

**LAPORAN AKHIR**  
**STUDI INDEPENDEN BERSERTIFIKAT**  
**Foundations of AI and Life Skills for Gen-Z**  
**di Orbit Future Academy**

Diajukan untuk memenuhi persyaratan kelulusan  
Program MSIB MBKM

oleh :

Aulia Sava Febiyana / 13040119130073



**S-1 Ilmu Perpustakaan**  
**Fakultas Ilmu Budaya**  
**Universitas Diponegoro**  
**Semarang**  
**2022**

**Lembar Pengesahan**  
**S-1 Ilmu Perpustakaan Universitas Diponegoro**  
**Sistem Rekomendasi Buku Perpustakaan**  
**(Literature Recommender)**  
**di Orbit Future Academy**

oleh :

Aulia Sava Febiyana / 13040119130073

disetujui dan disahkan sebagai  
Laporan Magang atau Studi Independen Bersertifikat Kampus Merdeka

Semarang, 16 Juni 2022  
Pembimbing Magang atau Studi Independen  
S-1 Ilmu Perpustakaan  
Universitas Diponegoro



Dra. Rukiyah, M.Hum.

NIP 196405281991032011

**Lembar Pengesahan**  
**Sistem Rekomendasi Buku Perpustakaan**  
**(Literature Recommender)**  
**Di Orbit Future Academy**

oleh :

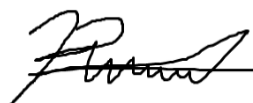
Aulia Sava Febiyana / 13040119130073

disetujui dan disahkan sebagai

Laporan Magang atau Studi Independen Bersertifikat Kampus Merdeka

Semarang, 15 Juni 2022

AI Coach



Muhammad Rifal Alfarizy

NIP: 2201058

## Abstraksi

Kegiatan Studi Independen Bersertifikat dalam Program Foundations of AI and Life Skills for Gen-Z bersama Orbit Future Academy, pada proses dan pencapaiannya, kami harus mengerjakan sebuah proyek yang bisa bermanfaat bagi masyarakat. Proyek akhir yang kami kerjakan adalah “*Sistem Rekomendasi Buku Perpustakaan*”. Proyek ini berhubungan dengan bidang pendidikan. Sistem yang kami buat dan kami kembangkan sangat berguna bagi dunia *literasi* karena rendahnya minat baca pada masyarakat Indonesia. Permasalahan yang sering dijumpai oleh para pembaca buku adalah menentukan buku-buku yang akan mereka baca selanjutnya. Kesulitan pembaca buku dalam menentukan buku yang akan dibaca disebabkan oleh banyaknya jumlah buku dan beragamnya jumlah buku yang ada. Solusi untuk permasalahan yang dialami pembaca adalah dengan menerapkan sistem rekomendasi buku yang dapat memberikan rekomendasi buku kepada pembaca buku.

Sehingga, kami membuat sistem rekomendasi buku perpustakaan. Pada penerapannya, *Sistem Rekomendasi Buku Perpustakaan* ini dapat merekomendasikan buku yang sesuai dengan kebutuhan pengguna berdasarkan kemiripan nama penulis buku di suatu perpustakaan. Kami menggunakan domain *Natural Language Processing* dalam mengaplikasikan *Sistem Rekomendasi Buku Perpustakaan* ini.

***Kata kunci : Perpustakaan, Artificial Intellegence, Orbit AI For Gen Z.***

## Kata Pengantar

Puji dan syukur kami panjatkan kepada Allah Subhanallahu Wa Ta'ala karena atas karunianya, kami dapat melaksanakan Program Magang dan Studi Independen Bersertifikat Kampus Merdeka dengan topik Foundations of AI and Life Skills for Gen-Z bersama Orbit Future Academy serta kami dapat menyelesaikan Laporan Akhir Studi Independen Bersertifikat ini sebagai pertanggungjawaban dari pelaksanaan program tersebut yang dilaksanakan pada 21 Februari – 22 Juli 2022 di Orbit Future Academy.

Selain itu, laporan ini sebagai salah satu bukti terlaksananya Program Magang Studi Independen Bersertifikat tersebut. Dalam menyelesaikan program ini kami mendapatkan banyak dukungan dari semua pihak yang bersangkutan. Maka dari itu, kami mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Orang tua kami yang selalu mendo'akan dan memberikan dukungan untuk kami.
2. Kemendikbud RI yang telah mengadakan Program Merdeka Belajar MSIB ini dan memberikan kesempatan mahasiswa belajar di luar perkuliahan selama 1 semester ini.
3. Orbit Future Academy yang telah memberikan peluang bagi kami untuk bisa mengembangkan dan meningkatkan skill melalui ilmu dan segala fasilitas yang diberikan.
4. Coach Muhammad Rifal Alfarizy sebagai *Homeroom* Coach yang selalu membina dan membimbing kami dalam Program Merdeka Belajar MSIB.
5. Coach Adi Syahadi sebagai *Technical AI* Coach yang selalu membimbing dan mendukung kami dalam Program Merdeka Belajar MSIB.
6. Semua Coach yang telah mengajar dan membantu kami untuk bisa berkembang dalam program Merdeka Belajar MSIB.
7. Universitas Diponegoro beserta para akademisi kampus yang telah mendukung mahasiswanya untuk bisa mengikuti Program Merdeka Belajar MSIB.
8. Semua teman-teman kelas Ibnu Sina yang selalu memberikan dukungan dan motivasi dalam Program Merdeka Belajar MSIB.

9. Semua peserta MSIB Kampus merdeka angkatan 2 bersama Orbit Future Academy yang selalu memberikan semangat untuk terus belajar dan berkembang.

Penulis menyadari terdapat banyak sekali kesalahan dan kekurangan dalam penyusunan laporan akhir ini dan jauh dari kata sempurna. Hal ini disebabkan oleh terbatasnya kemampuan dan ilmu yang dimiliki penulis. Oleh karena itu penulis sangat mengharapkan saran dan kritik yang membangun dari seluruh pihak untuk perbaikan hasil laporan akhir ini.

Semoga dengan adanya laporan akhir ini dapat memberikan manfaat untuk penulis dan pembaca sebagai acuan semangat untuk terus meraih mimpi dengan keterbatasan pengetahuan dari penulis, maka penulis meminta maaf yang seluas-luasnya. Atas perhatiannya, penulis ucapkan terima kasih kepada semua pihak.

Semarang, 16 Juni 2022

Penulis

## DAFTAR ISI

<b>Bab I</b>	<b>Pendahuluan</b>	<b>1</b>
I.1	Latar belakang	1
I.2	Lingkup	3
I.3	Tujuan	4
<b>Bab II</b>	<b>Orbit Future Academy</b>	<b>5</b>
II.1	Struktur Organisasi	5
II.2	Lingkup Pekerjaan	6
II.3	Deskripsi Pekerjaan	7
II.4	Jadwal Kerja	8
<b>Bab III</b>	<b>Sistem Rekomendasi Buku Perpustakaan</b>	<b>10</b>
III.1	Latar Belakang Proyek Akhir	10
III.2	Proses Pelaksanaan Proyek Akhir	11
III.3	Hasil Proyek Akhir	33
<b>Bab IV</b>	<b>Penutup</b>	<b>37</b>
IV.1	Kesimpulan	37
IV.2	Saran	38
<b>Bab V</b>	<b>Referensi</b>	<b>39</b>
<b>Bab VI Lampiran</b>	<b>A. TOR</b>	<b>41</b>
<b>Bab VII Lampiran</b>	<b>B. Log Activity</b>	<b>43</b>
<b>Bab VIII Lampiran</b>	<b>C. Dokumen Teknik</b>	<b>45</b>

**DAFTAR TABEL**

Tabel 2.1 Agenda Kelas	9
Tabel 7.1 Agenda Kelas	42
Tabel 8.1 Aktivitas selama pengerjaan Proyek Akhir	43



## DAFTAR GAMBAR

Gambar 2.1 Logo Orbit Future Academy	5
Gambar 2.2 Struktur Organisasi OFA	6
Gambar 3.1 Tampilan Goodreads dengan target dataset	16
Gambar 3.2 Tampilan Website Goodreads (Book-Shelves)	16
Gambar 3.3 Tampilan Import/Export	17
Gambar 3.4 Tampilan Dataset yang digunakan (.CSV)	17
Gambar 3.5 Tampilan jumlah total dataset	18
Gambar 3.6 Tampilan dataset buku	18
Gambar 3.7 Tampilan dataset detail buku	19
Gambar 3.8 Tampilan dataset rating dengan detail	19
Gambar 3.9 Tampilan Error cek pada dataset buku	19
Gambar 3.10 Tampilan Error cek pada dataset rating	20
Gambar 3.11 Preprocessing Data	20
Gambar 3.12 Rumus TF-IDF.[4]	22
Gambar 3.13 Rumus <i>Cosine Similarity</i>	22
Gambar 3.14 Tampilan format dataset dari website kaggle	23
Gambar 3.15 Block-Of-Code TF-IDFVectorizer	23
Gambar 3.16 <i>Mapping</i> antara judul dengan index	24
Gambar 3.17 Array untuk judul buku dengan iterasi guna untuk mencari kecocokan dengan <i>Input</i> data yang dimasukkan oleh <i>user</i>	24
Gambar 3.18 Model Content-Based dengan TF-IDF	25
Gambar 3.19 Inisialisasi index dari variable meta dengan parameter judul yang terdapat dalam parameter fungsi <code>get_title()</code>	25
Gambar 3.20 Enumerasi dengan <code>cosine_sim</code> parameter index menjadi <i>list</i>	25

