serta papan permainan yang berisikan dinding, ulo, dan apel yang ditampilkan secara acak di dalam dinding permainan.



Papan Skor yang sudah didapatkan

Gambar Tengkorak dan pesan Game Over

Papan mengecek ingin bermain atau tidak

Gambar 8 Tampilan Game Over Aplikasi Ulo Asem **(Sumber: Penulis)**

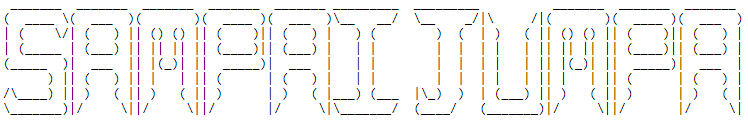
Tampilan ketiga adalah tampilan GameOver, yang akan ditampilkan ketika permainan berakhir. Tampilan ini akan menampilkan pesan GameOver dengan tulisan dan ASCII Art, serta menampilkan skor yang didapatkan selama permainan. Selain itu, tampilan ini juga akan menampilkan skor tertinggi yang pernah didapatkan selama permainan. Dengan tampilan antarmuka yang sederhana dan intuitif, pengguna akan dengan mudah mengoperasikan aplikasi Ulo Asem dan memahami cara bermain game.



Gambar 9 Tampilan Keluar Aplikasi Ulo Asem **(Sumber: Penulis)**

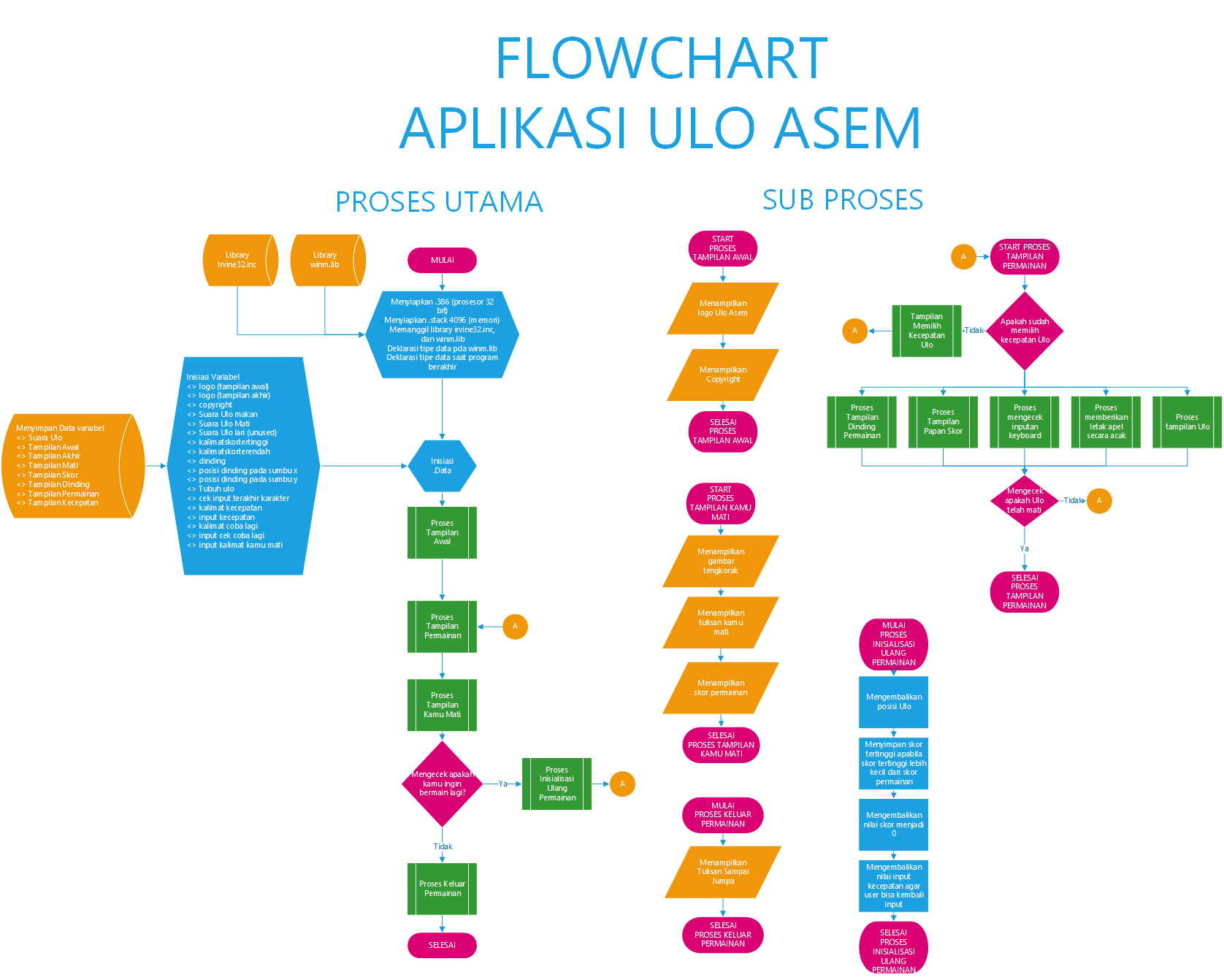
Tampilan terakhir adalah Tampilan Keluar adalah tampilan yang ditampilkan ketika aplikasi Ulo Asem ditutup. Di tampilan ini, terdapat pesan yang ditampilkan dengan ASCII Art bertuliskan "SAMPAI JUMPA" yang menandakan bahwa pengguna telah selesai menggunakan aplikasi.

Berikut adalah rancangan tulisan SAMPAI JUMPA



Gambar 10Rancangan tulisan "SAMPAI JUMPA" dengan ASCII ART

1. **Diagram proses aplikasi**



Gambar 11 Flowchart Ulo Asem **(Sumber: Penulis)**

Berikut adalah alur program aplikasi Ulo Asem

1. Memuat dan menampilkan interface permainan

* Menggunakan library Irvine32 untuk memanggil Prosedur clrscr yang akan membersihkan layar
* Memanggil Prosedur TampilanDinding untuk menggambar tembok pada layar permainan
* Memanggil Prosedur TampilanPapanSkor untuk menampilkan papan skor pada layar permainan
* Memanggil Prosedur MemilihKecepatanUlo untuk memberikan pemain pilihan kecepatan ulo
* Memulai loop yang akan menampilkan ulo dengan panjang awal 5 segmen dengan Prosedur TampilanPemain
* Memanggil Prosedur Randomize dari library Irvine32 untuk mengacak posisi apel yang akan muncul pada layar
* Memanggil Prosedur BuatAcakApel untuk menentukan posisi apel yang akan muncul
* Memanggil Prosedur TampilanApel untuk menampilkan apel pada layar permainan.
* Loop akan terus berjalan dan menunggu input dari pemain sampai pemain memutuskan untuk keluar dari permainan dengan menekan tombol x atau sampai pemain mati.

1. Meminta input kecepatan Ulo dari pemain

Pada bagian ini, aplikasi akan meminta pemain untuk memasukkan kecepatan Ulo yang diinginkan pada awal permainan. Input yang dimasukkan akan digunakan sebagai parameter untuk menentukan seberapa cepat gerakan Ulo pada permainan. Setelah pemain memasukkan kecepatan yang diinginkan, aplikasi akan melanjutkan ke langkah berikutnya.

1. Menampilkan Ulo dengan panjang 5 segmen

Pada tahap ini, Ulo akan ditampilkan di layar dengan panjang 5 segmen. Setiap segmen ditampilkan dengan menggunakan koordinat x dan y yang disimpan dalam array xPosisi dan yPosisi. Posisi awal Ulo ditentukan secara acak di layar.

1. Menampilkan Apel secara acak di layar

Pada poin ini, aplikasi akan menampilkan apel secara acak di layar. Apel akan muncul di posisi yang tidak sama dengan posisi Ulo. Proses ini akan terus berulang setiap kali apel dimakan oleh Ulo.

1. Meminta input gerakan dari pemain

Pada poin ini, program akan meminta input dari pemain tentang arah gerakan Ulo. Input yang diperbolehkan adalah arah atas, bawah, kiri, dan kanan. Setelah input diterima, program akan memperbarui posisi Ulo sesuai dengan arah yang dipilih pemain.

1. Memeriksa apakah input yang dimasukkan valid atau tidak

Dalam tahap ini, program akan memeriksa apakah input gerakan yang dimasukkan pemain merupakan salah satu dari empat arah yang tersedia, yaitu ke atas, ke bawah, ke kiri, atau ke kanan. Jika input yang dimasukkan valid, maka program akan melanjutkan ke tahap selanjutnya. Namun jika input yang dimasukkan tidak valid, program akan mengabaikan input tersebut dan terus menunggu input yang valid dari pemain.

1. Menggerakkan Ulo sesuai dengan input yang dimasukkan oleh pemain

Dalam tahap ini, program akan memeriksa apakah input yang diberikan pemain valid atau tidak. Jika input yang diberikan pemain valid, maka Ulo akan bergerak sesuai dengan arah yang diberikan pemain. Jika input yang diberikan pemain tidak valid, maka Ulo akan terus bergerak ke arah yang sama seperti sebelumnya. Misalnya, jika pemain memberikan input untuk bergerak ke kiri, tetapi Ulo sedang berada di ujung kanan layar, maka Ulo tidak akan bergerak ke kiri dan akan terus bergerak ke arah yang sama seperti sebelumnya.

1. Memeriksa apakah Ulo telah memakan Apel

Untuk tahap ini, program akan memeriksa apakah koordinat Ulo sama dengan koordinat Apel. Jika sama, maka Ulo akan memakan Apel dan panjang Ulo akan bertambah 1 segmen. Sebaliknya, jika tidak sama, maka Ulo tidak akan memakan Apel dan tidak ada perubahan pada panjang Ulo.

1. Jika Ulo telah memakan Apel, maka tambahkan 1 segmen pada Ulo dan menampilkan Apel baru secara acak

Tahap ini merupakan bagian dari permainan Ulo Asem di mana setelah Ulo berhasil memakan Apel, maka panjang Ulo akan ditambah sebanyak 1 segmen. Selain itu, Apel yang telah dimakan juga akan dihapus dan digantikan dengan Apel baru yang akan ditampilkan secara acak di layar permainan. Ini bertujuan untuk menjaga keseruan permainan, sehingga pemain harus terus-menerus mengontrol gerakan Ulo untuk memakan Apel yang muncul di layar.

1. Memeriksa apakah Ulo telah menabrak tembok atau menabrak dirinya sendiri

Tahap ini adalah tahap yang mengecek apakah Ulo telah menabrak tembok atau menabrak dirinya sendiri. Proses ini dilakukan dengan memeriksa posisi Ulo apakah sudah melewati batas layar atau menyentuh salah satu segmen tubuhnya sendiri. Jika kondisi tersebut terpenuhi, maka permainan akan dihentikan dan Ulo dinyatakan mati.

1. Jika Ulo telah menabrak tembok atau menabrak dirinya sendiri, maka permainan berakhir dan menampilkan skor akhir

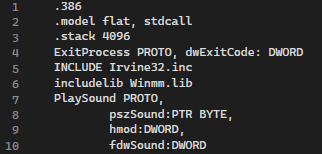
Apabila Ulo telah menabrak tembok atau menabrak dirinya sendiri, maka permainan berakhir dan menampilkan skor akhir. Tahap ini bertujuan untuk menghentikan permainan jika kondisi tersebut terjadi. Skor akhir akan ditampilkan pada layar sebagai hasil dari permainan yang telah dimainkan oleh pemain.

1. Meminta input dari pemain untuk memainkan lagi atau keluar dari permainan

Jika pemain memilih untuk memainkan lagi, maka ulangi tahap 2 sampai 11. Jika pemain memilih untuk keluar dari permainan, maka program akan selesai dan menampilkan pesan "Sampai Jumpa".

1. **Penjelasan koding/ skrip program**
2. **Inisialisasi Variabel**

Inisialisasi variabel dalam kode aplikasi Ulo Asem adalah proses pembuatan dan pemberian nilai awal pada variabel yang digunakan dalam aplikasi. Variabel yang diinisialisasi dapat berupa tipe data seperti integer, string, atau boolean. Pada kode aplikasi Ulo Asem, variabel yang diinisialisasi dapat digunakan untuk menyimpan data seperti input dari pengguna, hasil perhitungan, atau status dari aplikasi. Inisialisasi variabel juga dapat dilakukan pada saat aplikasi pertama kali dijalankan untuk mengatur kondisi awal dari aplikasi.



Gambar 12 Cuplikan kodingan ke-1 Aplikasi Ulo Asem **(Sumber: Penulis)**

Ini adalah cuplikan koding dari bagian awal dari program Ulo Asem yang ditulis dalam bahasa Assembly. Pada bagian ini, terdapat beberapa perintah yang digunakan untuk mengatur model, stack, dan library yang digunakan dalam program.

Perintah ".386" digunakan untuk mengatur versi processor yang digunakan dalam program. Pada kasus ini, program ini dioptimalkan untuk digunakan pada processor dengan arsitektur 386 atau lebih tinggi.

Perintah ".model flat, stdcall" digunakan untuk mengatur model memory yang digunakan dalam program. "Flat" mengindikasikan bahwa program akan menggunakan memory model yang tidak terbagi-bagi, sedangkan "stdcall" mengindikasikan bahwa program akan menggunakan standar pemanggilan prosedur.

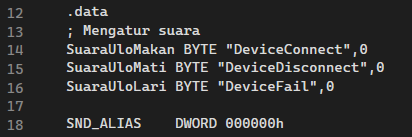
Perintah ".stack 4096" digunakan untuk mengatur ukuran stack yang digunakan dalam program. Dalam hal ini, ukuran stack yang digunakan adalah 4096 byte.

