

IMPLEMENTASI PYTHON DALAM PENGOLAHAN DATA PRIBADI MAHASISWA ILMU KOMPUTER ANGKATAN 23 PADA UNIVERSITAS NEGERI MEDAN MENGGUNAKAN STRUKTUR DATA *LINKED LIST*

Muhammad Rizki Andrian Fitra, Ali Afr Rahman S. Effendi, Fanny Ramadhani

Ilmu Komputer, Universitas Negeri Medan

Jalan William Iskandar Pasar V, Medan, Sumatera Utara, Indonesia

andrian25544.4232550001@mhs.unimed.ac.id

ABSTRAK

Kemajuan teknologi dalam dunia komputasi digital yang semakin terus berkembang efisiensi dalam pengelolaan data menjadi kebutuhan yang sangat penting, sehingga dibutuhkan sebuah struktur data yang tepat untuk proses dalam mengelola data berjalan dengan efisien seperti struktur data *Linked list*. Proses pengelolaan data mahasiswa sering kali menghadapi tantangan seperti keterbatasan fleksibilitas dalam struktur penyimpanan data dan kompleksitas dalam menambah, mencari, atau menghapus data. Penelitian ini bertujuan untuk mengembangkan sistem pengelolaan data berbasis GUI dengan memanfaatkan *Linked list* untuk menambah, mencari, dan menghapus data menjadi lebih mudah. Dengan metode pengujian sistem aplikasi GUI dengan menggunakan struktur data *Linked list* dalam pengelolaan data seperti fungsi tambah, cari, dan hapus data. Hasil uji sistem aplikasi pada penelitian ini menunjukkan bahwa implementasi struktur data *Linked list* memungkinkan untuk pengelolaan data yang lebih fleksibel dan efisien. Penelitian ini diharapkan dapat memberikan sebuah wawasan bagi pengembang perangkat lunak yang ingin membangun sistem pengelolaan data yang aman, dan efisien.

Kata kunci : Struktur data, linked list, tkinter, python, mahasiswa

1. PENDAHULUAN

Dalam dunia komputasi dan dalam era teknologi digital yang semakin terus berkembang, struktur data memegang peranan penting dalam mempengaruhi efisiensi dan performa berbagai aplikasi dan sistem. Salah satu struktur data yang sering digunakan adalah linked list [1]. Manajemen memori merupakan aspek penting dalam pemrograman, yang pasti dimana programmer perlu mengalokasikan dan membebaskan memori secara efisien untuk menghindari pemborosan sumber daya dan menjaga performa program. Maka, dalam hal ini, pemilihan struktur data yang tepat memiliki peran yang sangat krusial. Terdapat dua struktur data yang umum digunakan untuk menyimpan data secara berurutan adalah struktur data *array* dan juga struktur data *Linked list* [2]. Oleh karena itu, biasanya *Linked list* memungkinkan alokasi memori secara dinamis saat program berjalan. Tidak seperti *array* yang memiliki ukuran tetap, struktur data *Linked list* dapat menambah atau menghapus elemen tanpa harus memindahkan elemen lainnya atau menentukan ukuran awal.

Linked list biasanya terdiri dari beberapa *node-node* (simpul-simpul) yang saling terhubung atau yang biasa disebut *linked*. Simpul berupa *struct*, sedangkan *link* adalah sebuah komponen yang bertipe pointer yang terletak kedalam simpul [3]. Pada penelitian terdahulu yang berjudul *Linked List Data Structure and Library Management System* oleh Nayana Joshi et al. Mengatakan bahwa dengan menerapkan *Linked list* kedalam sistem manajemen perpustakaan, akan bisa memudahkan untuk menambah, menghapus, dan mencari buku atau materi yang diinginkan oleh pembaca yang terletak pada perpustakaan [4].

Oleh karena itu, penelitian ini bertujuan dan berfokus pada pengimplementasian struktur data *Linked list* untuk mengelola data, yang dimana data yang akan dikelola merupakan data pribadi mahasiswa prodi Ilmu Komputer angkatan 2023 di Universitas Negeri Medan yang diperoleh. Serta pada jurnal ini mengeksplorasi manfaat dan masalah yang akan dihadapi dalam proses pengelolaan data mahasiswa. Dengan tindakan menyoroti langkah-langkah teknis, pengaturan sistem, dan praktik yang terbaik dalam penggunaan struktur data *Linked list*, diharapkan jurnal ini dapat memberikan wawasan-wawasan yang akan berguna bagi siapapun yang ingin memanfaatkan teknologi pemrograman Python untuk pengelolaan data yang efisien, dan aman dengan menggunakan struktur data *Linked list*.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Banjarnahor [5] membahas tentang bagaimana menggunakan dari berbagai model penyimpanan data, termasuk penyimpanan kolom, kunci atau nilai, dan grafik, serta konsep dasar database dan struktur data seperti *Linked list*. Penggunaan *Linked list* karena data dapat disimpan secara dinamis, memungkinkan penggunaan memori dapat dilepas sesuai kebutuhan aplikasi. Penelitian yang dilakukan oleh Sara [6] mampu untuk mengembangkan sebuah struktur data yang bisa memberikan kinerja yang lebih baik dalam operasi penambahan data, penyisipan data, dan penghapusan data pada aplikasi dengan menggunakan struktur data *Hybrid Array List* yang menggabungkan elemen dari *Linked list* dan *array*. Pada penelitian yang dilakukan oleh Roihan [7] menggunakan struktur data *array object* dan data base untuk membuat sebuah