Gambar 3.21 <i>Sorted</i>	26
Gambar 3.22 <i>Similarity Score</i>	26
Gambar 3.23 Iterasi untuk skor_sim	26
Gambar 3.24 Membuat <i>Dataframe</i> kosong untuk memasukan hasil fungsi <i>Get_title()</i> .	26
Gambar 3.25 Hasil <i>Similarity Score</i> dengan bookId sebagai Index	27
Gambar 3.26 Skor Similarity 3 dari 5 buku yang direkomendasikan dimana terdapat nama judul dari hasil pencarian.	28
Gambar 3.27 Tampilan Repository GitHub	30
Gambar 3.28 Tampilan <i>Text Editor</i> Visual Studio Code	30
Gambar 3.29 <i>Framework</i> Flask yang sudah di integrasikan dengan logika python	31
Gambar 3.30 Memasukan penulisan kode python kedalam content.py sebagai <i>API</i> untuk app.py	32
Gambar 3.31 Tampilan <i>home</i> aplikasi literature recommender	34
Gambar 3.32 Fitur <i>recommender system</i>	34
Gambar 3.33 Tampilan <i>our booklist</i>	35
Gambar 3.34 Tampilan <i>about us</i>	35
Gambar 8.1 Tampilan Goodreads dengan target dataset	49
Gambar 8.2 Tampilan Website Goodreads (Book-Shelves)	49
Gambar 8.3 Tampilan Import/Export	50
Gambar 8.4 Tampilan Dataset yang digunakan (.CSV)	50
Gambar 8.5 Tampilan jumlah total dataset	51
Gambar 8.6 Tampilan dataset buku	51
Gambar 8.7 Tampilan dataset detail buku	52
Gambar 8.8 Tampilan dataset rating dengan detail	52

Gambar 8.9 Tampilan Error cek pada dataset buku	52
Gambar 8.10 Tampilan Error cek pada dataset rating	53
Gambar 8.11 Preprocessing Data	53
Gambar 8.12 Skor Similarity 3 dari 5 buku yang direkomendasikan dimana terdapat nama judul dari hasil pencarian.	55
Gambar 8.13 Tampilan Repository GitHub	57
Gambar 8.14 Tampilan <i>Text Editor</i> Visual Studio Code	57
Gambar 8.15 <i>Framework</i> Flask yang sudah di integrasikan dengan logika python	58
Gambar 8.16 Memasukan penulisan kode python kedalam content.py sebagai <i>API</i> untuk app.py	59
Gambar 8.17 <i>User interface</i> web Literature Recommender	61

**I.1 Latar belakang**

Era digital 4.0 ditandai dengan pesatnya perkembangan teknologi. Pada era ini teknologi dapat digunakan dalam semua aspek. Teknologi tersebut berasal dari implementasi program *Artificial Intelligence*. *Artificial Intelligence* (AI) merupakan cabang ilmu komputer yang membahas tentang penangkapan, pemodelan, dan penyimpanan kecerdasan manusia ke dalam sebuah teknologi informasi yang nantinya dapat dimanfaatkan untuk pengambilan keputusan [1]. AI juga memberikan kemudahan kepada manusia dengan meminimalkan kesalahan, membantu mengotomasi pekerjaan, dan mempercepat pekerjaan tersebut. Oleh karena itu, AI dapat membantu manusia untuk mengefisienkan pekerjaan.

Urgensi penggunaan AI di masa depan dapat mendorong generasi penerus untuk mempelajari bagaimana memahami dan mengembangkan bidang teknologi khususnya AI. Pengembangan potensi mahasiswa sebagai generasi penerus dalam mempelajari AI sebagai kompetensi di masa depan didukung penuh oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi. Hal tersebut diwujudkan dalam adanya penyelenggaraan program Kampus Merdeka Studi Independen Bersertifikat yang memiliki visi dan misi dalam mendorong mahasiswa untuk mempelajari kompetensi yang spesifik, praktis, dan dibutuhkan di masa depan [2].

Program ini memfasilitasi mahasiswa untuk belajar secara luas terkait bidang yang ingin digeluti. Program ini tidak mewajibkan mahasiswa harus memilih pembelajaran sesuai dengan jurusan yang diambil di kampus. Sehingga kebebasan untuk belajar sangat dijamin dalam program tersebut. Kebebasan belajar tersebut memberikan solusi pada permasalahan kualitas sumber daya manusia di bidang teknologi. Salah satu upaya yang dapat dilakukan untuk meningkatkan kualitas sumber daya manusia di bidang teknologi dengan mengikuti program kampus merdeka studi independen. Mitra yang dipilih untuk program studi independen ini adalah PT. Orbit Ventura Indonesia. PT. Orbit Ventura Indonesia merupakan perusahaan penyedia jasa pembelajaran terkait teknologi.

Program Magang dan Studi Independen Bersertifikat Kampus Merdeka di Orbit Future Academy dengan topik Foundations of AI and Life Skills for Gen-Z dilaksanakan karena Indonesia sangat membutuhkan peningkatan intensitas dan kualitas transfer pengetahuan yang berfokus pada teknologi, AI, dan keterampilan masa depan, demi menjalankan era Industri 4.0 secara produktif dan konstruktif. Pada saat ini dan masa depan, wawasan teknologi dan keterampilan rekayasa AI merupakan kompetensi masa depan yang mutlak harus dikuasai sebanyak mungkin oleh generasi bangsa Indonesia.

Data terbaru Kamar Dagang dan Industri (Kadin) Indonesia menguatkan hal tersebut dengan mencatat terdapat kebutuhan tenaga kerja domestik sebanyak lebih dari 40 juta orang hingga 2050 untuk menempati berbagai posisi dan fungsi pekerjaan masa depan yang sebelumnya tidak ada di masa lalu, yang berlaku lintas-industri. Mayoritas dari posisi dan fungsi pekerjaan tersebut sangat erat berkaitan dengan penguasaan teknologi dan rekayasa AI [3].

Program MSIB Kampus Merdeka bersama Orbit Future Academy ini tidak dipungut biaya apapun untuk mahasiswa semester 5, 6, 7, serta tersertifikasi resmi karena Orbit Future Academy telah berizin dari Kemdikbudristek. Program ini berbobot 20 SKS, di mana konversi SKS dan kualitas program dijamin oleh Kemdikbudristek serta kegiatan dapat diikuti secara daring.

Foundations of AI and Life Skills for Gen-Z merupakan program pembelajaran yang relevan bagi mahasiswa semua jurusan atau program studi, dengan konten materi pembelajaran yang lebih sederhana, beragam, serta dilengkapi dengan sejumlah modul keterampilan bisnis dan kemandirian hidup di era Industri 4.0, misalnya *entrepreneurship skill*. Ada pula modul *corporate skill*, *business skill*, *leadership and management skill*, *technology skill*, dan materi dasar tentang rekayasa AI, sehingga Foundations of AI and Life Skills for Gen-Z cocok dipelajari oleh para pencari pekerjaan atau para pencipta lapangan kerja.

Dalam pengerjaan Proyek Akhir ini, kami menggunakan Domain *Natural Language Preprocessing* serta membuat dan mengembangkan “*Sistem Rekomendasi Buku Perpustakaan*”. Kami mengembangkan ide tersebut dalam pembuatan proyek akhir karena ide tersebut berhubungan dengan materi yang kami pelajari selama

pelaksanaan program serta berkaitan dengan tujuan, visi, misi, maupun fungsi dari Orbit Future Academy.

*Sistem Rekomendasi Buku Perpustakaan*, pada penerapannya dapat merekomendasikan buku yang sesuai dengan kebutuhan pengguna berdasarkan kemiripan nama penulis buku yang dicari pengguna di suatu perpustakaan. Proyek ini merupakan bentuk nyata teknologi dan AI yang akan membantu masyarakat melalui efektivitas, efisiensi, akurasi, dan konsistensi yang ditawarkan pada *Sistem Rekomendasi Buku Perpustakaan* ini. Proyek ini juga diharapkan dapat membawa transformasi signifikan bagi dunia pendidikan Indonesia pada saat ini dan masa depan.

## **I.2 Lingkup**

Pelaksanaan studi independen kampus merdeka batch 2 ini dilakukan di PT. Orbit Ventura Indonesia (Orbit Future Academy) yang merupakan salah satu perusahaan yang bergerak dalam bidang Pendidikan teknologi di Indonesia. Dengan mengikuti salah satu program Studi Independen Kampus Merdeka di Orbit Future Academy, yaitu Foundations of AI and Life Skills for Gen-z, diharapkan penulis bisa mendapatkan pengetahuan dan bisa menguasai terkait program yang diambil, yaitu *Artificial Intelligence* (Kecerdasan Buatan). Sebelum melaksanakan proyek akhir, mahasiswa telah dibimbing oleh mentor yang telah disediakan oleh mitra. Mentor juga membantu mahasiswa untuk menentukan arah dari proyek akhir dan memberikan masukan serta arahan terkait dengan proyek akhir. Lingkup pekerjaan dari anggota kelompok dalam mengerjakan proyek akhir ini memiliki peran dan tugas pokok masing-masing, antara lain sebagai berikut:

- Data akuisisi : Tanty Lianasondari
- Modelling AI : Rifky Abdul Hakim
- Deployment : Aulia Sava Febiyana & Shofa Mumtazah
- Laporan : Putri Anggi Nauli

### **I.3 Tujuan**

Tujuan atau hasil dari mengikuti Program MSIB Kampus Merdeka yang telah kami peroleh adalah kami mendapatkan kesempatan untuk belajar dan mengembangkan diri melalui aktivitas di luar kelas perkuliahan namun tetap diakui sebagai bagian dari perkuliahan, mampu menguasai kompetensi spesifik dan praktis yang juga dicari oleh dunia usaha dan dunia industri kerja, memiliki wawasan mengenai AI beserta perkembangannya, mampu merancang dan mengimplementasikan AI Project Cycle, mampu menggunakan bahasa pemrograman Python untuk mengembangkan aplikasi AI, mampu mengembangkan salah satu dari 3 domain AI (DS, NLP, dan CV) hingga tahap *deployment*, mampu menggunakan *soft skills* dan *hard skills* dalam dunia industri dan lingkungan perusahaan, mampu mengaplikasikan kiat-kiat yang dibutuhkan seorang wirausahawan yang bergerak di bidang *start-up* dalam mentransformasikan ide ke dalam bentuk produk/jasa sehingga dapat menciptakan peluang bisnis yang terus berinovasi, berevolusi, dan berkelanjutan.