Perintah "ExitProcess PROTO, dwExitCode: DWORD" digunakan untuk mendeklarasikan prototipe fungsi ExitProcess yang digunakan untuk keluar dari program. Parameter "dwExitCode" digunakan untuk menentukan kode keluar yang akan dikembalikan saat program keluar.

Perintah "INCLUDE Irvine32.inc" digunakan untuk memasukkan library Irvine32 yang berisi berbagai fungsi yang dapat digunakan dalam pemrograman assembly. Library ini dikembangkan oleh Dr. Kenneth L. Irvine dan menyediakan berbagai fungsi yang berguna untuk pemrograman assembly, seperti fungsi untuk menangani input-output, operasi matematika, dan lain-lain.

Perintah "includelib Winmm.lib" digunakan untuk memasukkan library Windows Multimedia (Winmm.lib) yang digunakan untuk mengelola audio dan musik. Dalam koding di atas, library ini digunakan untuk mengelola suara yang digunakan dalam aplikasi Ulo Asem.

Perintah "PlaySound PROTO, pszSound:PTR BYTE, hmod:DWORD, fdwSound:DWORD" digunakan untuk mendeklarasikan prototipe dari fungsi PlaySound. Fungsi ini digunakan untuk memutar suara yang ditentukan oleh parameter pszSound. Parameter hmod dan fdwSound digunakan untuk menentukan modul yang digunakan dan flag yang digunakan untuk mengontrol cara suara diputar.

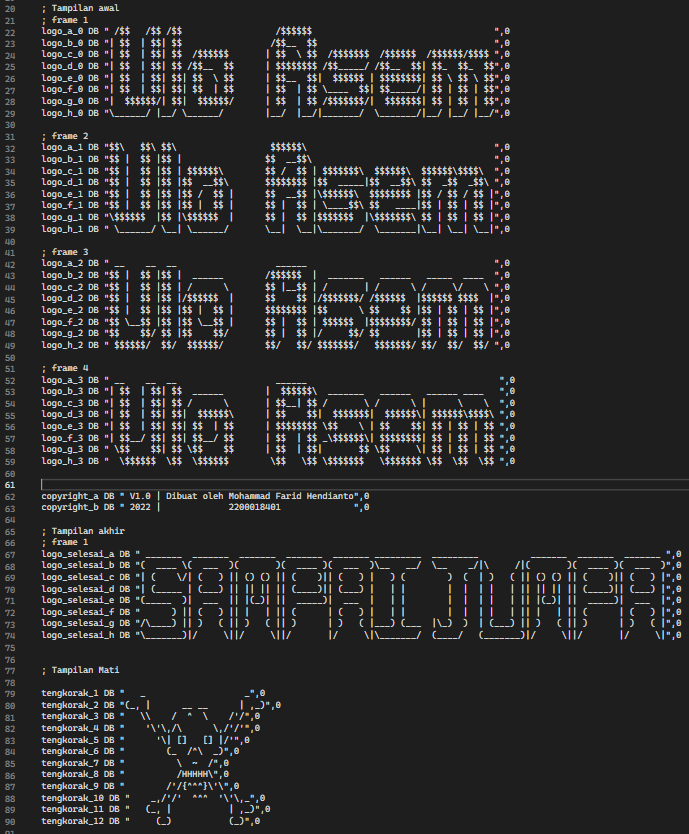


Gambar 13 Cuplikan kodingan ke-2 Aplikasi Ulo Asem **(Sumber: Penulis)**

Perintah "SuaraUloMakan BYTE "DeviceConnect",0" digunakan untuk mendefinisikan suara yang akan diputar ketika ular berhasil memakan apel dalam permainan. Perintah "SuaraUloMati BYTE "DeviceDisconnect",0" digunakan untuk mendefinisikan suara yang akan diputar ketika ular mati dalam permainan. Perintah "SuaraUloLari BYTE "DeviceFail",0" digunakan untuk mendefinisikan suara yang akan diputar ketika ular bergerak dalam permainan. Perintah "SND\_ALIAS DWORD 000000h" digunakan untuk mengatur suara yang akan diputar menjadi suara yang telah didefinisikan sebelumnya.

Perintah "INCLUDE Irvine32.inc" digunakan untuk memasukkan library Irvine32 yang berisi berbagai fungsi yang dapat digunakan dalam program Assembly. Perintah "includelib Winmm.lib" digunakan untuk memasukkan library Winmm yang berisi fungsi-fungsi multimedia yang dapat digunakan dalam program Assembly, seperti fungsi untuk memutar suara. Perintah "PlaySound PROTO," digunakan untuk mendefinisikan prototipe dari fungsi PlaySound yang akan digunakan dalam program untuk memutar suara.

Perintah "includelib Winmm.lib" digunakan untuk memasukkan library Winmm yang berisi fungsi-fungsi multimedia seperti pemutaran suara. Fungsi PlaySound digunakan untuk memutar suara sesuai dengan event yang terjadi dalam permainan, seperti Ulo makan, Ulo mati, dan Ulo bergerak. Variabel SuaraUloMakan, SuaraUloMati, dan SuaraUloLari digunakan untuk menyimpan nama suara yang akan dimainkan. Variabel SND\_ALIAS digunakan untuk menentukan tipe suara yang akan dimainkan, yaitu suara sistem.



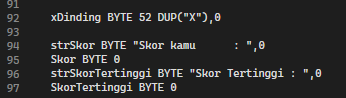
Gambar 14 Cuplikan kodingan ke-3 Aplikasi Ulo Asem **(Sumber: Penulis)**

Bagian koding yang ditunjukkan di sini adalah bagian yang menangani tampilan logo dari aplikasi Ulo Asem. Ditunjukkan dengan variabel yang diawali dengan "logo\_", yang masing-masing menyimpan sebuah string yang ditampilkan sebagai ASCII art. Setiap variabel tersebut menyimpan string yang digunakan untuk menampilkan sebuah frame dari logo Ulo Asem. Di sini, terdapat 4 frame yang digunakan dalam menampilkan logo Ulo Asem, yang ditampilkan secara bergantian untuk memberikan efek animasi pada logo tersebut.

Bagian tersebut menampilkan juga variabel untuk menyimpan string copyright dari aplikasi yang dibuat. Pada bagian ini, ditampilkan informasi bahwa aplikasi ini adalah versi 1.0 dan dibuat oleh seseorang dengan nama Mohammad Farid Hendianto, serta tahun pembuatan aplikasi yaitu 2022 dan nomor identitas pembuatnya yaitu 2200018401.

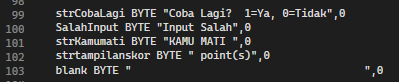
Selain terdapat ascii art yang digunakan untuk menampilkan logo game Ulo Asem pada tampilan awal dan copyright, terdapat juga tampilan untuk game over, dan tampilan akhir. Ascii art digunakan untuk menampilkan gambar dengan menggunakan karakter-karakter dari tabel ASCII. Di sini, telah didefinisikan beberapa variabel yang berisi baris-baris dari karakter ASCII yang digunakan untuk menampilkan logo Ulo Asem, pesan "Game Over", dan pesan "Sampai Jumpa" pada tampilan akhir. Di samping itu, juga didefinisikan ascii art dari tengkorak sebagai tampilan game over.

Tampilan game over menggunakan ASCII art yang menggambarkan sebuah tengkorak. ASCII art ini dibentuk dari beberapa baris yang ditentukan dalam variabel tengkorak\_1 sampai tengkorak\_12. Baris-baris tersebut digabungkan untuk membentuk gambar tengkorak yang akan ditampilkan saat pesan "Game Over" ditampilkan.



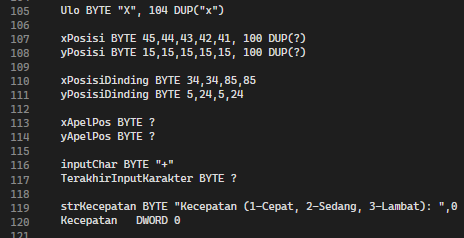
Gambar 15 Cuplikan kodingan ke-4 Aplikasi Ulo Asem **(Sumber: Penulis)**

Bagian ini adalah deklarasi variabel untuk digunakan dalam aplikasi. Variabel xDinding digunakan untuk menampilkan dinding permainan dengan karakter "X" sebanyak 52 kali. Variabel strSkor digunakan untuk menampilkan teks "Skor kamu: " saat menampilkan skor saat ini, sedangkan variabel Skor digunakan untuk menyimpan nilai skor saat ini. Variabel strSkorTertinggi digunakan untuk menampilkan teks "Skor Tertinggi: " saat menampilkan skor tertinggi, dan variabel SkorTertinggi digunakan untuk menyimpan nilai skor tertinggi.



Gambar 16 Cuplikan kodingan ke-5 Aplikasi Ulo Asem **(Sumber: Penulis)**

Pada bagian kode ini digunakan beberapa variabel untuk menyimpan string yang akan ditampilkan di aplikasi. Variabel strSkor digunakan untuk menyimpan string "Skor kamu : ", dan digunakan saat aplikasi menampilkan skor yang didapatkan saat bermain. Variabel Skor digunakan untuk menyimpan skor yang didapatkan saat bermain. Variabel strSkorTertinggi digunakan untuk menyimpan string "Skor Tertinggi : ", yang digunakan saat aplikasi menampilkan skor tertinggi yang pernah didapatkan. Variabel SkorTertinggi digunakan untuk menyimpan skor tertinggi yang pernah didapatkan. Variabel strCobaLagi digunakan untuk menyimpan string "Coba Lagi? 1=Ya, 0=Tidak", yang akan ditampilkan saat aplikasi menanyakan kepada pengguna apakah ingin mencoba lagi. Variabel SalahInput digunakan untuk menyimpan string "Input Salah", yang akan ditampilkan saat pengguna memasukkan input yang salah. Variabel strKamumati digunakan untuk menyimpan string "KAMU MATI", yang akan ditampilkan saat Ulo mati. Variabel strtampilanskor digunakan untuk menyimpan string " point(s)", yang digunakan untuk menampilkan skor pada saat gameover. Variabel ini digunakan untuk memperjelas bahwa skor yang didapat dalam permainan adalah dalam bentuk poin, dan digunakan setelah skor didapat ditampilkan. Variabel blank digunakan untuk mengatur jarak pada saat menampilkan skor dan pesan pada saat gameover. Variabel ini digunakan untuk menjaga agar tampilan tetap rapi dan teratur. Variabel blank digunakan untuk menyimpan string " ", yang digunakan untuk mengosongkan tampilan pada saat menampilkan skor dan skor tertinggi.



Gambar 17 Cuplikan kodingan ke-6 Aplikasi Ulo Asem **(Sumber: Penulis)**

Variabel "Ulo" digunakan untuk menyimpan string "X" yang digunakan sebagai tampilan dari ular. Kemudian diikuti oleh 104 karakter "x", yang digunakan untuk mengisi bagian tubuh ular yang lain.

Variabel xPosisi dan yPosisi digunakan untuk menyimpan posisi dari setiap bagian tubuh ular dalam koordinat x dan y. Kemudian diikuti oleh 100 karakter "?", yang digunakan sebagai placeholder untuk posisi tubuh ular yang akan datang.

Variabel xPosisiDinding dan yPosisiDinding digunakan untuk menyimpan posisi dinding dari permainan dalam koordinat x dan y. xPosisiDinding menyimpan posisi dinding dari AtasKiri, BawahKiri, AtasKanan, dan BawahKanan, sementara yPosisiDinding menyimpan posisi dinding dari Atas, Bawah.

Variabel xApelPos dan yApelPos digunakan untuk menyimpan posisi apel dalam koordinat x dan y.

Variabel inputChar digunakan untuk menyimpan karakter input dari user saat permainan berlangsung. Variabel ini diinisialisasi dengan karakter "+" yang digunakan sebagai tanda permainan baru.

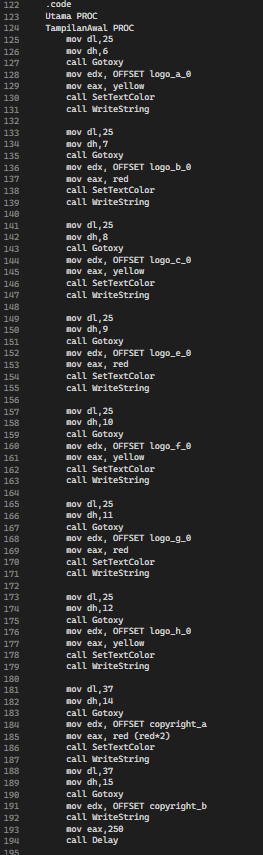
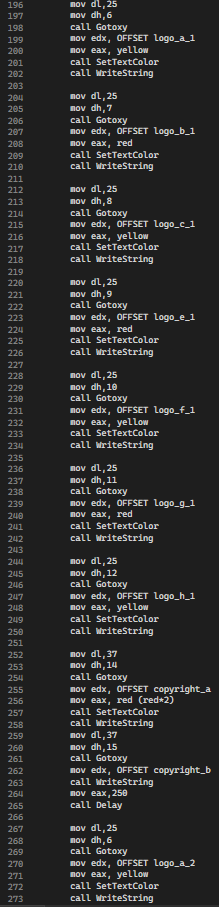
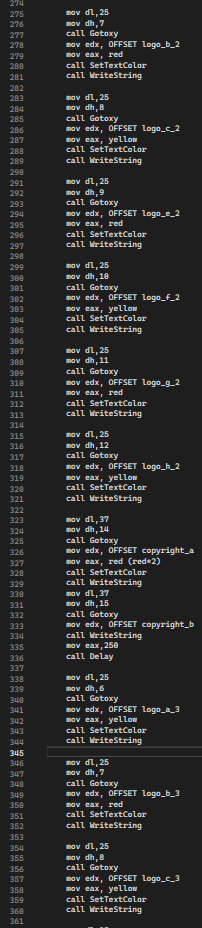
Variabel TerakhirInputKarakter digunakan untuk menyimpan karakter input terakhir dari user.