aplikasi mobile yang dapat mengelola data mahasiswa secara lebih efektif dengan menguji pengaruh sistem pengolahan data secara real-time. Nopriyansah [8] mengembangkan sebuah aplikasi *service* data mahasiswa pada website Jurusan Ilmu Komputer di Universitas Lampung, yang dimana dalam database mencakup nama parameter, dan berbagai data yang terkait dengan dosen, mahasiswa dan jadwal skripsi di Universitas Lampung. Dan pada penelitian Revita [9] menyoroti analisis dan perancangan sistem pengarsipan perpustakaan berbasis GUI untuk mengelola perpustakaan, termasuk data peminjaman, buku dan peminjaman. Data pada penelitian diorganisir dalam fitur - fitur terpisah seperti Fitur Buku, Fitur Jenis Buku, dan Fitur Peminjaman yang memudahkan pengelompokan dan pengolahan data mirip seperti struktur data *Linked list*.

2.2. Struktur Data

Istilah stuktur data merupakan cara penyimpanan penyusunan dan pengaturan data yang terletak di dalam media penyimpanan Sedangkan dalam istilah pemrograman struktur data yang mengartikan tata letak suatu data yang berisi kolom kolom. Struktur data adalah suatu model logika yang secara khusus mengorganisasi data. Struktur data juga dapat di pahami bagaimana data dapat disimpan. Dalam pemahaman struktur data dikenal dengan struktur data statis yaitu struktur data tidak berubah dan satu lagi merupakan struktur data dinamis yaitu struktur data yang berubah [10].

Struktur data adalah sebuah cara khusus untuk mengorganisasi, menyimpan, serta mengatur data pada sebuah perangkat teknologi seperti komputer agar data tersebut dapat diakses dan dimodifikasi secara efisien oleh pengguna [11].

2.3. Linked List

Linked list adalah sebuah strukur data yang linier berbentuk rantai simpul yang di mana setiap simpul menyimpan 2 item, yaitu nilai data dan pointer ke simpul elemen berikutnya. Berbeda dengan *array*, elemen *Linked list* tidak ditempatkan kedalam alamat memori yang berdekatan melainkan elemen ditautkan menggunakan pointer [12].

Linked list memiliki beberapa ciri - ciri tertentu seperti data utama dalam suatu format tertentu dapat mempermudah sebuah operasi seperti penyimpanan, pencarian dan manipulasi data. Struktur data *Linked list* juga memiliki sebuah fitur operasi seperti penambahan data, penghapusan atau pencarian data dengan mudah.

2.4. Python

Salah satu bahasa pemrograman didunia yang berkembang saat ini yaitu Python, Python adalah salah satu bahasa pemrograman yang biasa menggunakan interpreter dalam kode programnya. Interpreter dapat menerjemahkan kode secara langsung, dan Python dapat dijalankan didalam berbagai platform, seperti Windows, Linux, dan lainnya [13].

Bahasa pemrograman Python memiliki struktur bahasa yang sederhana dan mudah dipelajari, karena kelebihan ini yang telah membuat Python populer diberbagai kalangan. Bahasa pemrograman Python memiliki *library* yang banyak, tersedia beberapa modul yang siap digunakan untuk berbagai keperluan, dan memiliki sistem pengelolaan memori otomatis [14].

2.5. Tkinter

Tkinter adalah salah satu alat yang biasanya akan digunakan untuk menghubungkan grafis pada bahasa pemrograman Python. Tkinter merupakan salah satu library Python yang biasanya paling sering digunakan untuk mengembangkan suatu atau merancang suatu GUI (Graphical User Interface) [15].

Tkinter merupakan suatu metode yang paling umum digunakan karena modul ini cukup mudah dipakai untuk mempelajari dasar - dasar GUI atau yang biasa disebut dengan kalimat *Graphic User Interface* [16].

3. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini menggunakan beberapa tahapan penting dalam proses seperti penginputan data yang sudah didapatkan, konfigurasi program, dan pengujian sistem Python dengan menggunakan visual studio code.

3.1. Subjek Penelitian

Subjek Penelitian pada penelitian ini, adalah mahasiswa Ilmu Komputer angkatan 2023 di Universitas Negeri Medan. Jumlah subjek penelitian pada penelitian ini berjumlah 5 orang dari program studi Ilmu Komputer angkatan 2023, data yang diambil dari subjek penelitian adalah berupa data mahasiswa baik itu data pribadi ataupun data seputar universitas yang diperoleh melalui google form.