**II.1 Struktur Organisasi****Gambar 2.1 Logo Orbit Future Academy**

Orbit Future Academy (OFA) didirikan pada tahun 2016 dengan tujuan untuk meningkatkan kualitas hidup melalui inovasi, edukasi, dan pelatihan keterampilan. Label atau *brand* Orbit merupakan kelanjutan dari warisan mendiang Prof. Dr. Ing. B. J. Habibie (presiden Republik Indonesia ke-3) dan istrinya, Dr. Hasri Ainun Habibie. Mereka berdua telah menjadi penggerak dalam mendukung perkembangan inovasi dan teknologi pendidikan di Indonesia. OFA mengkurasi dan melokalkan program/kursus internasional untuk *upskilling* atau *reskilling* pemuda dan tenaga kerja menuju pekerjaan masa depan. Hal ini sesuai dengan slogan OFA, yakni “*Skills-for-Future-Jobs*”.

**Visi:**

Memberikan pembelajaran berbasis keterampilan transformatif terbaik untuk para pencari kerja & pencipta lapangan kerja.

**Misi:**

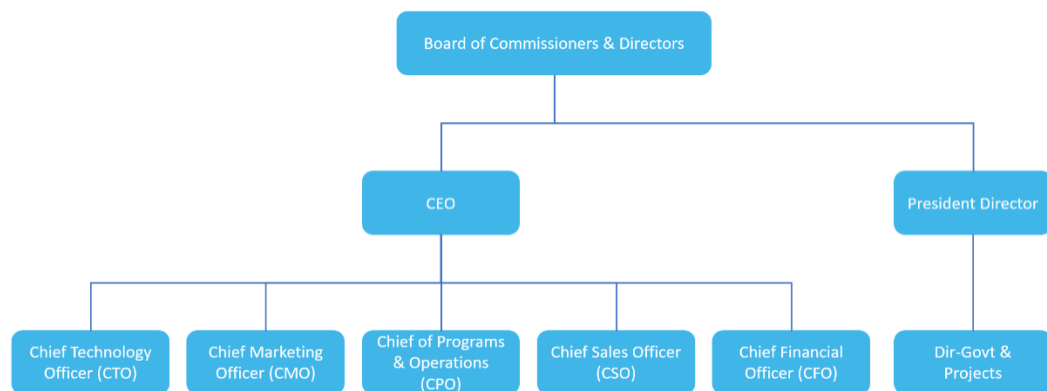
1. Membangun jaringan Orbit Transformation Center (OTC) secara nasional untuk menyampaikan kurikulum keterampilan masa depan berbasis sertifikasi melalui Platform Konten Digital.



2. Secara proaktif bekerja dengan pemerintah & organisasi dengan mengubah tenaga kerja mereka agar sesuai dengan perubahan pekerjaan yang terjadi karena Industri 4.0.
3. Melatih pemuda dengan keterampilan kewirausahaan & mencocokkan mereka dengan peluang masa depan yang muncul di berbagai industri.
4. Menghubungkan jaringan inkubator dan akselerator yang dikurasi ke industri, investor, dan ekosistem start-up global.

Struktur organisasi OFA dapat dilihat pada Gambar 2.2.

**Gambar 2.2 Struktur Organisasi OFA**



## II.2 Lingkup Pekerjaan

Seorang fasilitator akan mendampingi kurang lebih 40 peserta MSIB (student) dalam satu kelas. Terdapat dua jenis fasilitator, yakni:

### a. Homeroom Coach

Homeroom coach bertugas menyampaikan materi tentang dasar-dasar AI, memberikan penilaian pada student, dan mengarahkan *student* saat pengerjaan Proyek Akhir (PA).

### **b. Domain Coach**

Domain coach bertugas menyampaikan materi tentang domain AI atau life skills dan memberikan penilaian pada student.

## **II.3 Deskripsi Pekerjaan**

Berikut adalah deskripsi pekerjaan student sebelum pengerjaan PA:

- a. Mengikuti pre-test.
- b. Mengikuti kelas sesi pagi pada pukul 08.00 hingga 11.30 WIB.
- c. Mengikuti kelas sesi siang pada pukul 13.00 hingga 16.30 WIB.
- d. Mengulang materi yang telah disampaikan di kelas sesi pagi dan siang, setelah kelas sesi siang, selama 1 jam (*self-study*).
- e. Mengerjakan latihan individu atau kelompok yang diberikan oleh homeroom atau domain coach saat kelas berlangsung.
- f. Mengerjakan tugas yang diberikan homeroom atau domain coach hingga batas waktu tertentu.
- g. Mengerjakan *mini project* yang diberikan homeroom atau domain coach hingga batas waktu tertentu
- h. Mengikuti post-test.

Student memiliki tugas dalam tahap *deployment* selama pengerjaan PA, dengan deskripsi pekerjaan sebagai berikut:

### **a. Mengidentifikasi *problem scoping***

Pada tahap ini, saya mengidentifikasi *problem scoping* dari aplikasi yang akan dirancang dengan menguraikannya dalam model berikut:

WHO - Siapa yang memiliki masalah tersebut?

WHAT - Apa permasalahan sebenarnya? Apa dari peran pihak terkait? Apa solusi yang ditawarkan untuk memecahkan masalah tersebut?

WHERE - Dimana/pada saat apa permasalahan ini muncul?

WHY - Mengapa kamu yakin masalah ini sangat penting dibahas?

**b. *Data cleansing* dari *data acquisition***

Tahap ini, saya melakukan *data cleansing* dari data yang sudah didapatkan pada tahap *data acquisition*. Saya melakukan pembersihan pada data yang dibutuhkan, menginputkan data yang masih kurang lengkap, dan menyeleksi kembali data yang telah diperoleh.

**c. Membuat *data dummy***

Dalam tahap ini, saya membuat *data dummy* dari data yang diperoleh pada proses *data acquisition*. Saya membuat *data dummy* berupa *rating* dari buku-buku.

**d. Melakukan perencanaan desain *interface* aplikasi**

Pada proses ini, saya berdiskusi dan memberikan rekomendasi tema aplikasi, *tools* yang ingin ditampilkan, menu yang disediakan dengan teman sekelompok. Selain itu, saya juga mencari *interface* yang tepat untuk aplikasi yang kami kembangkan.

**e. Melakukan *web programming* aplikasi**

Jika rancangan *interface* dari aplikasi sudah disepakati oleh teman sekelompok, saya melakukan *web programming* untuk membuat halaman *web* untuk aplikasi AI yang sudah kami buat pada tahap *modelling*. Tidak ada kendala yang berarti pada tahap ini, sehingga pengerjaannya pun relatif lancar dan memakan waktu sekitar  $\pm 7$  hari saja. *Tools* yang saya gunakan dalam tahap ini adalah Visual Studio Code sebagai *text editor* HTML & CSS dan GitHub untuk repositori kode yang sudah dibuat di Visual Studio Code.

## **II.4 Jadwal Kerja**

Program ini berlangsung setiap hari kerja (Senin sampai dengan Jumat) selama 8 jam per harinya, dengan rincian sebagai berikut:

**Tabel 2.1 Agenda Kelas**

<b>Pukul (WIB)</b>	<b>Durasi (jam)</b>	<b>Aktivitas</b>
08.00 s.d. 11.30	3.5	Kelas Sesi Pagi
13.00 s.d. 16.30	3.5	Kelas Sesi Siang
16.30 s.d. 17.30	1	<i>Self-Study</i>

Program ini berlangsung dari bulan Februari 2022 sampai dengan bulan Juli 2022.

**III.1 Latar Belakang Proyek Akhir**

Banyaknya jumlah buku dan beragamnya jenis buku yang ada di perpustakaan, membuat pembaca terkadang kesulitan dalam menentukan buku yang hendak mereka baca selanjutnya. Terkadang dijumpai pembaca yang hanya ingin membaca buku-buku yang memiliki reputasi penjualan terbaik. Ada pula pembaca yang hanya ingin membaca buku dengan *rating* yang tinggi dan memiliki *Review*. Tidak jarang pula ditemui pembaca yang menentukan buku-buku yang akan dibaca selanjutnya dari kemiripan judul ataupun nama penulis buku tersebut. Sehingga bisa mendapatkan referensi baru melainkan hanya dari *rating* pengguna saja, kemiripan nama penulis buku dapat dijadikan landasan untuk memberikan rekomendasi buku kepada pembaca.

Mencari buku secara manual di perpustakaan akan segera berlalu karena akan muncul serangkaian aplikasi teknologi informasi yang bisa membantu kinerja para pustakawan menjadi lebih mudah, yaitu dengan adanya kecerdasan buatan atau *Artificial Intelligence* (AI) [4]. *Artificial Intelligence* merupakan suatu kecerdasan buatan yang terintegrasi dengan sistem.

Kecerdasan buatan (*Artificial Intelligence*) adalah pengetahuan yang memungkinkan komputer untuk meniru kecerdasan manusia dan komputer dapat melakukan apa yang perlu dilakukan manusia [5]. AI dapat dimanfaatkan menjadi sebuah strategi pengelola perpustakaan untuk mempromosikan layanan dan fasilitas yang nantinya akan hadir dan kemudian diberikan kepada pengguna perpustakaan untuk mempermudah proses pencarian buku [6].

Sistem rekomendasi dapat digunakan untuk memprediksi barang tertentu yang disukai oleh pengguna atau untuk mengidentifikasi beberapa barang yang mungkin disukai oleh pengguna tertentu. Sistem rekomendasi memberikan rekomendasi barang-barang kepada penggunanya berdasarkan preferensi eksplisit dan implisit, preferensi dari pengguna lain, dan atribut dari barang yang direkomendasikan.

Sistem rekomendasi memberikan solusi terhadap permasalahan dalam menentukan buku yang belum pernah dibaca oleh pengguna. Dengan hadirnya sistem rekomendasi ini, bisa membantu calon pembaca yang bingung karena banyaknya jenis dan pilihan buku yang ada. Sistem ini akan memberikan calon pembaca rekomendasi buku yang diharapkan dapat sesuai dengan kemiripan nama penulis buku. Sehingga, pembaca tidak akan mengalami kesulitan dalam melakukan pencarian literasi di rak buku satu persatu.