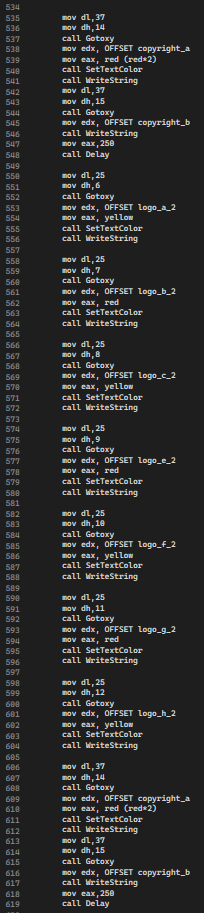
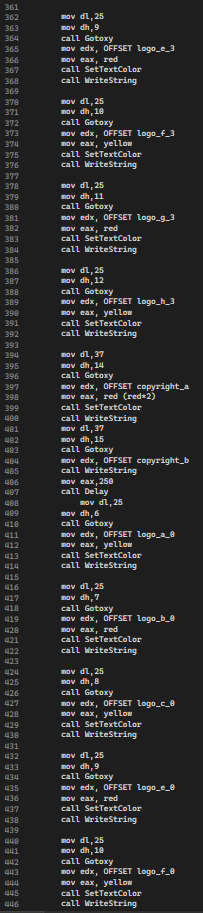
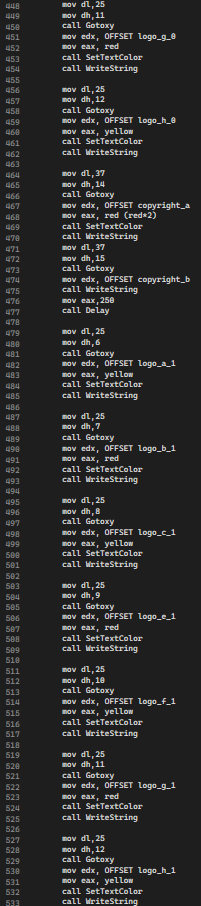
Variabel strKecepatan digunakan untuk menyimpan string "Kecepatan (1-Cepat, 2-Sedang, 3-Lambat): " yang digunakan untuk menampilkan pilihan kecepatan permainan pada saat permainan baru dimulai. Variabel Kecepatan digunakan untuk menyimpan nilai kecepatan yang dipilih oleh user.

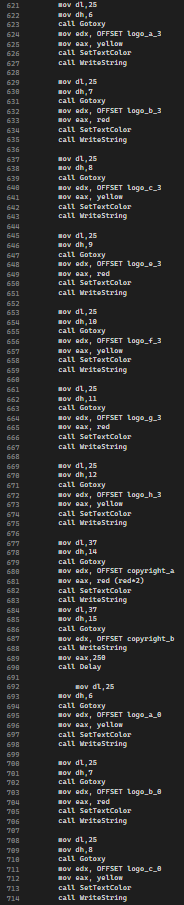
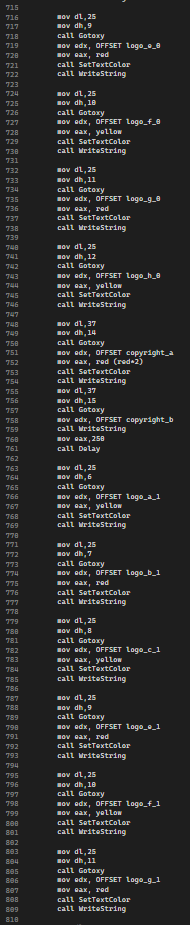
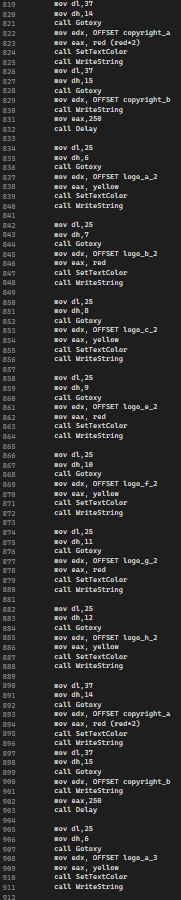
Kecepatan adalah variabel yang digunakan untuk menyimpan nilai yang menentukan kecepatan gerak dari Ulo. Nilai ini akan ditentukan oleh user melalui input. Nilai yang dapat ditentukan oleh user adalah 1, 2, dan 3, yang masing-masing menunjukkan kecepatan cepat, sedang, dan lambat. Setelah user memasukkan input, nilai akan disimpan di dalam variabel Kecepatan untuk digunakan dalam perhitungan gerak Ulo.

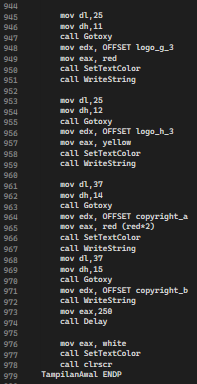
1. **Prosedur Proses Utama**

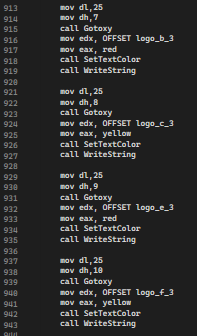
Pada bagian kode ini, digunakan Prosedur TampilanAwal yang digunakan untuk menampilkan logo permainan yang ditampilkan saat awal permainan. Pertama, dilakukan pemanggilan fungsi Gotoxy yang digunakan untuk menentukan posisi cursor saat menuliskan teks. Kemudian, dilakukan pemanggilan fungsi WriteString untuk menuliskan teks pada posisi cursor yang telah ditentukan. Selain itu, digunakan juga fungsi SetTextColor untuk menentukan warna teks yang akan dituliskan. Setelah semua teks dituliskan, dilakukan delay selama 250 milisecond sebelum menampilkan logo yang kedua. Proses ini dilakukan berulang hingga semua teks yang ditampilkan di logo permainan ditampilkan. Setelah itu, dilakukan pemanggilan fungsi untuk menampilkan string copyright\_a dan copyright\_b yang menyimpan informasi tentang pembuat dan tahun pembuatan permainan. Kemudian, dilakukan delay selama 250 milisecond sebelum melanjutkan ke proses selanjutnya.











Gambar 18 Cuplikan kodingan ke-7 Aplikasi Ulo Asem **(Sumber: Penulis)**

Pada bagian kode ini, digunakan Prosedur TampilanAwal yang digunakan untuk menampilkan logo dan copyright pada saat permainan dimulai.

Syntax yang digunakan pada kode ini adalah:

* mov: digunakan untuk memindahkan data dari register atau memory ke register lain
* dl, dh: digunakan untuk menentukan posisi x dan y dari cursor saat menulis string
* call: digunakan untuk memanggil Prosedur atau function yang telah didefinisikan sebelumnya, seperti Gotoxy, SetTextColor, dan WriteString
* OFFSET: digunakan untuk mengambil alamat memory dari suatu variable
* eax: register yang digunakan untuk menyimpan data
* yellow, red (red\*2): digunakan untuk memberikan warna pada string yang ditulis
* Delay: digunakan untuk memberikan delay selama 250 milisecond

Secara garis besar, kode ini digunakan untuk menampilkan logo pada saat permainan dimulai dan memberikan warna pada logo dan copyright. Kemudian menggunakan call Gotoxy untuk menentukan posisi dari cursor, lalu menggunakan call WriteString untuk menuliskan string pada posisi yang telah ditentukan. Dan di akhir menggunakan Delay untuk memberikan delay selama 250 milisecond.

Pada bagian kode ini, digunakan beberapa prosedur yang digunakan untuk menampilkan logo dan copyright pada saat awal permainan.

Prosedur TampilanAwal digunakan untuk menampilkan logo sebelum permainan dimulai. Pada prosedur ini, digunakan beberapa perintah seperti mov, call, dan delay untuk mengatur posisi dan tampilan logo.

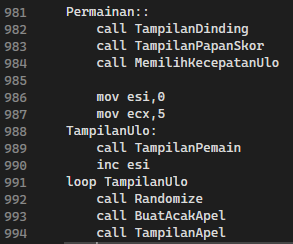
Pertama, digunakan perintah mov untuk mengatur posisi cursor pada koordinat (dl, dh) dan digunakan perintah call Gotoxy untuk menempatkan cursor pada posisi tersebut. Kemudian, digunakan perintah mov edx, OFFSET untuk menyimpan alamat dari setiap baris logo yang akan ditampilkan. Setelah itu, digunakan perintah call WriteString untuk menampilkan logo pada posisi yang telah ditentukan sebelumnya.

Selain itu, digunakan perintah call SetTextColor untuk mengubah warna tulisan logo dan perintah delay untuk menunda waktu sebelum logo ditampilkan.

etelah logo ditampilkan, copyright ditampilkan dengan cara yang sama dengan logo yaitu dengan mengatur posisi cursor, menyimpan alamat dari setiap baris copyright, menampilkan copyright dan mengubah warna tulisan.

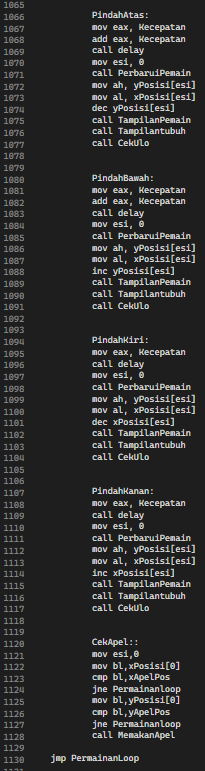
Setelah proses penampilan logo dan copyright selesai, prosedur TampilanAwal akan selesai juga.

1. Prosedur Proses Utama



Gambar 19 Cuplikan kodingan ke-8 Aplikasi Ulo Asem **(Sumber: Penulis)**

Pada bagian kode ini, terdapat beberapa prosedur yang dijalankan yaitu TampilanDinding, TampilanPapanSkor, MemilihKecepatanUlo, TampilanUlo, dan TampilanApel. Prosedur TampilanDinding digunakan untuk menampilkan tembok pada layar, TampilanPapanSkor digunakan untuk menampilkan papan skor pada layar, MemilihKecepatanUlo digunakan untuk memberikan pemain pilihan untuk memilih kecepatan Ulo, TampilanUlo digunakan untuk menampilkan Ulo dengan panjang 5, dan TampilanApel digunakan untuk menampilkan Apel pada layar. Pada TampilanUlo, digunakan loop dengan instruksi loop TampilanUlo dan digunakan mov esi,0 dan mov ecx,5 sebagai counter untuk mengatur jumlah panjang Ulo yang akan ditampilkan. Kemudian digunakan pula prosedur Randomize dan BuatAcakApel yang digunakan untuk mengacak posisi Apel yang akan ditampilkan.



Gambar 20 Cuplikan kodingan ke-9 Aplikasi Ulo Asem **(Sumber: Penulis)**

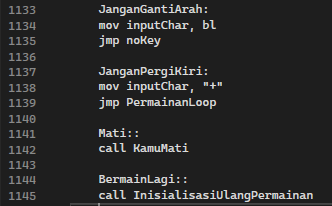
Pada bagian kode ini, digunakan label PermainanLoop untuk melakukan perulangan yang akan terus berjalan selama permainan berlangsung. Pada dalam perulangan tersebut, digunakan beberapa instruksi seperti "call ReadKey" untuk membaca input dari keyboard yang diberikan pemain, "cmp inputChar,"x"" untuk mengecek apakah pemain ingin keluar dari permainan dengan menekan tombol x, dan "cmp inputChar,"w"" untuk mengecek apakah pemain ingin bergerak ke atas. Selanjutnya, ada juga instruksi seperti "CekBawah: cmp TerakhirInputKarakter, "w"" yang digunakan untuk mengecek apakah pemain baru saja bergerak ke atas sebelumnya, sehingga tidak bisa bergerak ke bawah langsung setelah itu.

Selain itu, terdapat juga "CekKiri" dan "CekKanan" yang digunakan untuk mengecek apakah pemain ingin bergerak ke kiri atau kanan. Pada setiap perintah gerakan, digunakan juga perintah seperti "mov cl, yPosisiDinding[1]" untuk mengambil posisi dinding dari bawah dan "cmp yPosisi[0],cl" untuk membandingkan posisi Ulo dengan posisi dinding. Jika posisi Ulo sama atau kurang dari posisi dinding, maka Ulo akan mati dan permainan akan berakhir. Selain itu, jika pemain menekan tombol x, maka permainan akan segera berakhir dan akan ditampilkan skor dan opsi untuk mencoba lagi.

Prosedur CekAtas, CekBawah, CekKiri, dan CekKanan digunakan untuk mengecek apakah Ulo akan bergerak ke atas, bawah, kiri, atau kanan. Pada setiap Prosedur, terdapat pengecekan untuk menentukan apakah Ulo akan mati atau tidak. Jika Ulo akan mati, maka program akan langsung pindah ke Prosedur Mati. Jika Ulo tidak akan mati, maka program akan pindah ke Prosedur PindahAtas, PindahBawah, PindahKiri, atau PindahKanan.

Prosedur CekAtas, CekBawah, CekKiri, dan CekKanan juga digunakan untuk mencegah Ulo berbalik arah secara tiba-tiba. Misalnya, jika pemain baru saja bergerak ke atas dan segera menekan tombol ke bawah, maka Ulo tidak akan bergerak ke bawah dan tetap bergerak ke atas. Ini dilakukan dengan mengecek variabel TerakhirInputKarakter yang menyimpan input terakhir dari pemain.