3.2. Tabel Data Pribadi Mahasiswa

Tabel 1. Tabel data pribadi mahasiswa

Nama Mahasiswa	Jenis Kelamin	Kelas	Tanggal Lahir	Cita-Cita
Oka Alvansyah	Laki-laki	PSIK23B	10 November 2004	Programmer
Josua Anugrah Deo Tampubolon	Laki-laki	PSIK23B	30 Oktober 2003	Polisi Siber
Atta Zul Fahrizan	Laki-laki	PSIK23D	24 Desember 2005	Data Analyst
Sybil Auzi	Laki-laki	PSIK23A	04 April 2005	Programmer
Pedro Stella Mario Meyar Waruwu	Laki-laki	PSIK23B	08 Maret 2006	Polisi Siber

Dari Tabel 1 dapat dilihat bahwa terdapat beberapa data pribadi mahasiswa yang didapatkan

melalui form yang dikirimkan secara online melalui aplikasi google form. Data yang telah didapatkan dari google form merupakan data pribadi dari mahasiswa, seperti nama lengkap mahasiswa, jenis kelamin mahasiswa, kelas mahasiswa, tanggal lahir mahasiswa dan yang terakhir adalah cita - cita mahasiswa Ilmu komputer angkatan 2023 Universitas Negeri Medan.

3.3. Tabel Data Universitas Mahasiswa

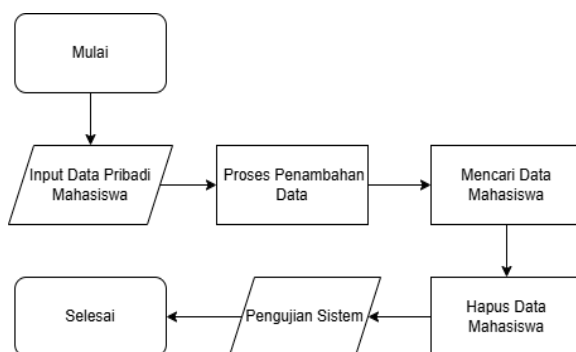
Tabel 2. Tabel data universitas mahasiswa

Nama Mahasiswa	NIM	Jalur Masuk	IPK Terakhir
Oka Alvansyah	4233250036	SNBT	3,57
Josua Anugrah Deo Tampubolon	4233550036	MANDIRI	3,20
Atta Zul Fahrizan	4233250021	SNBT	3,6
Sybil Auzi	4233250026	SNBT	3,61
Pedro Stella Mario Meyer Waruwu	4233550044	MANDIRI	3,4

Dari Tabel 2 dapat dilihat bahwa terdapat data universitas mahasiswa yang juga didapatkan melalui form yang dikirimkan secara online melalui aplikasi google form. Data yang telah didapatkan dari google form merupakan beberapa data universitas dari mahasiswa, seperti NIM atau biasa disebut dengan nomor induk mahasiswa, jalur masuk mahasiswa yang dimana jalur masuk ini merupakan sebuah jalur diterimanya mahasiswa dalam Universitas Negeri Medan, dan IPK terakhir dari mahasiswa Ilmu Komputer angkatan 2023. Kemudian, setelah data - data mahasiswa telah didapatkan dan dikumpulkan, maka proses selanjutnya dalam penelitian ini adalah dengan menyusun sebuah rangkaian atau gambaran tahapan - tahapan penelitian yang akan dilakukan oleh peneliti.

3.4. Tahapan Penelitian

Berikut adalah gambar sebuah proses dari tahapan - tahapan penelitian yang akan dilakukan dalam penelitian ini yang dapat dilihat pada Gambar 1 sebagai berikut :



Gambar 1. Alur tahapan penelitian

Pada tahap pertama penelitian akan dilakukan dengan pengumpulan data yang terdiri dari dua sumber data yaitu data pribadi mahasiswa, dan juga data universitas mahasiswa.

Pada tahap kedua pengguna kemudian akan memasukkan data pribadi mereka melalui program Python yang berbasis GUI yang dibuat dengan menggunakan *library* Python yaitu Tkinter. Pada tahap ini pengguna akan melakukan proses penambahan data pada program yang telah dibuat.

Pada tahap ketiga setelah data terkumpul, pengguna akan bisa mencari data - data yang telah mereka masukkan pada program Python berbasis GUI yang telah dibuat oleh peneliti.