### **III.2 Proses Pelaksanaan Proyek Akhir**

Pada proses pelaksanaan proyek akhir MSIB ini, kami membuat *Sistem Rekomendasi Buku Perpustakaan*. Kami menyusun AI Project Cycle terlebih dahulu, yaitu melakukan analisis masalah dan solusi yang ditawarkan, sumber data, jumlah data, proses pembersihan data, cara pengambilan data, proses modelling dan algoritma yang digunakan, dan deployment (web, smartphone, atau local deployment). Kami menemukan suatu masalah dan mencoba untuk menganalisis masalah tersebut kemudian menawarkan solusi yang dapat kami berikan melalui proyek akhir yang kami kerjakan.

Kami menemukan suatu permasalahan yang berkaitan dengan bidang pendidikan, yaitu rendahnya literasi masyarakat Indonesia. Banyaknya jumlah buku dan beragamnya jenis buku yang ada di perpustakaan, membuat pembaca terkadang kesulitan dalam menentukan buku yang hendak mereka baca selanjutnya. Sehingga kami menyusun *Problem Scoping* terlebih dahulu terkait topik yang akan kami bahas dalam pengerjaan proyek akhir ini. Solusi yang kami tawarkan adalah *Sistem Rekomendasi Buku* yang akan memberikan solusi terhadap permasalahan dalam menentukan buku yang belum pernah dibaca oleh pengguna. Dengan hadirnya sistem rekomendasi ini, bisa membantu calon pembaca yang bingung karena banyaknya jenis dan pilihan buku yang ada.

Sistem ini akan memberikan calon pembaca rekomendasi buku yang sesuai berdasarkan judul buku yang pengguna pernah baca atau pengguna suka, dimana model memberikan rekomendasi buku dari nama penulis yang sama atau nama penulis yang memiliki kemiripan. Domain yang kami gunakan dalam *Sistem*

*Rekomendasi Buku Perpustakaan* ini adalah Natural Language Processing (NLP) karena kami menggunakan Content Based dalam melakukan pemodelan. Berikut merupakan tahapan dalam proses pengerjaan Proyek Akhir kami.

#### Tahap Problem Scoping

##### **Topik yang diambil :**

Penerapan *Sistem Rekomendasi Buku Perpustakaan* ini adalah dengan merekomendasikan buku dari judul buku yang pengguna pernah baca atau pengguna suka, dimana model memberikan rekomendasi buku dari nama penulis yang sama atau nama penulis yang memiliki kemiripan. Domain yang kami gunakan dalam *Sistem Rekomendasi Buku Perpustakaan* ini adalah *Natural Language Processing (NLP)* karena kami menggunakan Content Based dalam melakukan pemodelan.

##### **• WHO**

###### **Siapa yang memiliki masalah tersebut?**

- Siapa yang mengalami permasalahan? =  
Pengunjung perpustakaan (pemustaka).
- Siapa sasaran pengguna aplikasi ini?  
Pengunjung perpustakaan (pemustaka) yang mengalami kesulitan dalam menemukan buku yang sesuai dan relevan dengan kebutuhan informasinya.
- Siapa saja pihak yang terlibat dalam proses pembuatan aplikasi ini?  
Perpustakaan, pengunjung perpustakaan (pemustaka), dan pustakawan.

##### **• WHAT**

###### **Apa permasalahan sebenarnya?**

- Apa permasalahan utamanya?

1. Sulitnya dalam mencari buku yang sesuai dengan kebutuhan dan ketertarikan pemustaka.
  2. Tidak adanya petunjuk buku apa saja yang memiliki kualitas yang baik untuk dijadikan pertimbangan pengunjung perpustakaan (pemustaka) dalam menentukan buku yang akan dibaca.
- Apa saja faktor pendukung permasalahan tersebut?
    1. Pemustaka kurang terampil untuk menentukan buku yang relevan dengan kebutuhan informasinya.
    2. Pemustaka kesulitan dengan proses temu balik pada buku perpustakaan untuk menemukan buku yang sesuai dengan kebutuhannya.
    3. Tidak adanya fasilitas untuk merekomendasikan buku yang sesuai dengan keinginan atau minat pembaca.

**Apa dari peran pihak terkait?**

- Apa peran dari pihak yang terlibat dalam proses pembuatan sistem ini?
  1. Perpustakaan berperan sebagai penyedia jasa layanan informasi harus mampu mengakomodir kebutuhan informasi pemustaka.
  2. Pustakawan sebagai seseorang yang membantu pemustaka dalam menemukan buku yang tepat dan sesuai dengan kebutuhan informasinya.

**Apa solusi yang ditawarkan untuk memecahkan masalah tersebut?**

- Apa tujuan dan manfaat dari pembuatan aplikasi ini?
  1. Membantu pemustaka dalam menentukan buku yang tepat sesuai dengan keinginan dan kebutuhan informasi pembaca.
  2. Memudahkan proses temu balik informasi di perpustakaan.



3. Pemustaka mendapatkan informasi baru dan saran terkait buku apa saja yang memiliki kualitas yang baik dan layak untuk dibaca.

- **WHERE**

**Dimana/pada saat apa permasalahan ini muncul?**

- Di mana saja calon '*user*' merasakan masalah ini?  
Calon '*user*' merasakan masalah ini ketika mereka berada di perpustakaan.
- Pada kondisi seperti apa calon '*user*' mengalami masalah ini?  
Calon '*user*' mengalami masalah tersebut ketika mereka sedang melakukan temu balik informasi pada koleksi buku milik perpustakaan dan merasa kebingungan untuk menentukan buku mana yang sesuai dengan kebutuhan informasinya.

- **WHY**

**Mengapa kamu yakin masalah ini sangat penting dibahas?**

- Mengapa permasalahan ini sangat penting?  
Salah satu indikator keberhasilan perpustakaan dari dilihat dari pemanfaatan koleksinya yang sudah sesuai dengan kebutuhan informasi pengguna atau pemustakanya [7].  
Oleh karena itu, perpustakaan harus memiliki fasilitas penunjang untuk membantu pemustaka agar dapat mempertemukan buku yang relevan dengan kebutuhan informasinya.
- Apa nilai utama dari permasalahan ini?  
Nilai utama dari permasalahan ini terdapat dalam pembukaan UUD, yaitu memiliki tujuan sesuai dengan poin mencerdaskan kehidupan bangsa, yang mana merupakan kedudukan perpustakaan dalam menyelenggarakan kegiatannya sebagai penyedia layanan informasi tanpa memandang SARA dan status sosial pemustakanya.

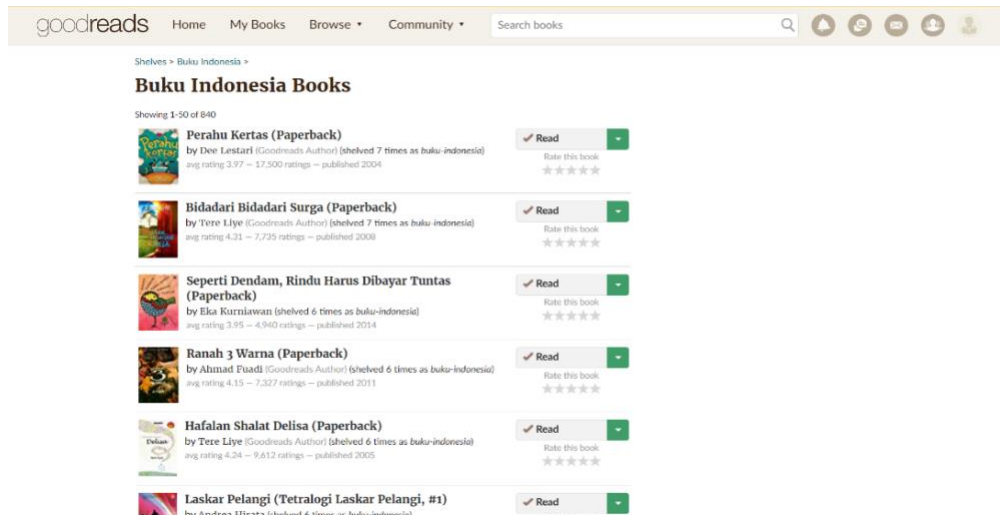
- Bagaimana aplikasi ini mampu mengatasi permasalahan tersebut?
  1. Mengembangkan teknologi, misalnya dengan membuat *recommender system* untuk koleksi buku perpustakaan dan mengintegrasikannya dengan OPAC (alat telusur koleksi perpustakaan) agar memudahkan pemustaka dalam mencari serta menentukan buku yang sesuai dan relevan dengan kebutuhan informasinya.
  2. Melakukan edukasi mengenai literasi informasi kepada pemustaka dari tahap identifikasi kebutuhan informasi pemustaka hingga tahap pemanfaatan buku yang sesuai tanpa melanggar kaidah hak cipta penerbitan buku tersebut.

### **Tahap Data Acquisition**

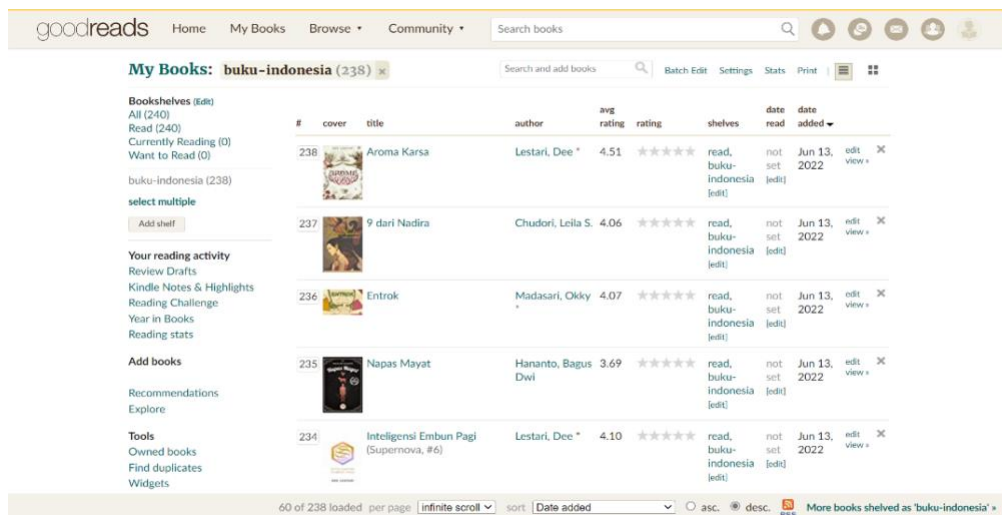
Kami mengambil data buku untuk proyek akhir ini dari website Goodreads dengan cara pengambilan manual dimana kami membuat akun pengguna terlebih dahulu untuk memakai fitur yang disediakan oleh goodreads dengan cara menambahkan data buku kedalam *bookshelves* dan mengekspor nya menjadi format CSV (*Comma Seperated Values*) guna keperluan untuk membuat dataset serta dengan menggunakan web-scrapping yaitu memakai ekstensi *Web Scraper* yang ada di internet browser Google chrome dan Microsoft Edge.