Prosedur TampilanDinding digunakan untuk menampilkan dinding pada permainan. Posisi dinding ditentukan oleh xPosisiDinding dan yPosisiDinding yang sudah didefinisikan sebelumnya. Prosedur ini akan menggunakan loop untuk menampilkan dinding secara berulang-ulang sesuai dengan jumlah dinding yang ditentukan dalam xPosisiDinding dan yPosisiDinding. Setiap kali loop dijalankan, posisi dinding akan diupdate dan ditampilkan pada layar. Juga, Prosedur TampilanPapanSkor digunakan untuk menampilkan skor dan skor tertinggi pada permainan, yang diperoleh dari variabel Skor dan SkorTertinggi yang sudah didefinisikan sebelumnya.



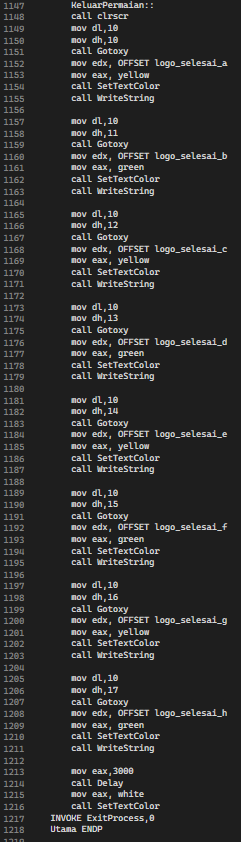
Gambar 21 Cuplikan kodingan ke-10 Aplikasi Ulo Asem **(Sumber: Penulis)**

Procedure JanganGantiArah digunakan untuk mencegah pemain dari mengubah arah gerak Ulo saat permainan sedang berlangsung. Jika pemain mencoba untuk mengubah arah gerak Ulo dengan menekan tombol yang berlawanan dengan arah yang sedang diterapkan, maka input akan diabaikan dan Ulo akan tetap bergerak dalam arah yang sama.

Procedure JanganPergiKiri digunakan untuk mencegah Ulo untuk bergerak ke Kiri pada saat permainan baru dimulai. Jika pemain mencoba untuk bergerak ke Kiri pada saat permainan baru dimulai, maka input akan diabaikan dan Ulo akan tetap bergerak ke Kanan.

Procedure Mati digunakan untuk menampilkan pesan "Kamu Mati" pada layar dan mengakhiri permainan.

Procedure BermainLagi digunakan untuk menginisialisasi ulang semua variabel dan mengulangi permainan dari awal.



Gambar 22 Cuplikan kodingan ke-11 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kode tersebut digunakan untuk menampilkan logo "Sampai Jumpa" pada saat permainan selesai. Logo tersebut ditentukan oleh beberapa variabel string yang diberi nama logo\_selesai\_a sampai logo\_selesai\_h. Setiap variabel tersebut diisi dengan karakter yang digunakan untuk membentuk tulisan "Sampai Jumpa" yang terdiri dari beberapa baris.

Posisi Gotoxy yang digunakan untuk menentukan koordinat posisi tulisan yang ingin ditampilkan. Dalam kode ini, posisi ditentukan dengan menentukan nilai dl dan dh. Dalam setiap perintah Gotoxy, dl digunakan untuk menentukan posisi horizontal (koordinat x) dan dh digunakan untuk menentukan posisi vertikal (koordinat y).

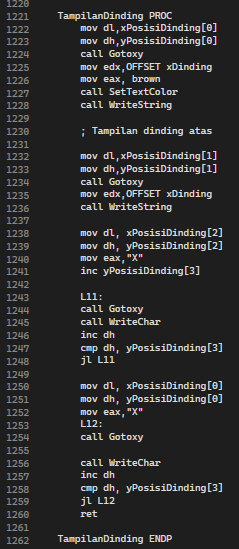
Setelah posisi ditentukan, perintah WriteString digunakan untuk menampilkan tulisan yang ditentukan oleh variabel edx yang diisi dengan offset dari variabel yang digunakan untuk membentuk logo.

Perintah ini diikuti dengan perintah SetTextColor untuk menentukan warna tulisan yang akan ditampilkan. Setiap baris tulisan memiliki warna yang berbeda yaitu yellow,green,dan white.

Proses yang dilakukan pada kode tersebut adalah menampilkan logo selesai yang terdiri dari 8 baris yang ditulis dengan ASCII "Sampai Jumpa" pada posisi (10,10) sampai (10,17) di layar console.

Setiap baris yang ditampilkan memiliki offset yang berbeda yaitu logo\_selesai\_a, logo\_selesai\_b, logo\_selesai\_c, dst. Offset digunakan untuk menentukan lokasi dari variabel atau label dalam memori.

Setelah semua baris ditampilkan, warna teks akan diubah menjadi hijau dan kuning dengan menggunakan perintah SetTextColor dan digunakan delay selama 3000 mili second. Kemudian program akan dihentikan dengan perintah ExitProcess.



Gambar 23 Cuplikan kodingan ke-12 Aplikasi Ulo Asem **(Sumber: Penulis)**

Pada bagian kode di atas, ada sebuah prosedur bernama "TampilanDinding" yang digunakan untuk menampilkan dinding pada layar. Prosedur ini menggunakan beberapa perintah Gotoxy dan WriteString yang digunakan untuk mengatur posisi dan menuliskan karakter "X" sebagai simbol dinding.

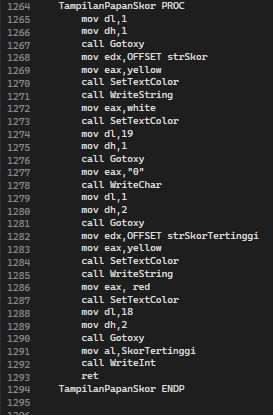
Pertama, perintah Gotoxy digunakan untuk mengatur posisi cursor di koordinat xPosisiDinding[0], yPosisiDinding[0]. Kemudian, perintah SetTextColor digunakan untuk memberikan warna pada dinding. Setelah itu, perintah WriteString digunakan untuk menuliskan string "X" sebagai simbol dinding.

Selanjutnya, perintah Gotoxy digunakan lagi untuk mengatur posisi cursor di koordinat xPosisiDinding[1], yPosisiDinding[1], dan perintah WriteString digunakan untuk menuliskan string "X" sebagai simbol dinding pada posisi tersebut.

Setelah itu, perintah Gotoxy digunakan untuk mengatur posisi cursor di koordinat xPosisiDinding[2], yPosisiDinding[2], dan perintah WriteChar digunakan untuk menuliskan karakter "X" sebagai simbol dinding. Kemudian, perintah loop digunakan untuk mengulang proses ini sampai posisi y cursor mencapai yPosisiDinding[3].

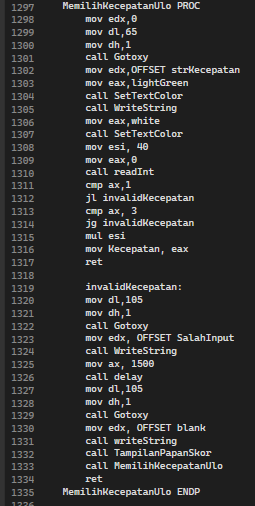
Kemudian, perintah Gotoxy digunakan lagi untuk mengatur posisi cursor di koordinat xPosisiDinding[0], yPosisiDinding[0], dan perintah WriteChar digunakan untuk menuliskan karakter "X" sebagai simbol dinding. Kemudian, perintah loop digunakan untuk mengulang proses ini sampai posisi y cursor mencapai yPosisiDinding[3].

Pada akhir prosedur, perintah ret digunakan untuk mengakhiri prosedur dan kembali ke alamat yang memanggil prosedur.



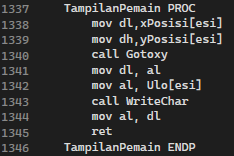
Gambar 24 Cuplikan kodingan ke-13 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kutipan kode di atas adalah prosedur yang digunakan untuk menampilkan papan skor pada permainan. Prosedur ini dimulai dengan memanggil fungsi Gotoxy untuk menentukan posisi kursor pada koordinat (1,1) dan mencetak string "Skor: " pada posisi tersebut. Kemudian, ditampilkan juga nilai skor saat ini yang diinisialisasi dengan 0. Selanjutnya, posisi kursor dipindahkan ke koordinat (1,2) dan mencetak string "Skor Tertinggi: " dan nilai skor tertinggi yang disimpan dalam variabel SkorTertinggi. Setelah itu, warna teks diatur kembali menjadi default (putih) dan prosedur selesai dengan perintah ret.



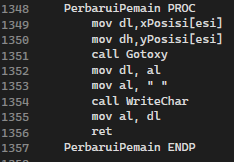
Gambar 25 Cuplikan kodingan ke-14 Aplikasi Ulo Asem **(Sumber: Penulis)**

Pada prosedur MemilihKecepatanUlo, digunakan untuk meminta input dari pemain untuk memilih kecepatan Ulo yang diinginkan. Pada bagian awal, digunakan perintah Gotoxy untuk menempatkan cursor pada posisi tertentu di layar dan kemudian mencetak string "Memilih Kecepatan : " menggunakan prosedur WriteString. Kemudian, digunakan perintah readInt untuk membaca input dari pemain dan menyimpannya ke dalam variabel eax. Selanjutnya, digunakan perintah cmp untuk membandingkan nilai yang diterima dari input pemain dengan 1 dan 3. Jika input pemain di luar jangkauan ini, maka program akan jump ke label invalidKecepatan. Pada label ini, digunakan perintah Gotoxy untuk menempatkan cursor pada posisi tertentu dan mencetak string "Input Salah" menggunakan prosedur WriteString. Kemudian, digunakan perintah delay untuk menunda program selama 1,5 detik sebelum menghapus pesan error dan memanggil prosedur MemilihKecepatanUlo lagi. Setelah input yang valid diterima, program akan mengalikan nilai input dengan 40 (yang merupakan perbedaan milisecond setiap level kecepatan) dan menyimpan nilai tersebut ke dalam variabel Kecepatan. Kemudian, prosedur akan diakhiri dengan perintah ret.



Gambar 26 Cuplikan kodingan ke-15 Aplikasi Ulo Asem **(Sumber: Penulis)**

Bagian ini adalah prosedur yang digunakan untuk menampilkan karakter pemain pada posisi xPosisi[esi] dan yPosisi[esi]. Prosedur ini menggunakan fungsi Gotoxy untuk menentukan posisi dari karakter pemain. Kemudian, karakter pemain disimpan dalam register al dan ditampilkan menggunakan fungsi WriteChar. Setelah itu, register al di-restore ke nilai sebelumnya sebagai sementara register. Prosedur ini kemudian akan mengembalikan kontrol ke program utama setelah selesai dijalankan.



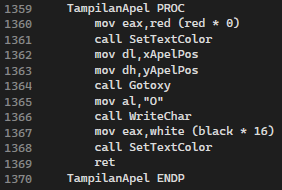
Gambar 27 Cuplikan kodingan ke-16 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kode di atas adalah prosedur yang digunakan untuk menghapus posisi sebelumnya dari pemain sebelum menampilkan posisi baru dari pemain. Prosedur ini dikenal sebagai "PerbaruiPemain".

Pertama, prosedur ini mengambil posisi x dan y dari pemain yang disimpan dalam array xPosisi dan yPosisi yang ditunjuk oleh register esi. Kemudian, prosedur memanggil prosedur Gotoxy untuk mengatur posisi cursor pada posisi x dan y yang diambil.

Kemudian, prosedur menyimpan sementara nilai dari register al ke dl. Lalu, register al diisi dengan karakter "spasi" untuk menghapus posisi sebelumnya dari pemain. Prosedur WriteChar digunakan untuk mencetak karakter "spasi" pada posisi x dan y yang diambil.

Setelah itu, nilai dari register al diisi kembali dengan nilai yang disimpan sementara di register dl dengan perintah "mov al, dl" sebelum prosedur diakhiri dengan perintah "ret". Dengan demikian, tampilan Pemain di posisi (xPosisi, yPosisi) akan dihapus dan siap untuk digerakkan ke posisi baru.



Gambar 28 Cuplikan kodingan ke-17 Aplikasi Ulo Asem **(Sumber: Penulis)**

Prosedur "TampilanApel" digunakan untuk menampilkan simbol "O" sebagai representasi dari buah apel. Pertama, warna teks diganti menjadi merah dengan memanggil fungsi "SetTextColor" dan passing nilai "red" (red \* 0) ke dalamnya. Kemudian, posisi cursor ditempatkan di (xApelPos, yApelPos) dengan memanggil fungsi "Gotoxy" dan passing nilai dari xApelPos dan yApelPos ke dalamnya. Setelah itu, simbol "O" ditampilkan pada posisi cursor saat ini dengan memanggil fungsi "WriteChar" dan passing nilai "O" ke dalamnya. Terakhir, warna teks dikembalikan menjadi hitam dan putih dengan memanggil fungsi "SetTextColor" dan passing nilai "white" (black \* 16) ke dalamnya.



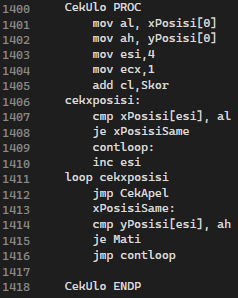
Gambar 29 Cuplikan kodingan ke-18 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kode di atas adalah prosedur untuk mengacak posisi buah apel pada layar. Pertama, fungsi RandomRange dipanggil dengan argumen 49, yang digunakan untuk menghasilkan angka acak dari 0 hingga 49. Kemudian, hasil dari fungsi tersebut ditambahkan dengan 35, sehingga rentang angka yang dihasilkan menjadi 35 hingga 84. Angka acak ini kemudian disimpan dalam register AL, yang digunakan sebagai posisi x dari buah apel.

Selanjutnya, fungsi RandomRange dipanggil lagi dengan argumen 17, yang digunakan untuk menghasilkan angka acak dari 0 hingga 17. Kemudian, hasil dari fungsi tersebut ditambahkan dengan 6, sehingga rentang angka yang dihasilkan menjadi 6 hingga 23. Angka acak ini kemudian disimpan dalam register AL, yang digunakan sebagai posisi y dari buah apel.