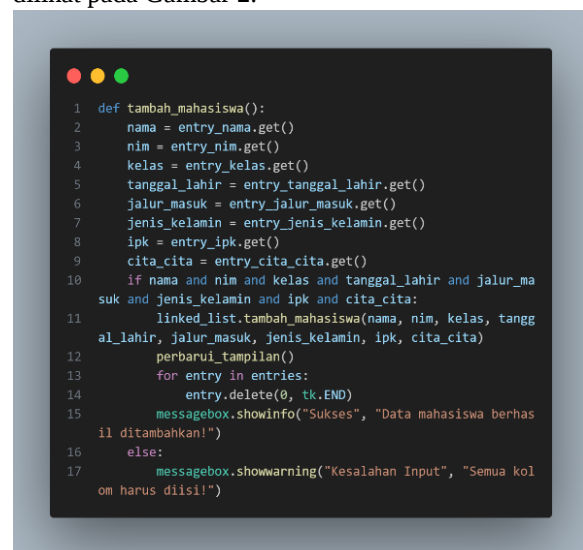
Pada tahap keempat setelah melakukan pencarian data yang ada pada program, maka pengguna akan bisa menghapus data - data yang telah dimasukkan tadi jika pengguna merasa bahwa data itu tidak diperlukan lagi dan menghindari penumpukan data pada program.

Pada tahap terakhir akan dilakukan pengujian sistem yang dilakukan pada saat memasukkan data kedalam program kemudian program akan diuji apakah data yang sudah dimasukkan atau data yang sudah ada dalam program bisa dicari dengan fitur cari data mahasiswa dan dihapus dengan fitur hapus data pada program yang telah dibuat.

Kemudian pada tahap berikutnya, akan dilakukan pembuatan fitur dalam program. Fitur yang pertama yaitu fitur untuk menambahkan data mahasiswa, fitur untuk mencari data mahasiswa, dan yang terakhir adalah fitur untuk menghapus data mahasiswa yang telah ditambahkan didalam program yang telah dibuat.

3.5. Fitur Penginputan Data

Dalam tahap ini, akan dibuat sebuah fitur untuk penginputan data - data mahasiswa. Program untuk penginputan data mahasiswa dalam program dapat dilihat pada Gambar 2.



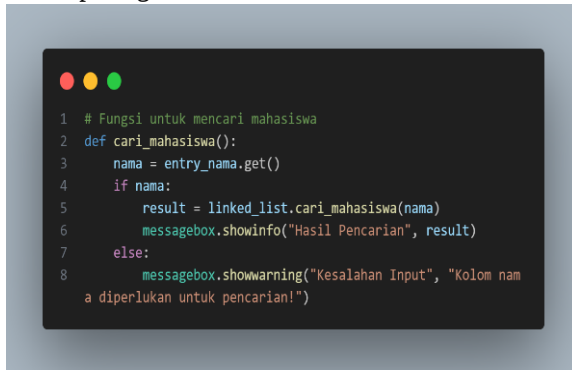
Gambar 2. Fitur input data

Pada Gambar 2 menunjukkan konfigurasi dari fungsi `tambah_mahasiswa` pada program yang dibuat untuk menambahkan data mahasiswa prodi Ilmu

Komputer angkatan 2023 Universitas Negeri Medan, yang telah diperoleh dari google form.

3.6. Fitur Mencari Data

Kemudian untuk tahap ini peneliti akan membuat sebuah fitur dalam program untuk mencari data-data mahasiswa yang telah diinput kedalam program. Program untuk mencari data - data mahasiswa dapat dilihat pada gambar 3.



Gambar 3 Fitur mencari data

Pada gambar 3 menunjukkan sebuah fungsi yang digunakan untuk membuat fitur mencari data dengan menggunakan kotak fitur pencarian yang terdapat didalam program python yang telah dibuat. Fitur ini digunakan untuk mencari data hanya dengan memasukkan nama dari data mahasiswa prodi Ilmu Komputer angkatan 2023 yang diperoleh dari google form. Pada fitur mencari data ini dibuat sebuah sistem jika nama yang dimasukkan tidak ada dan juga sebuah sistem yang akan menampilkan data mahasiswa jika nama lengkap yang diinputkan ada.

3.7. Fitur Menghapus Data

Kemudian untuk tahap ini peneliti akan membuat sebuah fitur yang ada didalam program untuk menghapus data - data mahasiswa yang telah diinput kedalam program untuk dihapus. Program untuk menghapus data - data mahasiswa dapat dilihat pada gambar 4.



Gambar 4. Fitur menghapus data

Pada Gambar 4 menunjukkan sebuah fungsi Python yang digunakan untuk membuat sebuah fitur menghapus data yang dimana fitur ini juga membuat sebuah program pada saat penghapusan data, kemudian akan muncul sebuah kotak format untuk konfirmasi terlebih dahulu sebelum menghapus data mahasiswa prodi Ilmu Komputer angkatan 2023 Universitas Negeri Medan yang diperoleh melalui google form.