Kami mengambil sekitar 200 data buku Indonesia terutama buku sastra seperti penulis Pramoedya Ananta Toer hingga penulis buku novel karangan Tere liye. Format data yang kami ambil dimulai dari Judul, Penulis, ISBN hingga Rating per-buku.

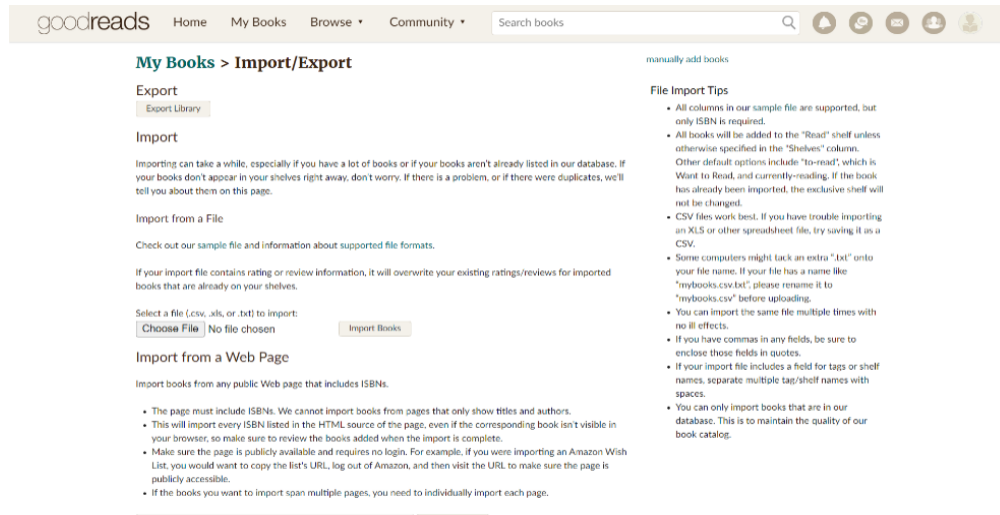
**Gambar 3.1 Tampilan Goodreads dengan target dataset**



**Gambar 3.2 Tampilan Website Goodreads (Book-Shelves)**



**Gambar 3.3 Tampilan Import/Export**



**Gambar 3.4 Tampilan Dataset yang digunakan (.CSV)**



## Tahap Data Exploration

Pada tahap ini, dataset yang berjumlah 200 data buku akan diproses agar bisa digunakan pada *Sistem Rekomendasi Buku Perpustakaan* yang akan kami buat. Berikut merupakan visualisasi dari dataset yang telah kami proses.

**Gambar 3.5 Tampilan jumlah total dataset**

```
Jumlah ISBN: 890
Jumlah Book-rating 1001

Jumlah judul buku: 987

judul buku:

['Karmila' 'Orang Miskin Dilarang Sekolah!' 'Jendela-Jendela'
 'Negeri Kabut' 'Angin Pun Berbisik' 'Tesaurus Bahasa Indonesia'
 'O Amuk Kapak' 'Tak Putus Dirundung Malang' 'Momoye: Mereka Memanggilku'
 'Dunia Sukab: Sejumlah Cerita'
 'Pergolakan Pemikiran Islam: Catatan Harian Ahmad Wahib' 'Perantau'
 'Cado-cado: Catatan Dodol Calon Dokter'
 'Tuhan & Hal-Hal yang Tak Selesai' 'Jazz, Parfum, dan Insiden'
 'Arok Dedes' 'Moslem Millionaire'
 'Perang Eropa Jilid I (Perang Eropa, #1)' 'Mencari Telur Garuda'
 'Azab dan Sengsara' 'Bibir dalam Pispot' 'Katastrofa Cinta'
 'Kitab Omong Kosong'
 'Masyarakat Sipil Untuk Transformasi Sosial: Pergolakan Ideologi LSM Indonesia'
 'Miss Jutek' 'The Da Peci Code' 'Politik Hati Nurani' 'Kun Fayakun'
 'Salah Pilih' 'Dian Yang Tak Kunjung Padam' 'Di Bawah Kibaran Sarung'
 'Menalar Tuhan' 'Ketika Jurnalisme Dibungkam Sastra Harus Bicara'
 'Jomblo: Sebuah Komedi Cinta'
 'Bulan Mati di Javasche Oranje (Johana Rijkaard, Buku 1)' 'Cala Ibi'
```

**Gambar 3.6 Tampilan dataset buku**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1001 entries, 0 to 1000
Data columns (total 8 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   ISBN                                  1001 non-null   object
1   Book-Title                           1001 non-null   object
2   Book-Author                          1001 non-null   object
3   Year-Of-Publication                  997 non-null    float64
4   Publisher                            996 non-null    object
5   Image-URL-S                          1001 non-null   object
6   Image-URL-M                          1001 non-null   object
7   Image-URL-L                          1001 non-null   object
dtypes: float64(1), object(7)
memory usage: 62.7+ KB
```

**Gambar 3.7 Tampilan dataset detail buku**

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
0	9789796623051	Karmila	Marga T.	1990.0	Gramedia Pustaka Utama
1	9793457171	Orang Miskin Dilarang Sekolah!	Eko Prasetyo	2004.0	INSISTPress
2	9796953870	Jendela-Jendela	Fira Basuki	2001.0	PT Gramedia Widiasarana Indonesia (Grasindo)
3	9795539930	Negeri Kabut	Seno Gumira Ajidarma	1999.0	Grasindo
4	9789791685238	Angin Pun Berbisik	Irwan Dwi Kusnanto	2008.0	Mitra Netra & VHRbook

**Gambar 3.8 Tampilan dataset rating dengan detail**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1001 entries, 0 to 1000
Data columns (total 3 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   User-ID         1001 non-null   int64
1   ISBN            1001 non-null   object
2   Book-Rating     1001 non-null   int64
dtypes: int64(2), object(1)
memory usage: 23.6+ KB
```

	User-ID	ISBN	Book-Rating
0	276001	9789796623051	7
1	276001	9793457171	7
2	276001	9796953870	7
3	276001	9795539930	8
4	276001	9789791685238	8

**Gambar 3.9 Tampilan Error cek pada dataset buku**

```
1 # mengecek jika ada error pada dataset
2 book.isnull().sum()
```

...	ISBN	0
	Book-Title	0
	Book-Author	0
	Year-Of-Publication	4
	Publisher	5
	Image-URL-S	0
	Image-URL-M	0
	Image-URL-L	0
	dtype: int64	



**Gambar 3.10 Tampilan Error cek pada dataset rating**

```
1 # mengecek jika ada error pada dataset
2 rating.isnull().sum()

[7]

... User-ID      0
     ISBN        0
     Book-Rating  0
     dtype: int64
```

### Tahap Pre-Processing

Pada tahap Pre-Processing, Kami memproses ulang kembali data yang sudah di ambil dengan metode manual dan webscraping sehingga siap digunakan untuk untuk tahap *modelling*.

**Gambar 3.11 Preprocessing Data**

```
10
11 ##### Data Preprocessing
12 # membersihkan kolom semua_buku dengan definisikan kepada berishkan_buku
13 bersihkan_buku = semua_buku.dropna()
14 # mengecek keadaan kolom yang sudah dibersihkan dan mereset urutan
15 bersihkan_buku.reset_index(drop=True,inplace=True)
16 # mensortir nilai pada kolom ISBN pada variabel berishkan_buku
17 persiapandata = bersihkan_buku.sort_values('ISBN',ascending=False)
18 # membersihkan duplikat data pada kolom ISBN
19 dataduplikat = persiapandata.drop_duplicates('ISBN')
20 # mengkonversi row rating menjadi list rating_buku
21 rating_buku = dataduplikat['Book-Rating'].tolist()
22 # mengkonversi row ISBN menjadi list id_buku
23 id_buku = dataduplikat['ISBN'].tolist()
24 # mengkonversi row Book-Author menjadi list author_buku
25 author_buku = dataduplikat['Book-Author'].tolist()
26 # mengkonversi row Book-Title menjadi list judul_buku
27 judul_buku = dataduplikat['Book-Title'].tolist()
28 # membuat key dictionary untuk data id_buku, author_buku, judul_buku
29 buku_baru = pd.DataFrame({
30     'id' : id_buku , #ISBN
31     'judul' : judul_buku ,
32     'author' : author_buku,
33     'penilaian': rating_buku
34 })
```

Pada tahap ini kami mencoba untuk melakukan preprocessing data dengan melakukan pembersihan dataset buku dari index kosong, membersihkan data duplikasi hingga dataset dimuat kembali kepada Pandas Dataframe dengan metode key dictionary yang berfungsi untuk mempermudah penamaan kolom dataset.

### **Tahap Modelling**

Pada tahap permodelan, kami menggunakan sistem rekomen berbasis item/konten (*Content based*) yaitu rekomen yang merekomendasikan konten berdasarkan kemiripan dengan konten lain [5] dimana sistem merekomendasikan buku dari nama penulis dengan memasukkan judul buku untuk permudah pengguna dalam mencari rekomendasi.

Kami menggunakan library scikit-learn yaitu model vektorisasi TF-IDF (*TfidfVectorizer*) dan menggunakan metrik Cosine similarity sebagai parameter untuk menentukan kesamaan skor kedekatan vektor dari satu nama penulis ke nama penulis lainnya yang berada dalam index dataset yang kami gunakan.