Setelah posisi x dan y dari buah apel ditentukan, kode mengecek apakah posisi x dan y dari buah apel sama dengan posisi x dan y dari ular. Jika posisi x dan y dari buah apel sama dengan posisi x dan y dari ular, maka prosedur BuatAcakApel akan dipanggil kembali untuk menentukan koordinat buah apel yang baru dan tidak ditempatkan pada posisi yang sama dengan ular. Namun, jika posisi x dan y dari buah apel tidak sama dengan posisi x dan y dari ular, maka kode akan kembali ke perulangan dan melanjutkan untuk mengecek posisi x dan y dari segmen ular selanjutnya.

Setelah posisi x dan y dari buah apel diacak, kode kemudian memasukkan nilai 5 pada register ecx dan menambahkannya dengan skor. Kemudian, register esi diatur sebagai 0. Lalu, ada perulangan yang disebut "CekApelxPosisi" yang digunakan untuk memeriksa apakah posisi x dari buah apel sama dengan posisi x dari Ulo pada indeks esi yang saat ini diperiksa. Jika sama, kode akan melompat ke perintah "CekApelyPosisi" yang digunakan untuk memeriksa apakah posisi y dari buah apel sama dengan posisi y dari Ulo pada indeks esi yang saat ini diperiksa. Jika tidak sama, kode akan kembali ke perintah "Melanjutkanloop" dan menambahkan 1 pada esi, sehingga indeks berikutnya akan diperiksa dalam perulangan. Jika posisi x dan y dari buah apel sama dengan posisi x dan y dari Ulo pada indeks esi yang saat ini diperiksa, maka kode akan memanggil prosedur "BuatAcakApel" lagi untuk mengacak posisi buah apel yang baru. Setelah perulangan selesai, kode akan kembali ke prosedur yang memanggil prosedur "BuatAcakApel".

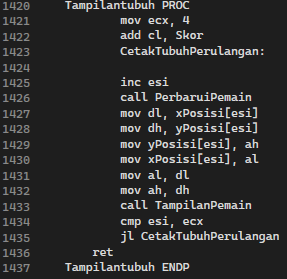


Gambar 30 Cuplikan kodingan ke-19 Aplikasi Ulo Asem **(Sumber: Penulis)**

Prosedur CekUlo digunakan untuk mengecek apakah ulo (snake) yang digerakkan oleh pemain menabrak tubuhnya sendiri. Hal ini dilakukan dengan membandingkan posisi x dan y dari kepala Ulo (indeks 0) dengan posisi x dan y dari segmen-segmen tubuh Ulo lainnya (indeks 4 hingga Skor+1).

Pertama, kode menyimpan posisi x dan y dari kepala Ulo (indeks 0) ke dalam register AL dan AH. Kemudian, kode mengatur register ESI sebagai 4 (indeks 4 dari segmen tubuh Ulo) dan menambahkan skor saat ini ke register ECX. Kemudian, kode mengeksekusi loop dengan perintah "loop cekxposisi" yang akan diulang sesuai dengan skor saat ini. Dalam loop ini, kode membandingkan posisi x dari Ulo pada indeks ESI dengan posisi x dari kepala Ulo. Jika posisi x sama, maka kode akan membandingkan posisi y dari Ulo pada indeks ESI dengan posisi y dari kepala Ulo. Jika kedua posisi x dan y sama, maka Ulo mati (menjalankan prosedur Mati). Jika posisi x atau posisi y tidak sama, loop akan dilanjutkan untuk membandingkan posisi Ulo selanjutnya.

Jika loop selesai tanpa Ulo mati, maka kode akan melompat ke prosedur CekApel untuk mengecek apakah Ulo telah menabrak buah.



Gambar 31 Cuplikan kodingan ke-20 Aplikasi Ulo Asem **(Sumber: Penulis)**

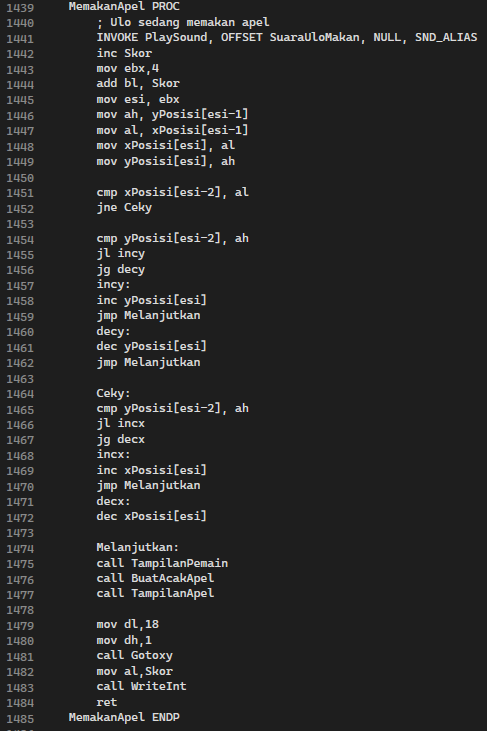
Kode di atas adalah prosedur yang digunakan untuk mencetak tubuh dari Ulo (ular) pada game. Prosedur ini memiliki nama Tampilantubuh.

Pada baris pertama, register ECX diinisialisasi dengan 4 dan kemudian ditambah dengan skor. Ini digunakan untuk menentukan jumlah perulangan yang diperlukan untuk mencetak tubuh Ulo dan ekornya.

Pada baris ke-3, ada perulangan dengan nama CetakTubuhPerulangan. Dalam perulangan ini, register ESI diincrement (ESI akan digunakan sebagai indeks untuk mengakses elemen dari array xPosisi dan yPosisi). Kemudian, prosedur PerbaruiPemain dipanggil.

Pada baris ke-5 dan 6, register DL dan DH diisi dengan xPosisi[esi] dan yPosisi[esi], yaitu posisi saat ini dari segmen tubuh Ulo. Kemudian, pada baris ke-7 dan 8, posisi baru ditentukan untuk segmen tubuh Ulo dengan mengubah xPosisi[esi] dan yPosisi[esi]. Pada baris ke-9 dan 10, posisi lama dikembalikan ke register AL dan AH. Kemudian, prosedur TampilanPemain dipanggil untuk mencetak segmen tubuh Ulo pada posisi baru.

Pada baris ke-11, register ESI dibandingkan dengan ECX. Jika ESI kurang dari ECX, maka perulangan CetakTubuhPerulangan akan diulang. Setelah perulangan selesai, prosedur akan mengembalikan kontrol ke pemanggil dengan perintah ret.



Gambar 32 Cuplikan kodingan ke-21 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kode di atas merupakan bagian dari sebuah program yang menangani logika untuk "memakan apel" dalam sebuah permainan ular. Prosedur ini disebut "MemakanApel".

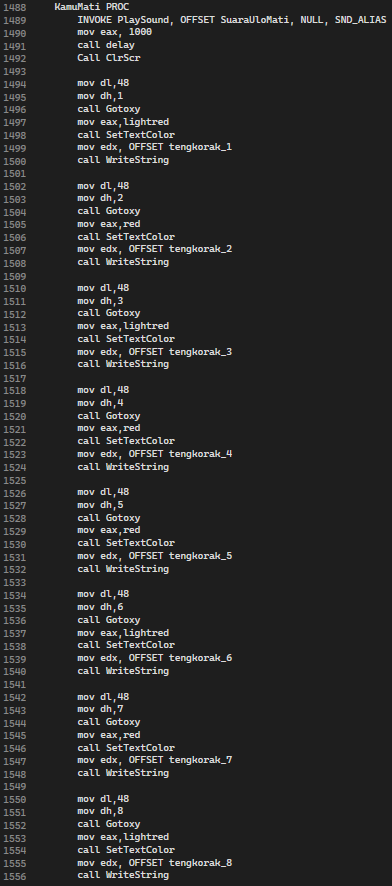
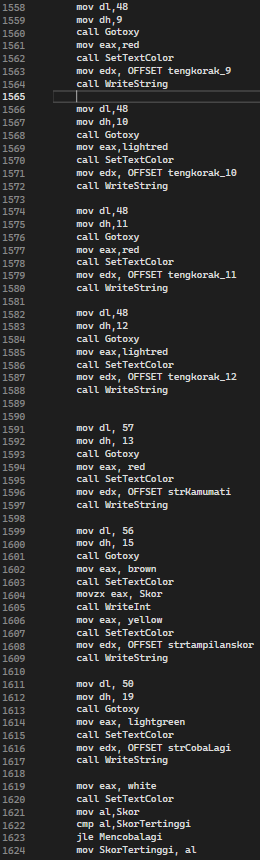
Pertama, prosedur ini memanggil suara Ulo yang sedang makan dengan menggunakan "INVOKE PlaySound, OFFSET SuaraUloMakan, NULL, SND\_ALIAS". Kemudian, skor ditambah satu dengan menggunakan "inc Skor".

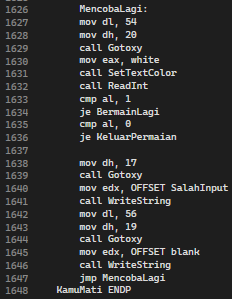
Lalu, prosedur ini menggunakan "mov ebx,4" dan "add bl, Skor" untuk menentukan posisi dari ekor baru dari ular. Kemudian, posisi dari ekor baru diatur sama dengan posisi dari ekor lama dengan menggunakan "mov esi, ebx", "mov ah, yPosisi[esi-1]" dan "mov al, xPosisi[esi-1]".

Selanjutnya, prosedur ini mengecek apakah ekor lama dan segmen ulo sebelum itu berada pada sumbu x atau y dengan "cmp xPosisi[esi-2], al" dan "jne Ceky". Jika ekor lama dan segmen ulo sebelum itu berada pada sumbu y, prosedur ini mengecek apakah ekor yang baru harus di atas atau di bawah ekor yang lama dengan "cmp yPosisi[esi-2], ah" dan "jl incy" atau "jg decy". Jika ekor yang baru harus di atas ekor yang lama, prosedur ini menambahkan 1 pada posisi y dari ekor baru dengan "inc yPosisi[esi]" dan jika ekor yang baru harus di bawah ekor yang lama, prosedur ini mengurangi 1 pada posisi y dari ekor baru dengan "dec yPosisi[esi]".

Jika ekor lama dan segmen ulo sebelum itu berada pada sumbu x, prosedur ini mengecek apakah ekor yang baru harus di kanan atau di kiri dari ekor yang lama dengan "cmp yPosisi[esi-2], ah" dan "jl incx" atau "jg decx". Jika ekor yang baru harus di kanan ekor yang lama, prosedur ini menambahkan 1 pada posisi x dari ekor baru dengan "inc xPosisi[esi]" dan jika ekor yang baru harus di kiri ekor yang lama, prosedur ini mengurangi 1 pada posisi x dari ekor baru dengan "dec xPosisi[esi]".

Setelah itu, prosedur ini memanggil pros .edur TampilanPemain untuk menampilkan ular di layar, memanggil prosedur BuatAcakApel untuk mengenerate posisi apel yang baru secara acak, dan memanggil prosedur TampilanApel untuk menampilkan apel di layar. Kemudian, prosedur ini menggunakan instruksi mov dl, 18 dan mov dh, 1 untuk mengatur posisi cursor di layar, dan memanggil prosedur Gotoxy untuk menempatkan cursor di posisi yang ditentukan. Selanjutnya, prosedur ini menggunakan instruksi mov al, Skor untuk memindahkan nilai skor ke register AL, dan memanggil prosedur WriteInt untuk menulis skor pada layar. Prosedur ini kemudian menyelesaikan dengan instruksi ret untuk kembali ke program utama.



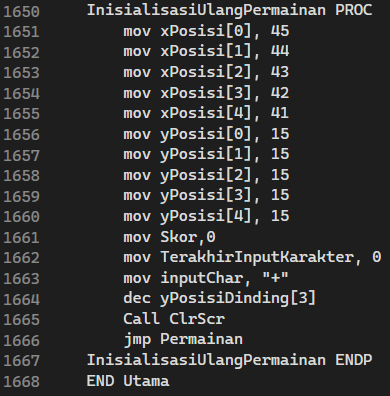


Gambar 33 Cuplikan kodingan ke-22 Aplikasi Ulo Asem **(Sumber: Penulis)**

Kode diatas adalah prosedur yang ditujukan untuk menampilkan tampilan ketika pemain kalah dalam permainan. Pertama, prosedur ini memanggil prosedur PlaySound untuk memainkan suara "SuaraUloMati", lalu menghentikan program selama 1000 milidetik dengan memanggil prosedur delay. Kemudian, prosedur ClrScr dipanggil untuk membersihkan layar.

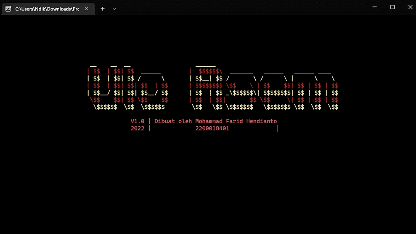
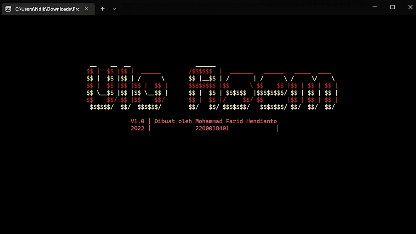
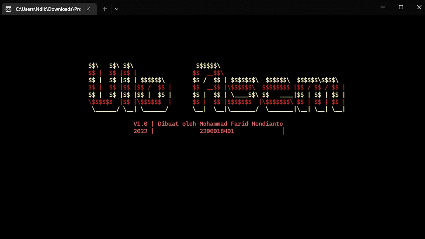
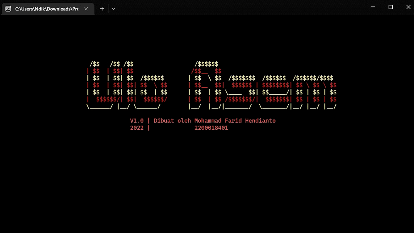
Setelah itu, terdapat beberapa perintah yang digunakan untuk menampilkan gambar tengkorak pada layar dengan menggunakan prosedur Gotoxy, SetTextColor, dan WriteString. Posisi dan warna dari setiap baris tengkorak ditentukan dengan mengatur nilai dl, dh, dan eax sebelum memanggil prosedur tersebut.