4. HASIL DAN PEMBAHASAN

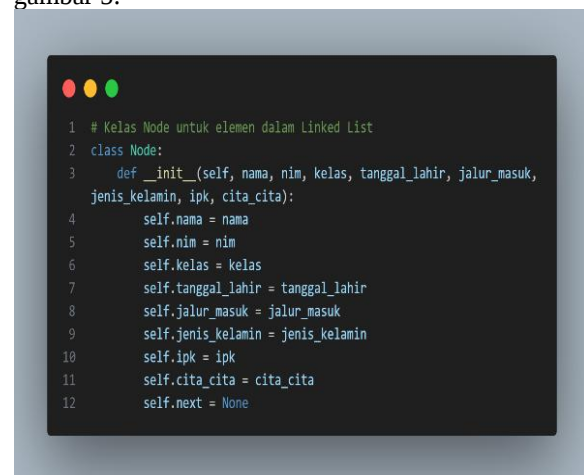
Hasil dari penelitian ini akan menunjukkan bagaimana struktur data *linked list* bisa diterapkan untuk mengelola suatu data yang ada, baik itu data pribadi ataupun data umum mahasiswa. Dalam implementasi struktur data *linked list* didalam pengelolaan data mahasiswa memberikan beberapa keunggulan dibandingkan struktur data statis seperti *array*. Dengan struktur data *linked list*, data mahasiswa dapat ditambahkan atau dihapus tanpa perlu memindahkan elemen lainnya, menjadikannya lebih efisien untuk operasi dinamis. Setiap *node - node* yang didalam *linked list* merepresentasikan satu mahasiswa, dengan atribut seperti nama, NIM, kelas, dan sebagainya sebagai data dalam *node* tersebut.

Pada program yang telah dibuat menggunakan bahasa pemrograman yaitu Python yang kemudian menggunakan *library Tkinter* untuk membuat sebuah GUI dari program dibuat untuk mempermudah serta memberikan sebuah kesan yang menarik ketika pengguna memasukkan data kedalam program yang telah dibuat pada penelitian ini. Program ini dapat dijalankan ketika pengguna telah memasukkan data kedalam program dengan struktur data *Linked list*.

Sebelum memasukkan sebuah data mahasiswa yang pertama perlu dilakukan adalah dengan membuat sebuah fungsi *node* untuk mendefinisikan data - data.

4.1. Pembuatan Node *Linked list*

Sebelum melakukan pembuatan program dengan menggunakan GUI, akan dibuat sebuah fungsi yang akan mendefinisikan sebuah *node - node* yang akan digunakan untuk membuat elemen - elemen dalam struktur data *Linked list*, yang dapat dilihat pada gambar 5.



Gambar 5. Node linked list

Pada Gambar 5 menunjukkan sebuah program yang mendefinisikan sebuah kelas bernama *node*, kelas *node* bisa dibayangkan seperti kotak yang menyimpan informasi tertentu. Kelas *node* digunakan untuk membuat elemen - elemen dalam struktur data *Linked list*. Kelas *node* berfungsi untuk menyimpan data - data pribadi mahasiswa prodi Ilmu komputer angkatan 2023. Setiap elemen pada *Linked list* akan diwakili oleh sebuah *node*, dimana setiap *node* menyimpan data tertentu dan memiliki petunjuk atau yang biasa disebut *pointer* ke *node* berikutnya.

4.2. Pembuatan Kelas *Linked list*

Setelah mendefinisikan sebuah elemen - elemen yang diperlukan dalam membuat program, kemudian akan dilakukan pembuatan kelas *Linked list*. Kelas *Linked list* berfungsi untuk mengelola data mahasiswa menggunakan struktur data ini. Kelas *Linked list* pada program ini memiliki tiga fungsi utama ; yang pertama yaitu menambah, menghapus, dan mencari data mahasiswa dalam *Linked list*. Pada pembuatan kelas *Linked list* dibuat tiga definisi yang digunakan untuk mengelola data mahasiswa prodi Ilmu Komputer angkatan 2023 yang diperoleh melalui Google Form, yaitu *def_tambah_mahasiswa* untuk menambahkan data mahasiswa, kemudian *def_hapus_mahasiswa* untuk menghapus data mahasiswa yang telah ditambahkan kedalam program, dan yang terakhir *def_cari_mahasiswa* yang dimana fungsinya untuk mencari data mahasiswa yang diinginkan oleh pengguna. Jika *node* dengan nama yang dicari tidak dapat ditemukan, fungsi akan mengembalikan program dan akan menampilkan Data tidak ditemukan.

4.3. Tampilan GUI Program *Linked list*

The screenshot shows a web application titled 'Pengelolaan Data Pribadi Mahasiswa'. It features a form with the following fields: Nama, NIM, Kelas, Tanggal Lahir (DD/MM/YYYY), Jalur Masuk Universitas, Jenis Kelamin, IPK Terakhir, and Cita-Cita. Below the form are three buttons: 'Tambah Mahasiswa' (green), 'Hapus Mahasiswa' (red), and 'Cari Mahasiswa' (blue). At the bottom, there is a large empty rectangular box for displaying results.