TF-IDF (*Term Frequencies – Inverse Document Frequencies*) adalah Metode yang menggabungkan dua konsep untuk perhitungan bobot, yaitu frekuensi kemunculan sebuah kata (*Term*) didalam sebuah dokumen dan *inverse* frekuensi dokumen yang mengandung kata tersebut [6]. Frekuensi kemunculan kata di dalam dokumen yang diberikan, menunjukkan seberapa penting kata tersebut didalam dokumen. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi di dalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut rendah pada sekumpulan dokumen [6].

berikut rumus untuk TF-IDF:



**Gambar 3.12 Rumus TF-IDF.[4]**

$$TF = \frac{\text{Frequency of the word in the sentence}}{\text{Total number of words in the sentence}}$$

$$idf(w) = \log\left(\frac{N}{df_t}\right)$$

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

tf<sub>i,j</sub> = number of occurrences of i in j

N = total number of documents  
tf<sub>i,j</sub> = number of occurrences of i in j

Dimana pada penerapan proyek kami gunakan untuk memetakan matriks *Term* atau Kata yang terdapat pada dataset Books1.csv yang sudah di *prerprocessing* dengan dataset Rating1.csv. Serta kami menggunakan metriks cosine similarity untuk menentukan skor kesamaan antara input yang dimasukan dengan dataset yang sudah dipetakan kedalam matriks vektorisasi TF-IDF.

Berikut untuk Rumus Cosine Similarity [6]:

**Gambar 3.13 Rumus Cosine Similarity**

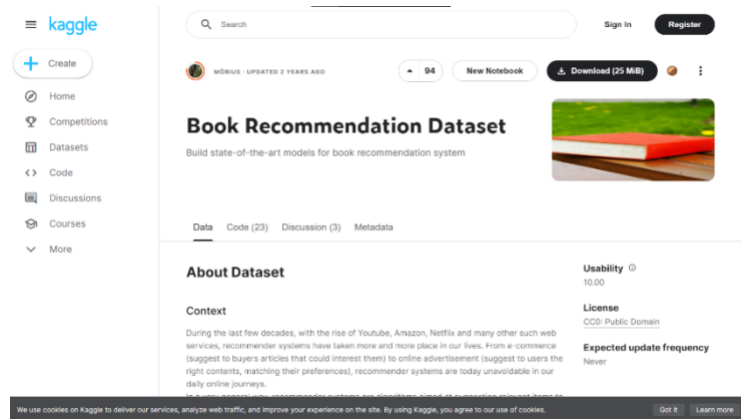
$$\text{Cos } \alpha = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}$$

Keterangan :

- A = Vektor A, yang akan dibandingkan kemiripannya
- B = Vektor B, yang akan dibandingkan kemiripannya
- A • B = *dot product* antara vektor A dan vektor B
- |A| = panjang vektor A
- |B| = panjang vektor B
- |A||B| = *cross product* antara |A| dan |B|

Pada awalnya kami melakukan *Preprocessing data*, dimana format data tersebut kami ambil refrensi dari proyek Kaggle [Book Recommendation Dataset | Kaggle](#)

**Gambar 3.14 Tampilan format dataset dari website kaggle**



Pertama kami mendefinisikan `Tfidfvectorizer` dan mengfittingkan nya dengan Kolom penulis menggunakan metode `.fit_transform` pada kolom dataset yang sudah dipreprocessing (Buku\_Baru) yaitu `Author`, lalu setelah dibuat vektorisasi `Tf-Idf`, kami mendefinisikan metriks cosine similarity, pada bagian line of code ini ada 2 fungsi yang bisa kita gunakan, yaitu `linear_kernel` dengan `cosine_similarity` pada penerapannya karena dataset yang kami ambil hanya berjumlah kurang lebih dari 200 entri data maka pada penerapan disini kita menggunakan cosine similarity dikarenakan jumlah data yang sedikit dan lebih efisien dibanding dengan `linear_kernel` yang diacukan untuk data yang berjumlah besar.

**Gambar 3.15 Block-Of-Code TF-IDFVectorizer**

```
#### CONTENT-BASED
# mendefinisikan book_author sebagai dasar konstruk matrix
book_author = buku_baru['author']
# inisialisasi tfidfvectorisasi
tfidf = TfidfVectorizer()
# membangun matrix TF-IDF
tfidf_matrix = tfidf.fit_transform(book_author)
# membuat matrix cosine similarity
cosine_sim = cosine_similarity(tfidf_matrix, tfidf_matrix)
# memapping antara judul dengan index
meta = pd.Series(buku_baru.index, index=buku_baru['judul'])
# membuat array untuk judul buku
semua_judul = [buku_baru['judul'][j] for j in range(len(buku_baru['judul']))]
```

Lalu pada proses line ke-3 pada block of code `Tfidfvectorizer`. Kami membuat variabel bernama `meta` yang digunakan untuk memapping kolom judul

dengan kolom index dataset. Dengan cara memberikan metode index pada variabel buku\_baru dimana index berisi index pada kolom judul. fungsi memapping judul dengan index dimaksudkan untuk mengsortir data agar berurutan. Kami menggunakan pandas.series yang digunakan untuk mengurutkan judul dengan index dataset.

**Gambar 3.16 Mapping antara judul dengan index**

```
# memapping antara judul dengan index
meta = pd.Series(buku_baru.index, index=buku_baru['judul'])
```

Lalu pada kolom line ke-4 pada block of code yang sama, kami membuat array untuk kolom judul buku yang dinamakan variabel semua\_judul, fungsi variabel ini dimaksudkan untuk tahap deployment dimana ketika pengguna memasukkan judul buku yang disukai pada box pencarian akan memberikan perintah untuk variabel semua\_judul untuk mencari buku yang berjudul sama dengan entri yang ada di dataset.

**Gambar 3.17 Array untuk judul buku dengan iterasi guna untuk mencari kecocokan dengan *Input* data yang dimasukan oleh *user***

```
# membuat array untuk judul buku
semua_judul = [buku_baru['judul'][j] for j in range(len(buku_baru['judul']))]
```

Pada proses nya kami membuat iterasi untuk penyelesaian masalah pencarian judul buku dari input yang dibuat oleh pengguna kepada dataset terutama pada kolom judul di variabel buku\_baru serta menggunakan parameter len() untuk memberitahu iterasi bahwa iterasi yang diterapkan sesuai dengan jarak index dari kolom judul pada variabel buku\_baru.

Lalu kami membuat block of code baru atau line of code baru untuk menginisialisasi fungsi dengan nama variabel Get\_title() guna untuk mengambil rekomendasi dari input user.

**Gambar 3.18 Model Content-Based dengan TF-IDF**

```
# TF-IDF recommender (content-based)
def get_title(judul):
    global skor_sim
    index = meta[judul]
    skor_sim = list(enumerate(cosine_sim[index]))
    skor_sim = sorted(skor_sim, key=lambda x: x[1], reverse=True)
    skor_sim = skor_sim[1:6]
    for j in skor_sim:
        print(j)
    book_beta = [j[0] for j in skor_sim]
    # membuat hasil keluaran dari get_title() function kedalam pandas dataframe (fetch)
    balikin_df = pd.DataFrame(columns=['Judul', 'Penulis', 'Nilai', 'ISBN', 'Skor']) # columns = [0],[1],[2]
    balikin_df['Judul'] = buku_baru['judul'].iloc[book_beta]
    balikin_df['Penulis'] = buku_baru['author'].iloc[book_beta]
    balikin_df['ISBN'] = buku_baru['id'].iloc[book_beta]
    balikin_df['Nilai'] = buku_baru['nilai'].iloc[book_beta]
    balikin_df['Skor'] = skor_sim
    return balikin_df
```

Pada block-of-code fungsi Get\_title ini kami membuat variabel index guna untuk menghubungkan index buku dengan judul buku dengan nama index dimana berisi variabel meta yang telah dibuat pada line ke-3 di block-of-code Tfidfvectorizer().

**Gambar 3.19 Inisialisasi index dari variable meta dengan parameter judul yang terdapat dalam parameter fungsi get\_title()**

```
index = meta[judul]
```

Lalu kami membuat variabel skor\_sim pada fungsi Get\_title() untuk membuat list dari function enumerasi yang digunakan untuk menghitung jumlah iterasi yang dihasilkan dari index dengan variabel cosine\_sim yang berisi matriks cosine similarity dengan vektorisasi tf-idf yang sudah di fitting dengan penulis buku.

**Gambar 3.20 Enumerasi dengan cosine\_sim parameter index menjadi list**

```
skor_sim = list(enumerate(cosine_sim[index]))
```

Lalu membuat fungsi sorted dari variabel skor\_sim yang sebelumnya sudah di definisikan dengan list enumerasi dan membalikan fungsi sorted dengan reverse agar membuat serta memberikan variabel key dengan fungsi lambda untuk membantu penyortiran data skor\_sim dengan variabel x.

**Gambar 3.21 Sorted**

```
skor_sim = sorted(skor_sim, key=lambda x: x[1], reverse=True)
```

dan pada line ke-4 pada block of code definisi fungsi Get\_title() saya mengatur skor similarity yang keluar dari index 1 hingga 6.

**Gambar 3.22 Similarity Score**

```
skor_sim = skor_sim[1:6]
```

Pada line ke-5 pada block-of-code definisi fungsi Get\_title() saya membuat variabel book\_beta dengan iterasi pada skor\_sim yang guna untuk menjabarkan skor similarity. Guna untuk memasukan kedalam dataframe balikin\_df dimana dataframe ini guna untuk memasukan data yang telah di temukan kemiripan nya dari input atau masukan yang dimasukan oleh user.

**Gambar 3.23 Iterasi untuk skor\_sim**

```
book_beta = [j[0] for j in skor_sim]
```

Pada line ke-6 hingga 10 guna untuk memasukan atau mengembalikan data yang telah di temukan kesamaanya kedalam dataframe bernama balikin\_df yang berguna untuk menampilkan hasil yang telah direkomendasikan dari fungsi get\_title() dengan tampilan tabel pandas

**Gambar 3.24 Membuat Dataframe kosong untuk memasukan hasil fungsi Get\_title().**

```
# membuat hasil keluaran dari get_title() function kedalam pandas dataframe (fetch)
balikin_df = pd.DataFrame(columns=['Judul','Penulis','Nilai','ISBN','Skor']) # columns = [0],[1],[2]
balikin_df['Judul'] = buku_baru['judul'].iloc[book_beta]
balikin_df['Penulis'] = buku_baru['author'].iloc[book_beta]
balikin_df['ISBN'] = buku_baru['id'].iloc[book_beta]
balikin_df['Nilai'] = buku_baru['nilai'].iloc[book_beta]
balikin_df['Skor'] = skor_sim

return balikin_df
```

Pengaplikasian *cosine similarity* pada proyek kami dipakai untuk mencari kecocokan nama penulis dari judul buku yang dimasukan, membuat mesin merkomendasikan berdasarkan nama penulis dengan judul yang berbeda.