Selanjutnya, prosedur menampilkan pesan "Kamu Mati" dan skor akhir pemain dengan menggunakan prosedur Gotoxy, SetTextColor dan WriteString. Kemudian, prosedur menampilkan pilihan "Coba Lagi?" dan meminta input dari pemain untuk memutuskan apakah ingin bermain lagi atau keluar dari permainan. Pertama, posisi cursor ditetapkan pada baris 20 dan kolom 54 menggunakan prosedur Gotoxy. Kemudian, warna teks diubah menjadi putih menggunakan prosedur SetTextColor. Input diterima dari pemain menggunakan prosedur ReadInt dan disimpan di register AL. Kemudian, input dibandingkan dengan 1 untuk memutuskan apakah pemain ingin bermain lagi atau 0 jika pemain ingin keluar dari permainan. Jika input sesuai dengan kondisi yang ditentukan, maka prosedur akan mengarahkan ke prosedur BermainLagi atau KeluarPermainan. Jika input tidak sesuai dengan kondisi yang ditentukan, maka akan ditampilkan pesan "Salah Input" dan input sebelumnya akan dibersihkan. Kemudian, prosedur menampilkan pilihan "Coba Lagi?" dan meminta input dari pemain untuk memutuskan apakah ingin bermain lagi atau keluar dari permainan. Jika pemain memasukkan input 1, maka permainan akan diulang dari awal. Jika pemain memasukkan input 0, maka permainan akan berakhir. Jika pemain memasukkan input yang salah, maka akan ditampilkan pesan "Salah Input" dan pemain akan diminta untuk memasukkan input lagi. Kemudian, prosedur akan membandingkan skor saat ini dengan skor tertinggi yang pernah didapatkan sebelumnya dan menyimpan skor tertinggi jika skor saat ini lebih tinggi.



Gambar 34 Cuplikan kodingan ke-22 Aplikasi Ulo Asem **(Sumber: Penulis)**

InisialisasiUlangPermainan adalah prosedur yang digunakan untuk mengatur ulang permainan ke kondisi awal. Pada bagian ini, kode tersebut mengatur posisi awal dari objek Ulo dan posisi dinding ke posisi awal, mengatur skor menjadi 0, mengatur inputChar dan TerakhirInputKarakter ke posisi awal, dan membersihkan layar. Kemudian, program akan menjump ke prosedur Permainan untuk memulai permainan kembali.

1. **Tampilan yang dihasilkan Aplikasi**

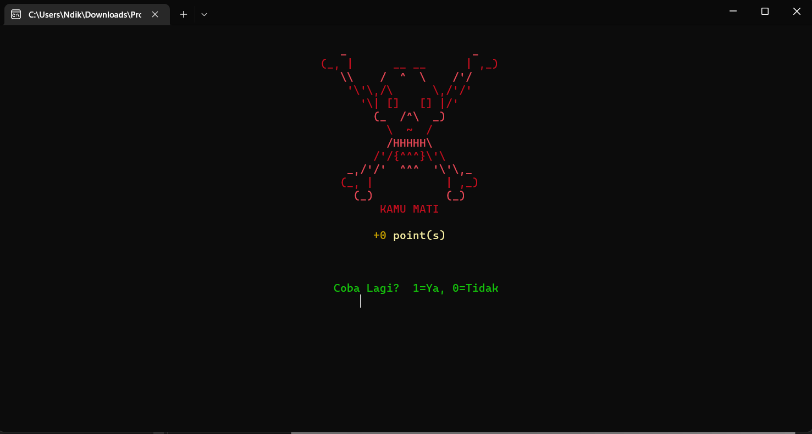
Gambar 35 4 frame untuk menampilkan animasi dalam tampilan saat aplikasi dijalankan. **(Sumber: Penulis)**

Dalam aplikasi Ulo Asem, tampilan pertama yang akan ditampilkan kepada pengguna adalah tampilan awal yang menampilkan logo game Ulo Asem dengan menggunakan ASCII Art. Logo Ulo Asem dalam berbentuk tulisan “Ulo Asem” akan ditampilkan dalam bentuk animasi. Selain itu, tampilan ini juga akan menampilkan informasi mengenai versi dan tahun pembuatan program Ulo Asem serta nama dan NIM pembuat aplikasi.

****

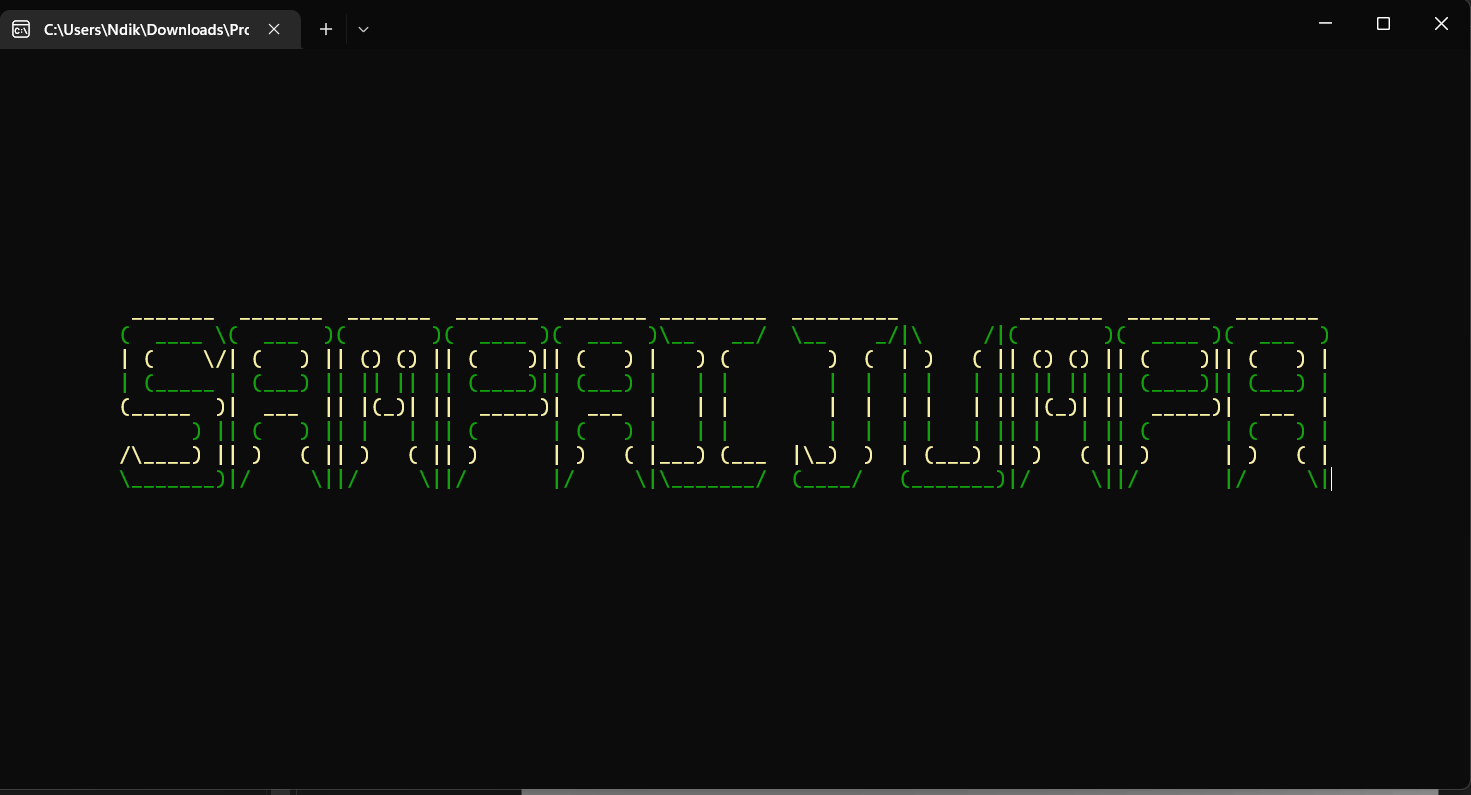
Gambar 36 Tampilan game.aplikasi Ulo Asem . **(Sumber: Penulis)**

Tampilan kedua adalah tampilan game, yang merupakan inti dari aplikasi ini. Tampilan ini akan menampilkan papan skor yang berisi skor yang sedang berlangsung di permainan, serta skor tertinggi dari permainan sebelumnya. Selain itu, tampilan ini juga akan menampilkan papan kecepatan ulo yang menunjukkan kecepatan ulo saat ini, serta papan permainan yang berisikan dinding, ulo, dan apel yang ditampilkan secara acak di dalam dinding permainan.



Gambar 37 Tampilan gameover.aplikasi Ulo Asem . **(Sumber: Penulis)**

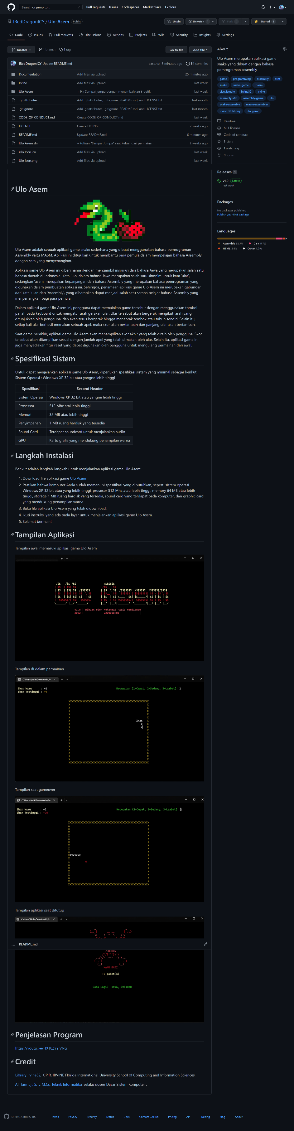
Tampilan ketiga adalah tampilan GameOver, yang akan ditampilkan ketika permainan berakhir. Tampilan ini akan menampilkan pesan GameOver dengan tulisan dan ASCII Art, serta menampilkan skor yang didapatkan selama permainan. Selain itu, tampilan ini juga akan menampilkan skor tertinggi yang pernah didapatkan selama permainan.



Gambar 38 Tampilan keluar.aplikasi Ulo Asem . **(Sumber: Penulis)**

Tampilan terakhir adalah Tampilan Keluar adalah tampilan yang ditampilkan ketika aplikasi Ulo Asem ditutup. Di tampilan ini, terdapat pesan yang ditampilkan dengan ASCII Art bertuliskan "SAMPAI JUMPA" yang menandakan bahwa pengguna telah selesai menggunakan aplikasi.

Semua tampilan aplikasi walaupun berbentuk teks, bisa menampilkan warna juga.

1. Status Unggah di Github

Gambar 39 **Tampilan Projek Ulo Asem di Github** . **(Sumber: Penulis)**

File projek aplikasi Ulo Asem sudah terupload ke dalam github. Dari source kode ulo.asm, dokumentasi, hingga linked library yang dibutuhkan seperti Irvine32.

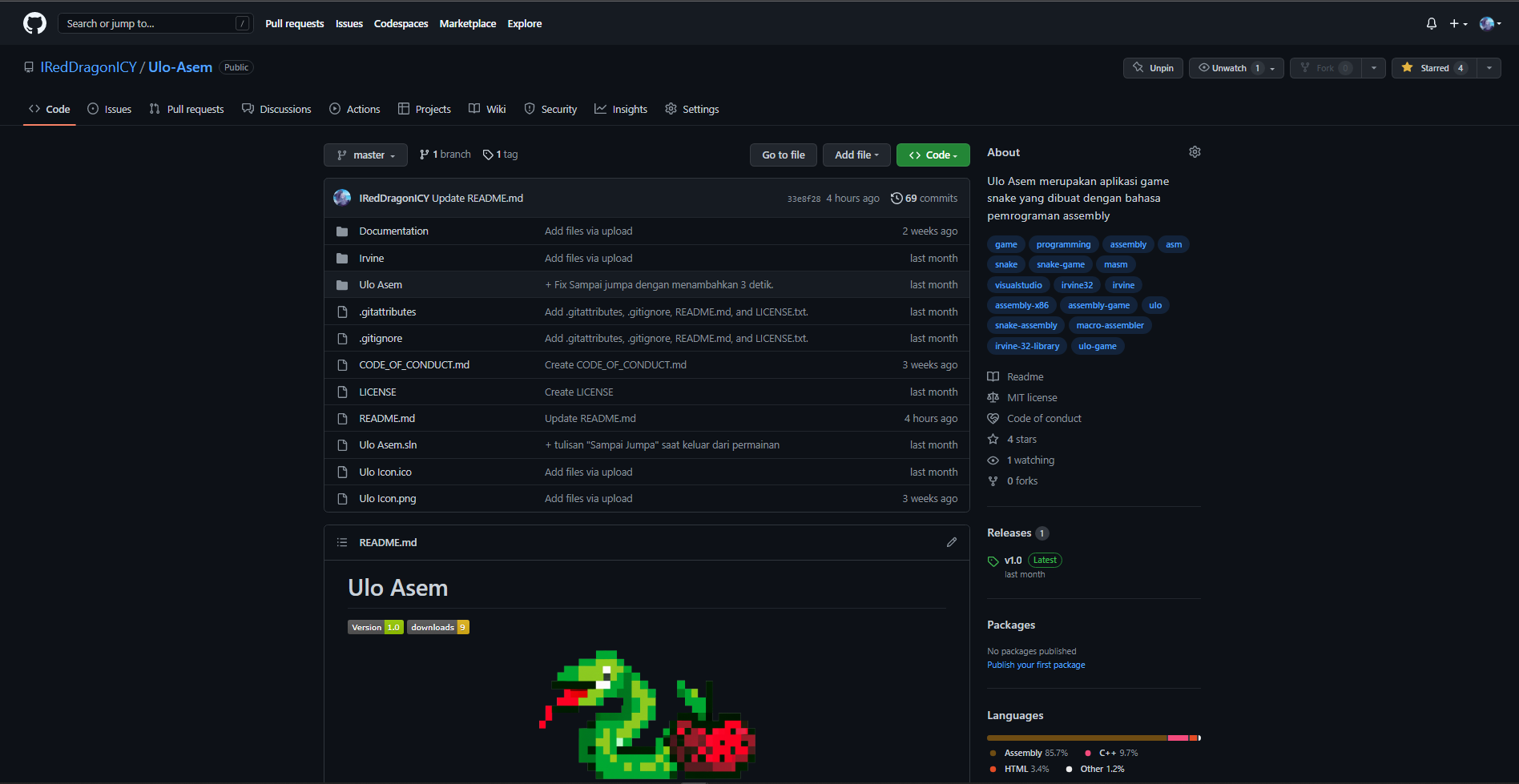
|  |  |
| --- | --- |
| **Link Github** | **:** [github.com/IRedDragonICY/Ulo-Asem](https://github.com/IRedDragonICY/Ulo-Asem) |
| **Shortlink** | **:** [bit.ly/uloasem](https://bit.ly/uloasem) |