Gambar 6. GUI pengelolaan data mahasiswa

Pada tahap ini hal yang harus dilakukan berikutnya adalah dengan membuat *node Linked list* terlebih dahulu, kemudian setelah mendefinisikan sebuah kelas *node* pada program, maka hal yang perlu dilakukan selanjutnya ada dengan membuat kelas *Linked list* dengan mendefinisikan fungsi *def*

tambah_mahasiswa, kemudian ada yang namanya fungsi *def_hapus_mahasiswa*, dan yang terakhir adalah fungsi *def_cari_mahasiswa*. Setelah mendefinisikan semua fungsi yang ada maka hal selanjutnya adalah dengan membuat sebuah GUI untuk membuat program menjadi lebih menarik dan interaktif yang dapat dilihat pada gambar 6.

Pada Gambar 6. bisa dilihat sebuah tampilan antar pengguna atau yang biasa disebut dengan *Graphic User Interface (GUI)* yang menggunakan struktur data *Linked list* untuk mengelola data mahasiswa baik itu data pribadi ataupun data universitas. Pada program dibuat sebuah tombol ataupun fitur yang berfungsi untuk menambahkan data mahasiswa, menghapus data mahasiswa, dan mencari data mahasiswa.

4.4. Uji Menambahkan Data Mahasiswa

Pada tahap ini, pengguna akan mencoba untuk menambahkan data yang telah diperoleh yang dapat dilihat pada Gambar 7.

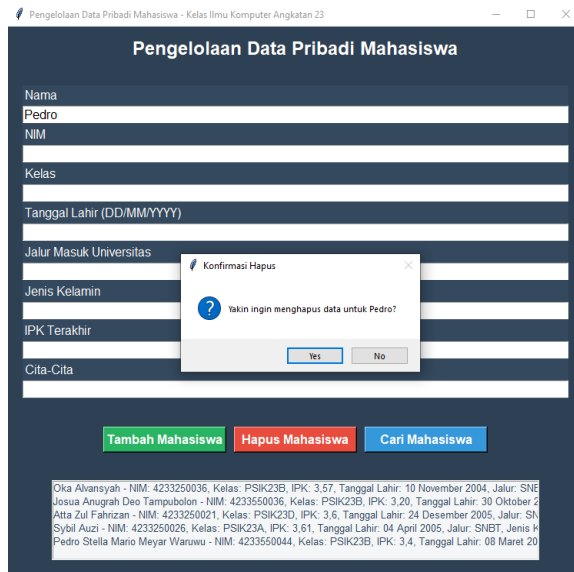
This screenshot shows the same GUI as Gambar 6, but with the 'Tambah Mahasiswa' button highlighted in green. Below the buttons, a list of student data is displayed, including names, NIMs, classes, IPKs, and birth dates.

Gambar 7. Uji penambahan data

Pada gambar 7. dapat dilihat untuk menambahkan data pada program, pengguna harus mengisi Nama, NIM, Kelas, Tanggal Lahir, Jalur Masuk Universitas, Jenis Kelamin, IPK Terakhir, dan Cita - cita untuk menambahkan data yang telah diperoleh melalui Google Form, kemudian pilih tombol Tambah Mahasiswa untuk menambahkan data yang ada.

4.5. Uji Menghapus Data Mahasiswa

Pada tahap ini, pengguna akan mencoba untuk menghapus data yang ada pada program yang dapat dilihat pada Gambar 8.

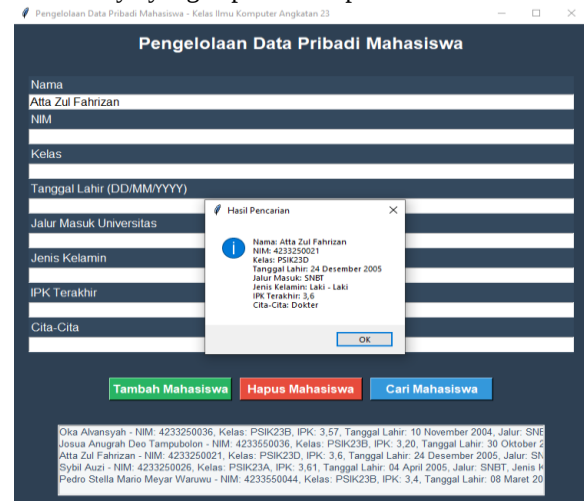


Gambar 8. Uji penghapusan data

Pada Gambar 8. dapat dilihat sebuah pengujian penghapusan data dilakukan dengan cara memasukkan nama mahasiswa yang telah berhasil ditambahkan sebelumnya, kemudian jika nama yang dicari pada program berhasil ditemukan maka akan menampilkan sebuah text box yang ingin memastikan pengguna apakah akan menghapus data tersebut atau tidak.

4.6. Uji Mencari Data Mahasiswa

Pada tahap uji terakhir, pengguna akan mencoba mencari data yang telah ditambahkan pada program sebelumnya yang dapat dilihat pada Gambar 9 :



Gambar 9. Uji mencari data

Pada Gambar 9 merupakan tahap pengujian mencari data mahasiswa yang dapat dijalankan pada program dengan memasukkan nama mahasiswa yang ingin dicari data - datanya pada kolom nama, jika nama mahasiswa yang dicari pada program tidak ada maka akan muncul sebuah notifikasi bahwa data tidak dapat ditemukan. Sebaliknya, jika data berhasil ditemukan maka program akan menampilkan data - data yang ada dari nama mahasiswa yang dicari.