*Cosine similarity* pada permodelan yang kami buat ditujukan untuk memberikan skor kemiripan antara input nama penulis dari judul buku yang dimasukan oleh user dengan nama penulis buku dari dataset kami.

Semakin besar kemiripan angka nya maka semakin mendekati dengan nama penulis dari judul buku yang direkomendasikan.

**Gambar 3.25 Hasil *Similarity Score* dengan bookId sebagai Index**

```

1 # Rekomendasi judul
2 get_title('Ayat-Ayat Cinta')
[12]
...
bookId    score
(90, 1.0)
(146, 1.0)
(147, 1.0)
(148, 1.0)
(175, 1.0)
(1, 0.0)
(2, 0.0)
(3, 0.0)
(4, 0.0)

90          Di Atas Sajadah Cinta
146          Bumi Cinta
147          Pudarnya Pesona Cleopatra
148  Ketika Cinta Bertasbih: Episode 1
175  Ketika Cinta Bertasbih: Episode 2
1          Supernova: Petir
2          Islam dan Sosialisme
3          Jomblo: Sebuah Komedi Cinta
4          Gadis Pantai
Name: judul, dtype: object

```

### Tahap Evaluasi:

Pada tahap evaluasi, pada dasar nya kami menggunakan metriks skor cosine similiarty namun untuk memperjelas kami menambahkan metriks Precision Top-N dimana rumus nya sebagai berikut [8]:

$$\text{Precision} = ((K \text{ dari Rekomendasi Relevant}) / (K \text{ dari item yang direkomendasikan}) \times 100\%$$

Pada permodelan ini kita hanya memanggil sekitar 5 skor similarity atau 5 judul buku dari function `get_title()` dimana rumus nya menjadi =

$$\text{Precision} = (3) / (5) \cdot 100\% = 60\%$$

kita mendapatkan 60% karena model yang saya buat telah diatur untuk 5 buku dari rekomendasi berdasarkan kedekatakan nama penulis dengan input judul buku yang dicocokkan dengan penulis buku [8]. Serta mengapa K dari Rekomendasi Relevant saya masukan 3 karena pada proses nya item dengan skor similarity tertinggi ada di tiga teratas.

**Gambar 3.26 Skor Similarity 3 dari 5 buku yang direkomendasikan dimana terdapat nama judul dari hasil pencarian.**

Penulis	Skor Similarity
Ahmad Fuadi	1.0
Ahmad Fuadi	1.0
Ahmad Tohari	0.4216597657539667
Ahmad Tohari	0.4216597657539667
Ahmad Tohari	0.4216597657539667

#### **Tahap Deployment:**

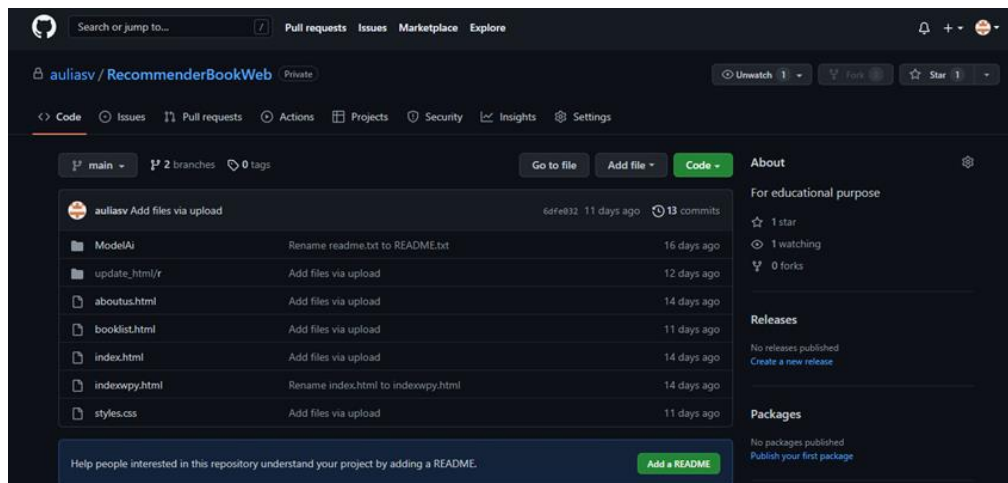
Tahap selanjutnya setelah menyelesaikan permodelan dan evaluasi dari model yang ada, kita mencoba untuk mengintegrasikan model yang kita buat menjadi aplikasi berbasis *web*, yaitu *Deployment*. Tujuan dari tahap *deployment* ini adalah untuk mengelola yang berkaitan dengan aksesibilitas dan ketersediaan aplikasi *web* yang sudah dibuat untuk pengguna melalui internet [9]. Selain itu, tahap ini merupakan salah satu kemampuan penting di bidang teknologi yang tercantum pada Strategi Nasional Kecerdasan Artifisial Indonesia Tahun 2020 - 2045 yang dicanangkan oleh Badan Pengkajian dan Penerapan Teknologi Indonesia. Oleh karena itu, tahap *deployment* merupakan tahap penting, bukan hanya dalam *AI cycle* saja, tetapi dalam bidang pengembangan teknologi, khususnya pada aplikasi *software*.

Pada tahap *deployment project* ini, kami terlebih dahulu melakukan perencanaan desain *interface* dengan melakukan diskusi dengan teman sekelompok untuk menentukan tema aplikasi, *tools* yang ingin ditampilkan, menu yang disediakan. Menurut Leavitt & Shneiderman [10], rancangan desain *user interface* bertujuan untuk memastikan hasil terbaik kepada pengguna. Jika rancangan desain sudah disepakati antar anggota kelompok, kemudian anggota yang bertugas untuk melakukan *web programming* aplikasi melakukan tugasnya untuk membuat halaman *web* untuk aplikasi AI yang sudah kami buat. Tidak ada kendala yang berarti dalam proses ini, tetapi dalam proses pembuatannya terdapat beberapa bagian yang harus diperbaiki ulang agar dapat menyesuaikan dengan konten aplikasi *recommender system* yang kami sudah kembangkan, sehingga pengerjaannya pun relatif lancar dan memakan waktu sekitar  $\pm 7$  hari saja.

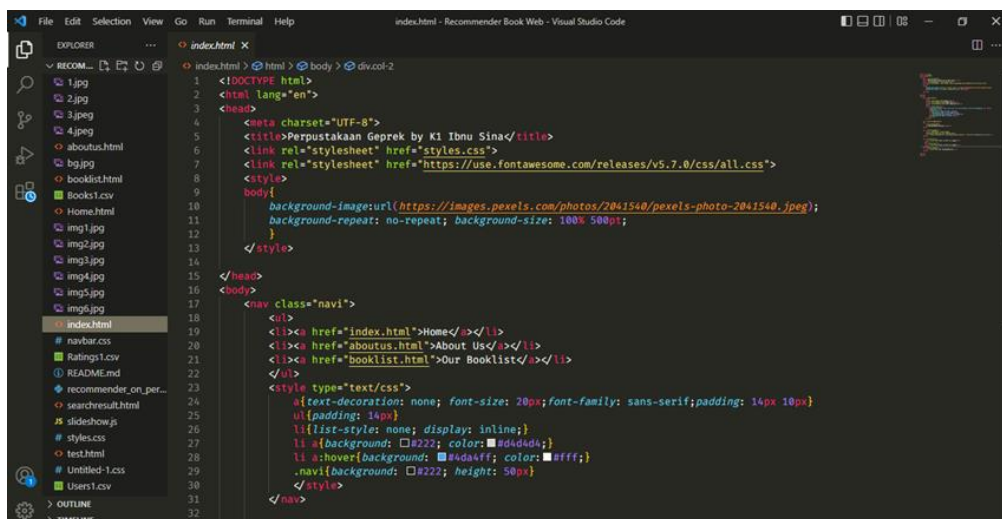
*Tools* yang kami gunakan pada tahap ini adalah Visual Studio Code dan GitHub. Visual Studio Code merupakan *tools text editor* berbasis *open source* yang digunakan untuk mengembangkan aplikasi dan didukung oleh ada 30 bahasa pemrograman, *markup*, hingga basis data yang berbeda [11]. Kami menggunakan Visual Studio Code untuk mengembangkan *web* dengan bahasa *markup*, yaitu HTML (HyperText Markup Language) sebagai *templating* dari *web* dan CSS (Cascading Style Sheets) untuk *styling* dari *templating web* yang dibuat menggunakan HTML. Selain itu, kami menggunakan GitHub untuk menyimpan kode HTML & CSS serta aset yang telah kami buat menggunakan Visual Studio Code. Selain itu, GitHub juga memberikan lingkungan dinamis dan kolaboratif kepada *developer*, sehingga disebut dengan *social coding platform* karena selain untuk penyimpanan kode, juga mendukung *peer-review*, komentar, hingga diskusi dan kolaborasi antar *developer* [12]. Oleh karena itu, *tools* tersebut sangat cocok untuk mendukung kerja tim untuk pembuatan *project* akhir kelompok kami.



**Gambar 3.27 Tampilan Repository GitHub**



**Gambar 3.28 Tampilan Text Editor Visual Studio Code**



Jika tahap pengembangan *web* selesai dilakukan, maka tahap selanjutnya adalah tahap pengintegrasian antara *user interface* laman aplikasi yang berisikan kode HTML & CSS dengan aplikasi *recommender system* yang telah dibuat pada tahap *modelling* yang berisi *code* Python yang merupakan konten utama dalam halaman *web* yang telah dikerjakan.