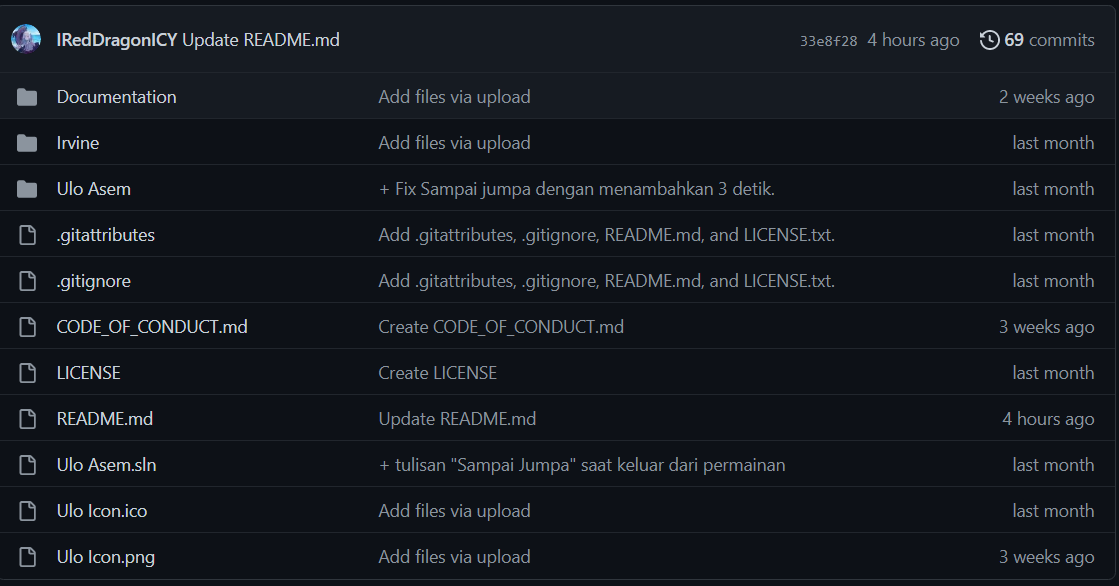
Gambar 40 QR kode untuk redirect ke link projek gituhb . **(Sumber: Penulis)**

Di dalam link projek github Ulo Asem, di dalam readme sudah dijelaskan penjelasan aplikasi secara ringkas seperti deskripsi aplikasi, spesifikasi sistem, langkah instalasi, cuplikan tampilan aplikasi berbentuk gif. Penjelasan program ke youtube dan ucapan terimakasih.

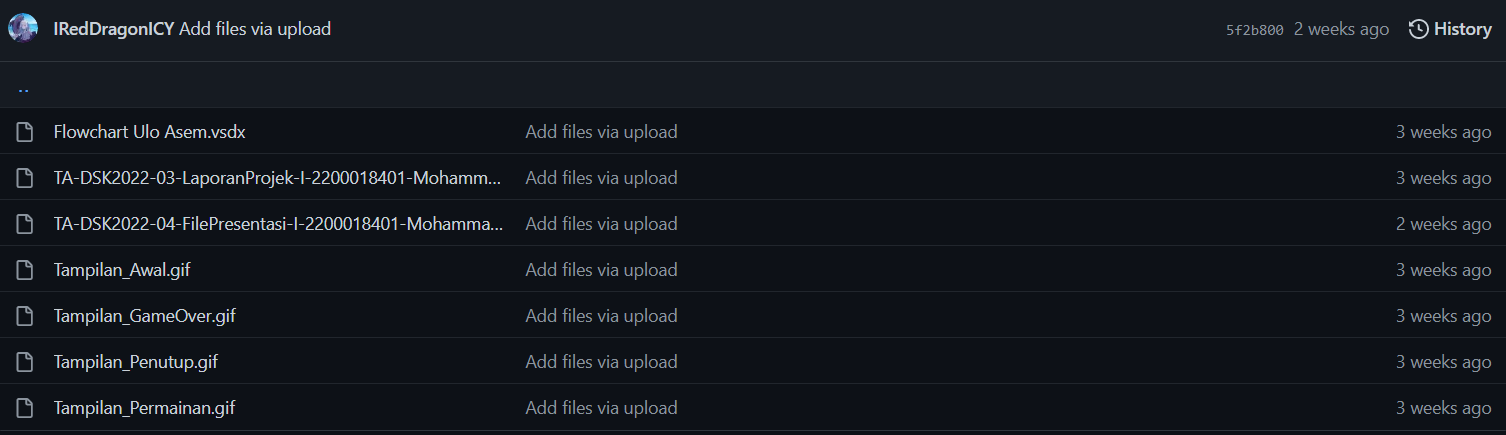
Selain itu, pada tabs releases sudah menampilkan aplikasi Ulo Asem yang siap di download. Di releases ditampilkan versi Ulo Asem ke v1.0.



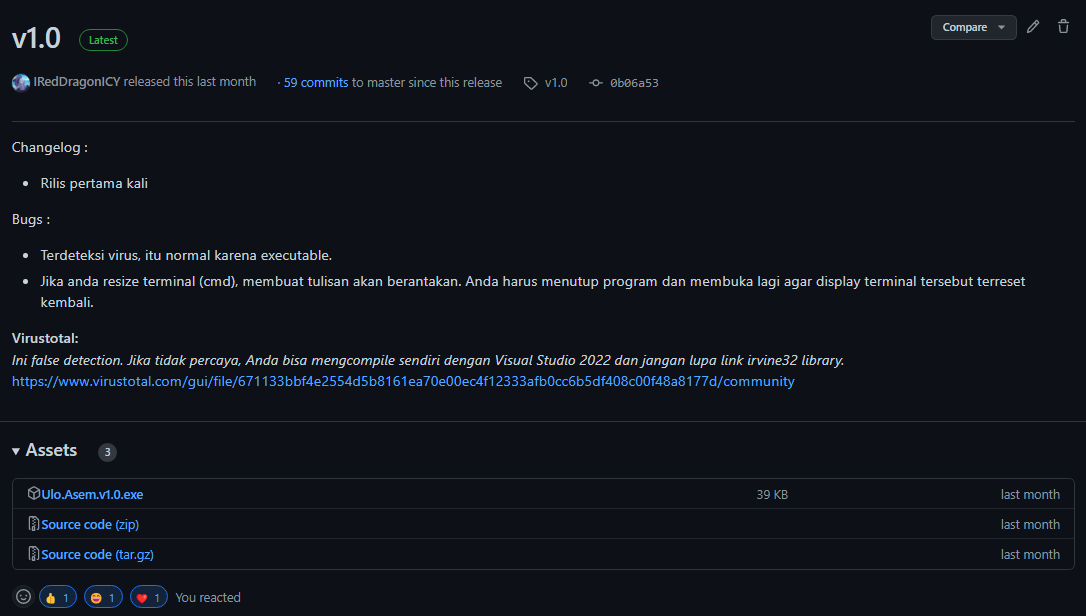
Gambar 41 Halaman depan projek Ulo Asem di github **(Sumber: Penulis)**



Gambar 42 Tampilan file projek di github **(Sumber: Penulis)**



Gambar 43 Tampilan file dokumentasi projek di github **(Sumber: Penulis)**



Gambar 44 Unggahan aplikasi Ulo Asem versi v1.0 **(Sumber: Penulis)**

1. **Analisis Pengerjaan Projek**
2. **Tinjauan dari Sisi Waktu**

Berikut adalah timeline pengerjaan projek Aplikasi Ulo Asem,

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **2022** | | | | **2023** | |
| 5 Des - 11 Des | 12 Des - 18 Des | 19 Des-25 Des | 26 Des-1 Jan | | 9 Jan-27 Jan |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |

**Keterangan:**

|  |  |
| --- | --- |
|  | : Pembuatan akun github |
|  | : Pemilihan judul projek aplikasi |
|  | : Pembuatan link projek aplikasi di github |
|  | : Pembuatan aplikasi |
|  | : Uji Coba Aplikasi |
| s | : Pembuatan laporan aplikasi |
|  | : Presentasi Aplikasi |
|  | : Bukti unggah Aplikasi |
|  | : Portofolio Aplikasi |

Sebelum membuat aplikasi Ulo Asem, ada beberapa tahap yang harus dilalui. Pertama, pembuatan akun di GitHub. GitHub adalah platform yang digunakan untuk pengembangan perangkat lunak yang menyediakan layanan hosting untuk repositori Git. Ini memungkinkan tim pengembang untuk berkolaborasi dalam pengembangan proyek, melacak perubahan kode, dan mengelola versi. Karena saya sudah memiliki akun GitHub, maka tugas untuk pembuatan akun tidak memakan banyak waktu, cukup 1 jam untuk menyelesaikan dan mengumpulkan bukti bahwa sudah memiliki akun GitHub.

Kemudian, minggu depan dilakukan pemilihan judul proyek aplikasi menggunakan bahasa assembly untuk tugas akhir mata kuliah Dasar Sistem Komputer. Diberikan waktu 1 minggu untuk menyelesaikan. Untuk mencari topik yang cocok, tidak terlalu banyak contoh projek yang sudah dibuat dalam bahasa assembly, mungkin dikarenakan merupakan bahasa yang paling bawah. Bahasa assembly yang tidak hanya TASM (Turbo Assembler), yang diajarkan dalam mata kuliah Dasar Sistem Komputer, melainkan ada MASM (Macro Assembler) yang dibuat. Saya mencari contoh projek yang menggunakan bahasa assembly MASM. Saya mencoba untuk membuat tantangan membuat game menggunakan bahasa assembly. Saya kepikiran membuat game sederhana, yaitu game snake. Setelah di cek di forum E-learning, ternyata belum ada yang membuat aplikasi game snake menggunakan assembly di forum E-learning sehingga saya memilih topik tersebut untuk dijadikan sebagai projek tugas akhir mata kuliah Dasar Sistem Komputer. Menentukan topik judul projek aplikasi membutuhkan sekitar 3 hari untuk mendapatkan topik yang sesuai.

Setelah menemukan judul projek aplikasi yang cocok, kemudian saya membuat projek di GitHub untuk menyimpan file-file yang dipakai dalam projek saya sebagai source code. Saya juga tidak lupa untuk mengundang dosen saya, Bapak Ali Tarmuji, S.T., M.Cs sebagai collaborator untuk memantau projek saya. Membuat projek di GitHub tidak memerlukan waktu yang lama, hanya memerlukan sekitar 1 hari untuk membuat dan menyertakan bukti bahwa projek di GitHub sudah dibuat di E-learning. Setelah wadah untuk menyimpan kode sudah dibuat (projek di GitHub), kemudian saya mengintegrasikan dengan git yang sudah memiliki fitur di Visual Studio. Setelah itu, saya juga mencari library yang cocok digunakan pada bahasa Assembly bertipe MASM. Salah satu library yang saya temukan adalah library Irvine yang cocok digunakan pada bahasa Assembly bertipe MASM. Karena judul yang sudah saya pilih sudah fix, maka saya akan langsung juga membuat kode untuk membuat aplikasi Ulo Asem.

Setelah menemukan judul projek aplikasi yang cocok, kemudian saya membuat projek di GitHub untuk menyimpan file-file yang dipakai dalam projek saya sebagai source code. Tidak lupa saya mengundang dosen saya, Bapak Ali Tarmuji, S.T., M.Cs sebagai collaborator untuk memantau projek saya. Membuat projek di GitHub tidak memerlukan waktu yang lama, hanya memerlukan 1 hari untuk membuat dan penyertaan bukti dimana projek di GitHub sudah dibuat di E-learning.

Setelah wadah untuk menyimpan kode sudah dibuat (projek di GitHub), kemudian saya mengintegrasikan dengan Git yang sudah ada fitur di Visual Studio. Setelah itu saya juga mencari library yang cocok digunakan pada bahasa Assembly bertipekan MASM. Salah satu library yang saya temukan adalah library Irvine yang cocok digunakan bahasa assembly bertipe MASM. Karena judul yang sudah saya pilih sudah fix, maka saya akan langsung juga membuat kode untuk membuat aplikasi Ulo Asem. Saya membuat kerangka untuk membuat game snake sekitar 2 minggu, dan finalisasi sekitar 4 hari. Saat finalisasi tersebut, saya menambah fitur seperti memberi warna, hingga debugging apabila terjadi bug, berlangsung bersamaan dengan fase uji coba aplikasi.