4.7. Hasil Pengujian Aplikasi

Tabel 3. Hasil pengujian penambahan data

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Data mahasiswa akan diisi sesuai dengan kolom yang ada	Input Nama : Atta Zul Fahrizan , NIM : 4233250021 , Kelas : PSIK23D , Tanggal Lahir : 24 Desember 2005 , Jalur Masuk Universitas : UTBK , Jenis Kelamin : Laki - laki , IPK Terakhir : 3,6 , Cita - cita : Data Analyst	Aplikasi akan menampilkan bahwa data telah berhasil ditambahkan	Sesuai harapan	Valid
2	Data mahasiswa akan diisi hanya sebagian saja pada kolom	Input Nama : Atta Zul Fahrizan , NIM : 4233250021 , Kelas : PSIK23D , Tanggal Lahir : 24 Desember 2005 ,	Aplikasi akan menampilkan sebuah pemberitahuan bahwa semua kolom harus diisi	Sesuai harapan	Valid

Dari tabel 3 dapat dilihat bahwa hasil pengujian penambahan data pada aplikasi akan berhasil jika data telah diinputkan sesuai dengan kolom yang ada maka data akan berhasil ditambahkan, jika data tidak ditambahkan sesuai dengan jumlah

kolom yang ada maka aplikasi akan memberikan notifikasi bahwa kolom pada aplikasi harus diisi semua.

Tabel 4. Hasil pengujian menghapus data

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Data mahasiswa akan dihapus dengan mengisi kolom nama saja	Input Nama : Atta Zul Fahrizan	Aplikasi akan menampilkan pemberitahuan bahwa data telah berhasil dihapus	Sesuai harapan	Valid

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
2	Data mahasiswa akan dihapus dengan kolom yang lain	Input NIM : 4233250021	Aplikasi akan menampilkan sebuah pemberitahuan bahwa kolom nama harus diisi	Sesuai harapan	Valid

Pada tabel 4 dapat dilihat bahwa pengujian menghapus data mahasiswa berhasil dilakukan, pada pengujian ini pengguna harus mengisi kolom nama pada aplikasi untuk menghapus data yang diinginkan,

tetapi jika pengguna menghapus data melalui kolom NIM atau kolom lain maka aplikasi akan menampilkan pemberitahuan bahwa kolom nama saja yang perlu diisi untuk melakukan penghapusan data mahasiswa.

Tabel 5. Hasil pengujian mencari data

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Data mahasiswa akan dicari pada kolom nama saja	Input Nama : Atta Zul Fahrizan	Aplikasi akan menampilkan data telah diisi sebelumnya	Sesuai harapan	Valid
2	Data mahasiswa akan dicari pada kolom yang lain	Input Kelas : PSIK23D	Aplikasi akan menampilkan sebuah pemberitahuan bahwa "kolom nama harus diisi"	Sesuai harapan	Valid

Dari tabel 5 dapat dilihat bahwa hasil pengujian pencarian data pada program berhasil dilakukan, pada pengujian ini sama seperti dengan pengujian menghapus data yang dimana pengguna hanya perlu mengisi kolom nama saja untuk mencari data yang ingin dicari, jika pengguna mengisi pada kolom lain untuk mencari sebuah data maka aplikasi akan memberikan sebuah pemberitahuan hanya kolom nama saja yang harus diisi.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil uji coba, data yang telah berhasil ditambahkan, dihapus, dan dicari pada program yang menggunakan Python dan *library* Tkinter bisa dijalankan dengan baik. Hal ini telah menunjukkan bahwa pengimplementasian bahasa pemrograman Python dengan menggunakan struktur data *Linked list* menghasilkan konfigurasi dari program telah berfungsi dengan baik dan efisien dalam pengelolaan data pribadi ataupun data universitas mahasiswa. Implementasi ini memberikan solusi untuk membuat suatu penyimpanan pada memori dalam mengelola data yang aman, efisien, dan terkontrol secara mandiri tanpa memerlukan banyak nya aplikasi atau *Software* yang digunakan. Dengan menggunakan struktur data *Linked list*, program ini dapat dijalankan dengan stabil dan aman di perangkat komputer pribadi, agar memaksimalkan kontrol atas data dan privasi yang ada. Diharapkan untuk penelitian selanjutnya untuk melakukan evaluasi dan pengujian berkala terhadap sistem yang telah diimplementasikan dan untuk mengelola data besar.