Selanjutnya, setelah aplikasi selesai dibuat, saya langsung membuat laporan aplikasi Ulo Asem selama 1 hari full, dan pada hari itu juga 1 jam 30 menit disisihkan untuk melakukan presentasi aplikasi Ulo Asem. Presentasi tersebut di upload ke dalam YouTube selama 1 jam dikarenakan file yang cukup besar dan kecepatan internet yang tidak terlalu cepat.

Terakhir, untuk pembuatan portofolio aplikasi diperkirakan selesai selama 4 hari. Saya memulai pembuatan portofolio aplikasi dan membuat pilihan waktu saat kosong di saat Ujian Akhir Semester 1.

1. **Ketercapaian Spesifikasi**

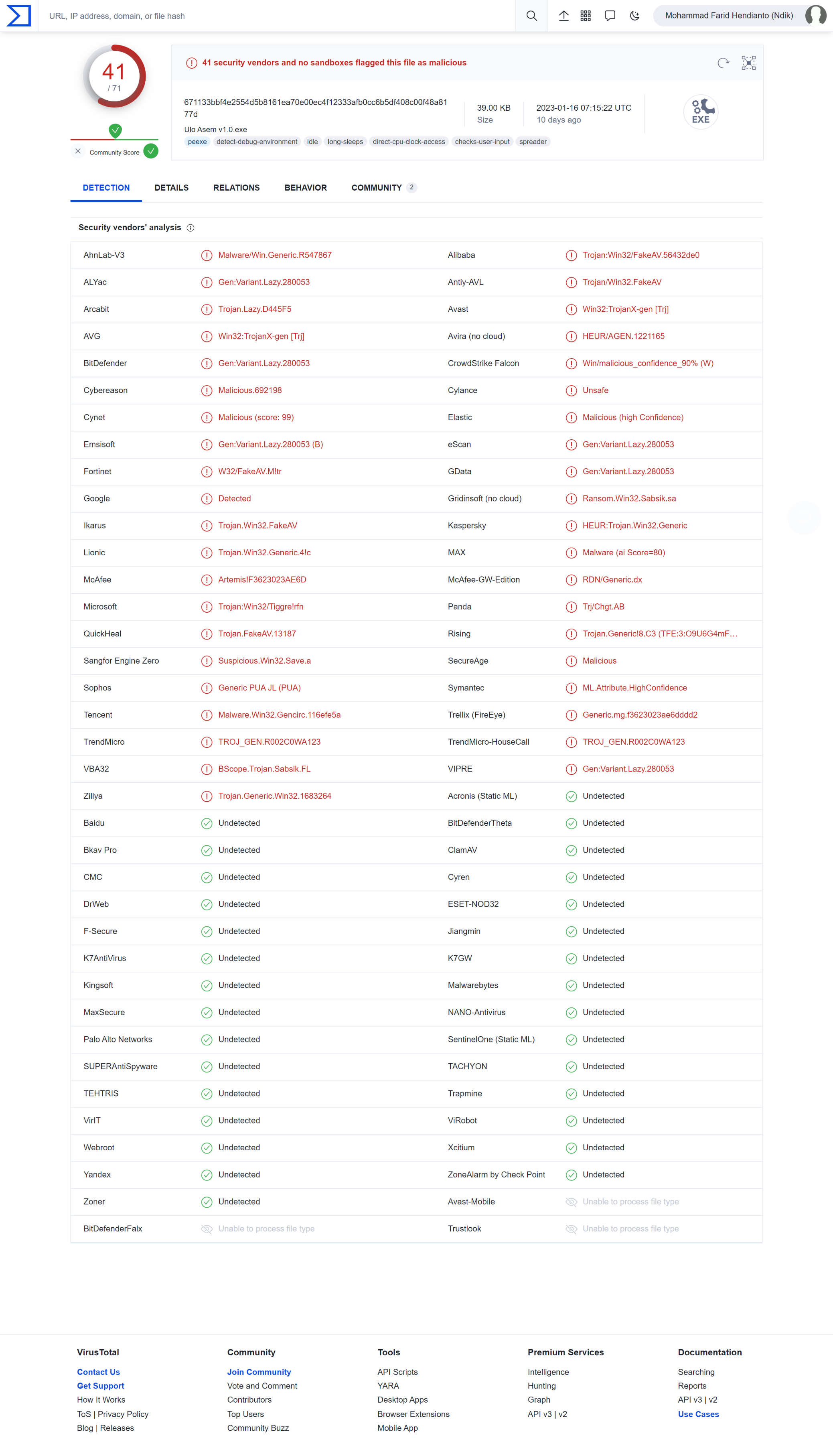
Untuk dapat menjalankan aplikasi game Ulo Asem, diperlukan spesifikasi sistem yang minimal sebagai berikut: sistem operasi Windows XP 32 bit atau yang lebih tinggi, prosessor 512 Mhz atau lebih tinggi, memory 32 MB atau lebih tinggi, ukuran file 39 KB, storage 1 MB ruang hardisk yang tersedia, sound card yang terdapat untuk menjalankan audio, dan graphic card yang mendukung penampilan warna. Pastikan bahwa semua perangkat lunak yang diperlukan, seperti sistem operasi dan aplikasi pendukung, telah terinstal dengan benar pada komputer Anda agar game Ulo Asem dapat berjalan dengan lancar. Selain spesifikasi sistem yang diperlukan, aplikasi game Ulo Asem juga dilengkapi dengan beberapa fitur yang akan membuat pengalaman bermain game semakin menyenangkan bagi pengguna. Fitur-fitur tersebut antara lain tampilan dinding, papan skor, pemilihan kecepatan Ulo, pembuatan acak apel, dan pergerakan Ulo yang dapat dikontrol dengan keyboard. Tampilan dinding pada aplikasi ini akan membuat permainan semakin menyenangkan bagi pengguna, sementara fitur pemilihan kecepatan Ulo akan memungkinkan pengguna untuk mengatur kecepatan gerakan ular sesuai dengan keinginan mereka. Pembuatan acak apel juga merupakan fitur yang akan membuat permainan semakin menantang dan tidak monoton. Pergerakan Ulo yang dapat dikontrol dengan keyboard akan membuat permainan semakin mudah dimainkan dan dikendalikan oleh pengguna. Fitur reset juga merupakan fitur yang dapat digunakan oleh pengguna untuk mengulang permainan dari awal. Keseluruhan, aplikasi game Ulo Asem ini telah dirancang dengan baik untuk dapat digunakan pada sistem operasi Windows dan ditulis dengan menggunakan bahasa Assembly. Spesifikasi system yang dibutuhkan untuk menjalankan aplikasi ini cukup rendah, sehingga dapat digunakan pada komputer dengan spesifikasi yang rendah. Selain itu, game ini juga dilengkapi dengan fitur-fitur yang menarik seperti grafis yang baik, suara yang menyenangkan, dan kontrol yang mudah digunakan. Meskipun game ini dirancang untuk sistem operasi Windows, aplikasi ini juga dapat digunakan pada sistem operasi lain dengan menggunakan software penerjemah. Aplikasi ini juga dilengkapi dengan dokumentasi yang lengkap yang memudahkan pengguna dalam menggunakannya.

1. **Biaya yang dibutuhkan**

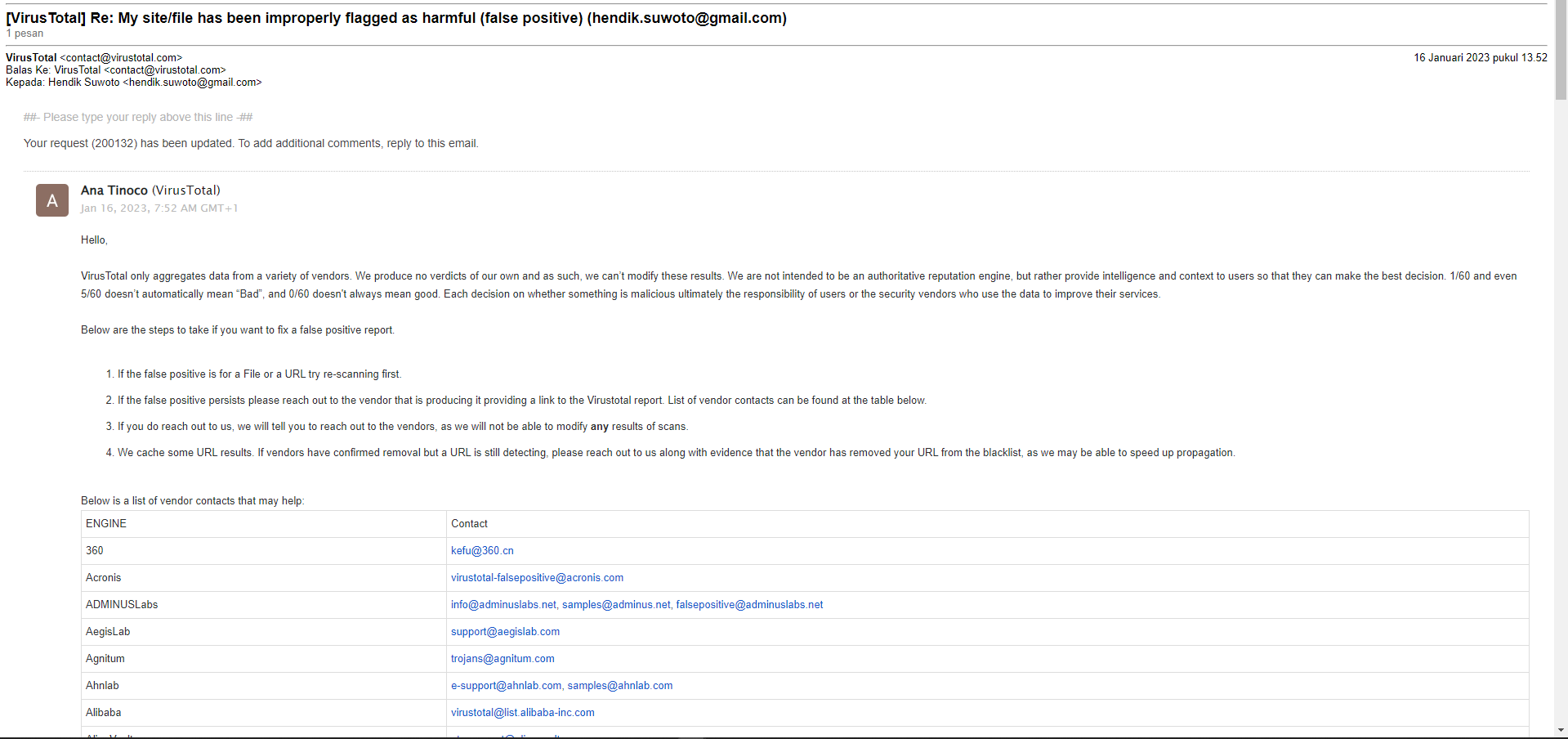
Biaya yang dibutuhkan dalam pengembangan aplikasi Ulo Asem sangat minim atau bahkan tidak ada sama sekali. Hal ini dikarenakan saya menggunakan Visual Studio 2022 Insiders Community Edition, yang merupakan perangkat lunak gratis dan tersedia untuk digunakan tanpa biaya. Selain itu, saya juga menggunakan library Irvine yang merupakan open source yang tidak berbayar.

Biaya yang dibutuhkan hanyalah listrik, wifi, makanan untuk memberikan semangat motivasi dan waktu karena waktu adalah uang. Namun, saya berharap bahwa dengan pengembangan aplikasi ini, akan ada potensi pendapatan yang akan menutupi biaya-biaya tersebut. Saya berusaha untuk menjaga biaya pengembangan aplikasi ini serendah mungkin agar dapat digunakan oleh seluruh masyarakat tanpa harus khawatir tentang biaya yang dibutuhkan.

1. **Kendala yang dihadapi**



Gambar 45 Aplikasi Ulo Asem yang terdeteksi virus pada Virustotal **(Sumber: Penulis)**



Gambar 46 Bertanya Customer Service pada Virustotal untuk melaporkan false detection terhadap aplikasi Ulo Asem **(Sumber: Penulis)**

Salah satu kendala yang dihadapi oleh aplikasi Ulo Asem adalah diterdeteksi oleh virustotal sebanyak 41 dari 71 provider. Hal ini mungkin dikarenakan aplikasi ini menggunakan library dari windows seperti winmm, dan menggunakan interrupt processor secara langsung dengan bahasa assembly. Bahasa assembly merupakan bahasa pemrograman yang jarang dipelajari karena lebih detail dan memerlukan pemahaman yang lebih dalam mengenai sistem operasi dan arsitektur komputer. Selain itu, bahasa assembly juga memerlukan lebih banyak waktu dan usaha untuk dapat membuat program yang efisien dan stabil.

1. **Tantangan Masa Depan**

Tantangan masa depan dalam membuat portofolio aplikasi Ulo Asem adalah menghilangkan false detection yang dapat muncul dari customer service Virus Total. Hal ini dapat dilakukan dengan cara re-scan dan menghubungi vendor yang menghasilkan false detection, serta menyediakan link dari laporan Virustotal. Selain itu, untuk meningkatkan pengalaman bermain dan menarik minat pengguna, akan ditambahkan fitur-fitur baru seperti mode frenzy (apel lebih banyak muncul), mode banyak rintangan seperti batu, fitur lainnya seperti memberikan warna pada ular, tampilan menu, tampilan option, grafis yang lebih memukau, dan maintance projek yang lebih lanjut. Namun, diperlukan kerja sama dengan vendor dan komunitas untuk menjamin keamanan dan kualitas aplikasi ini.