DAFTAR PUSTAKA

- [1] A. Wijoyo, A. Ramdhani, P. Afra, A. Salsabila, and K. Nife, "Evaluasi Efisiensi Struktur Data Linked List pada Implementasi Sistem Antrian," *J. Ris. Inform. dan Inov.*, vol. 1, no. 12, pp. 1244–1246, 2024.
- [2] A. Wijoyo, A. Azzahra, D. Nabila, and F. Silviana, "Perbandingan Struktur Data Linked List Dan Array Dalam Manajemen Memori," vol. 1, no. 12, pp. 1290–1293, 2024.
- [3] J. Sihombing, "PENERAPAN STACK DAN QUEUE PADA ARRAY DAN LINKED LIST DALAM JAVA Johnson," *Penerbit Mega Press*, vol. 7, no. 2, pp. 15–24, 2023, [Online]. Available: <https://download.garuda.kemdikbud.go.id/article.php?article=3056169&val=27825&title=PENERAPAN STACK DAN QUEUE PADA ARRAY DAN LINKED LIST DALAM JAVA>
- [4] Nayana Joshi, Nisha Satpute, Komal Walgude, Dhanashri Korpade, and Eeshwari Ransing, "Linked List Data Structure and Library Management System," *Int. Res. J. Adv. Eng. Manag.*, vol. 2, no. 05, pp. 1515–1521, 2024, doi: 10.47392/irjaem.2024.0205.
- [5] J. Banjarnahor, "Pemanfaatan Link List Untuk Mengatasi database Tidak Normal," *LOFIAN J. Teknol. Inf. dan Komun.*, vol. 2, no. 1, pp. 16–23, 2022, doi: 10.58918/lofian.v2i1.183.
- [6] M. Rasmi Abu Sara, M. F. J. Klaib, and M. Hasan, "Hybrid Array List: An Efficient Dynamic Array with Linked List Structure," *Indones. J. Comput.*, vol. 5, no. 3, pp. 47–62, 2020, doi: 10.34818/indoic.2021.5.3.527.
- [7] A. Roihan, A. A. Wisanto, Y. Sulaeman, F. M. Nur, and S. Williandi, "Implementasi Metode Realtime, Live Data Dan Parsing JSON Berbasis Mobile Dengan Menggunakan Android Studio Dan PHP Native," *J. Teknol. Inf.*, vol. 5, no. 2, 2019.
- [8] M. Nopriyansah, D. Sakethi, and D. Kurniawan, "Pengembangan Application Service Data Mahasiswa Pada Website Jurusan Ilmu Komputer," *J. Pepadun*, vol. 2, no. 1, pp. 81–89, 2021, doi: 10.23960/pepadun.v2i1.25.
- [9] E. Revita, "Analisis dan Perancangan Sistem Pengarsipan Perpustakaan Berbasis GUI," *J. SIMTIKA*, vol. 3, no. 2, pp. 1–6, 2020.
- [10] Serly Agustin, Arifin Yusuf Permana, Hari Noer

- Fazri, M. Rahssyal Daffa H, Mohammad Robi, and Ricky Firmansyah, "Implementasi Struktur Data Tree Pada Web Blog Sebagai Media Pembelajaran," *J. Inform. Dan Teknologi Komput.*, vol. 2, no. 2, pp. 206–212, 2022, doi: 10.55606/jitek.v2i2.316.
- [11] S. S. Winarsih and A. W. Wahono, "Implementasi Dan Pengujian Struktur Data Berbasis Acuan Untuk Program Aplikasi Mengungkap Kepribadian Berdasarkan Tanggal Lahir Dan Nama," *J. Teknol. Inf. dan Komun.*, vol. 10, no. 2, p. 11, 2022, doi: 10.30646/tikomsin.v10i2.631.
- [12] S. Tresnawati, "PELATIHAN MENGENALKAN STRUKTUR DATA PADA SMK NEGERI 2 CIMAHI Training on Introducing Data Structure at SMK Negeri 2 Cimahi," vol. 6, no. 1, pp. 341–350, 2024.
- [13] S. Susilowati, "Implementation of Ferry Parking Management System Using Python-Based Stack Method," *Int. J. Inf. Syst. Technol. Akreditasi*, vol. 8, no. 158, pp. 47–53, 2024.
- [14] B. Kurniawan and M. Romzi, "Pembuatan dan Pelatihan Administrator Website pada Dinas Kesehatan Kabupaten Ogan Komering Ulu," *J. Pengabd. Masy.*, vol. 2, no. 3, pp. 253–258, 2022, doi: 10.31004/abdira.v2i3.202.
- [15] R. Cahya Utama and R. Titi Komalasari, "STRING (Satuan Tulisan Riset dan Inovasi Teknologi) APLIKASI CHATBOT BERBASIS TEKS MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER FAQ GRABADS."
- [16] Sindy Nova, Nurul Khotimah, and Maria Y Aryati Wahyuningrum, "Pemanfaatan Chatbot Menggunakan Natural Language Processing Untuk Pembelajaran Dasar-Dasar Gui Tkinter Pada Bahasa Pemrograman Python," *J. Ilm. Tek.*, vol. 3, no. 1, pp. 58–65, 2024, doi: 10.56127/juit.v3i1.1162.