Pada tahap pengintegrasian ini, kami menggunakan Flask, *micro-framework* berbasis Python untuk membantu pengintegrasian sistem yang telah kami buat. Flask merupakan *tools* yang berperan sebagai *framework* dari aplikasi dan tampilan *web*, sehingga pengembang dapat membuat *web* yang terstruktur dan mengatur *behavior web* dengan mudah [13]. Kendala yang kami jumpai pada tahap ini adalah adanya usaha, kemampuan, serta pemahaman yang lebih untuk mempelajari lebih rinci mengenai dasar-dasar mekanisme kerja *web development* yang tidak begitu diulas mendalam pada pembelajaran di program ini. Dengan memperhatikan kendala kami tersebut, maka kami memutuskan untuk menggunakan Flask yang merupakan *micro-framework* berbasis Python yang cocok untuk pemula *web development* karena proses instalasi, penggunaan, dan fitur yang disediakan sudah cukup mengakomodir kebutuhan kami untuk melakukan pengintegrasian sistem aplikasi *recommender system*.

### Proses pengintegrasian Model AI kepada Flask:

Kami menggunakan file `app.py` untuk membuat struktur awal framework dengan memasukan logika python untuk menampilkan halaman dan menampilkan hasil dari permodelan menggunakan metode request POST dan Iterasi.

**Gambar 3.29 Framework Flask yang sudah di integrasikan dengan logika python**

```

1 from flask import Flask, render_template, request, url_for
2
3 from content import content
4 from content import semua_judul, get_title
5
6 app = Flask(__name__)
7
8 app.register_blueprint(content)
9
10 #inisialisasi aplikasi dan logik output dengan metode GET dan POST
11 @app.route('/', methods=['GET', 'POST'])
12 @app.route('/home', methods=['GET', 'POST'])
13
14 def home():
15
16     if request.method == 'GET':
17         return render_template('index.html')
18
19     if request.method == 'POST':
20         buku_title = request.form['tit']
21
22         if buku_title not in semua_judul:
23             gakada = "Book not found"
24
25             return render_template('notfound.html', user_input=buku_title, notfound=gakada)
26
27         else:
28             hasil_cari = get_title(buku_title)
29             namabuku = []
30             penulis = []
31             penilaian = []
32             isbn = []
33             skor = []
34             for i in range(len(hasil_cari)):
35                 namabuku.append(hasil_cari.iloc[i][0])
36                 penulis.append(hasil_cari.iloc[i][1])
37                 penilaian.append(hasil_cari.iloc[i][2])
38                 isbn.append(hasil_cari.iloc[i][3])
39                 skor.append(hasil_cari.iloc[i][4])
40
41             return render_template('found.html', namabuku=namabuku, penulis=penulis, user_input=buku_title, skor=skor)
42

```

Permodelan Ai yang dimasukan kedalam Content.py guna untuk menjadikan model sebagai API dengan metode Blueprint dari modul flask

**Gambar 3.30 Memasukan penulisan kode python kedalam content.py sebagai API untuk app.py**

```
content.py X app.py
1 from flask import Blueprint
2
3 content = Blueprint('content', __name__)
4
5
6 import pandas as pd
7 import numpy as np
8 # library cosine_similarity
9 from sklearn.metrics.pairwise import cosine_similarity
10 from sklearn.feature_extraction.text import TfidfVectorizer
11 #Load dataset
12 books = pd.read_csv('dataset/Books1.csv',encoding='latin1')
13 ratings = pd.read_csv('dataset/Ratings1.csv')
14 #diambil dari index 0 sampai 1000
15 book = books[0:1001]
16 rating = ratings[0:1001]
17 ### Data Preparation (Content Based)
18 book = book.drop(columns=['Image-URL-S','Image-URL-M','Image-URL-L'], axis=1)
19 rating.groupby('ISBN').sum()
20 semua_buku = pd.merge(rating, book[['ISBN','Book-Title','Book-Author']], on='ISBN', how='left')
21 ### Data Preprocessing
22 bersihkan_buku = semua_buku.dropna()
23 bersihkan_buku.reset_index(drop=True,inplace=True)
24 persiapandata = bersihkan_buku.sort_values('ISBN',ascending=False)
25 dataduplikat = persiapandata.drop_duplicates('ISBN')
26 rating_buk = dataduplikat['Book-Rating'].tolist()
27 id_buku = dataduplikat['ISBN'].tolist()
28 author_buku = dataduplikat['Book-Author'].tolist()
29 judul_buku = dataduplikat['Book-Title'].tolist()
30 buku_baru = pd.DataFrame({
31     'id' : id_buku , #ISBN [0]
32     'judul' : judul_buku , # [1]
33     'author' : author_buku , # [2]
34     'nilai' : rating_buk # [3]
35 })
36
37 ### CONTENT-BASED
38 # mendefinisikan book_author sebagai dasar konstruk matrix
39 book_author = buku_baru['author']
40 # inialisasi tfidfvectorisasi
41 tfidf = TfidfVectorizer()
42 # membangun matrix TF-IDF
43 tfidf_matrix = tfidf.fit_transform(book_author)
```

Tahap ini dapat dikatakan selesai jika konten dari *web* yang sudah dibuat akan menampilkan aplikasi *recommender system* yang telah dikerjakan pada tahap *modelling*. Tahap selanjutnya adalah melakukan *web hosting*. *Web hosting* merupakan suatu kegiatan untuk menyediakan ruang penyimpanan laman *web* untuk individu atau organisasi agar dapat diakses melalui *World Wide Web* (www) [14].

Setelah selesai melakukan pengintegrasian menggunakan Flask dan semua kode serta aset disimpan dalam repositori GitHub, maka kita membutuhkan *tools* untuk melakukan *web hosting* agar *web* yang sudah dikembangkan dapat diakses melui internet. *Tools* yang kami gunakan dalam melakukan *web hosting* adalah Heroku dan Git Bash. Pada dasarnya, Heroku adalah *platform* untuk *web developer*

yang membantu penggunaanya dalam mengelola aplikasi *web* dengan mudah, tanpa terganggu oleh perangkat keras atau server [15]. *Deployment* pada tahap *web hosting* dengan Heroku dapat menggunakan Git Bash. Git Bash adalah aplikasi untuk sistem operasi Windows yang menyediakan emulasi untuk perintah Git dan merupakan paket instalasi Bash, utilitas Bash umum, yang berguna untuk berinteraksi dengan sistem operasi melalui perintah tertulis [16]. Kendala yang dihadapi pada tahap ini adalah adanya limitasi terhadap ukuran *file* yang dapat diunggah pada repositori (maks. 25MB), sehingga kelompok kami harus mencari solusi dan akhirnya kami memutuskan untuk menggunakan *tools* tambahan berupa Git Bash untuk dapat melakukan *deployment* dengan Heroku. Tahap *web hosting* merupakan tahap terakhir dari *AI project cycle* yang dilakukan oleh kelompok kami. Aplikasi yang kami kembangkan sudah dapat dikatakan selesai, jika sudah dapat diakses pada *World Wide Web* (www) dengan pranala berikut: <https://literature-recommender-ibn-1.herokuapp.com/>

### **Hambatan dan Solusi dalam Proses Pengerjaan Proyek Akhir**

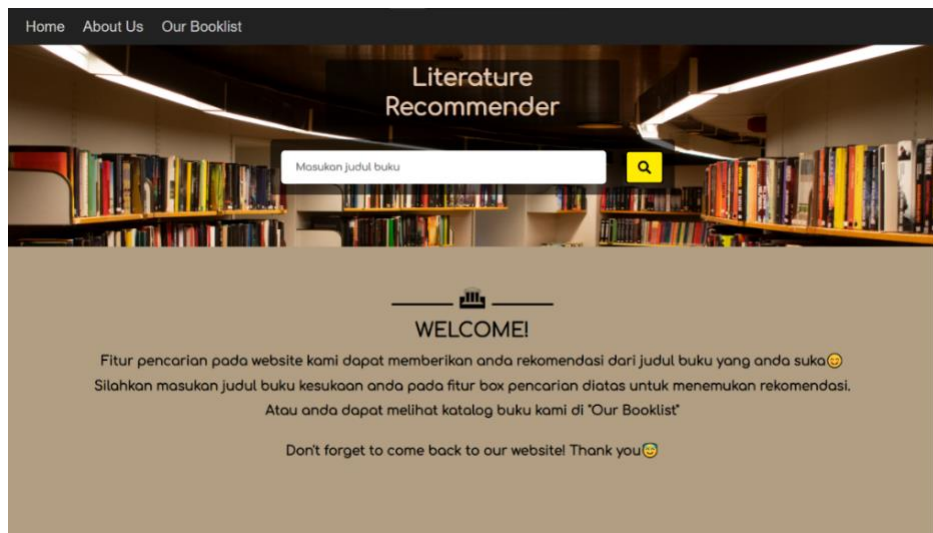
Hambatan kami dalam mengerjakan Proyek Akhir *Sistem Rekomendasi Buku Perpustakaan* ini adalah ketika melakukan pemodelan, sebelumnya kami menggunakan 2 model yaitu, *Content Based* dan *Collaborative*. Tetapi, jika menggunakan kedua model ini maka ukuran *file deployment* nya menjadi sangat besar. Sehingga, kami tidak dapat melakukan proses *Deployment* ke Heroku karena pada Heroku terdapat batas maksimal ukuran *file deployment* untuk dilakukan *deployment*. Dari masalah tersebut, akhirnya kami berdiskusi dan mempertimbangkan yang terbaik untuk Proyek Akhir ini. Sehingga, solusi dari permasalahan tersebut adalah kita hanya menggunakan satu model yaitu *Content Based* untuk memudahkan pengerjaan Proyek Akhir ini.

### **III.3 Hasil Proyek Akhir**

Website kami memberikan rekomendasi dari judul buku yang pengguna sukai. Pengguna dapat memasukkan judul buku yang pengguna inginkan melalui fitur Box pencarian pada menu “Home” untuk mendapatkan rekomendasi buku

yang sesuai dengan keinginan pengguna. Atau pengguna dapat melihat katalog buku kami pada menu “Our Booklist”. Website kami dapat diakses melalui link sebagai berikut: <https://literature-recommender-ibn-1.herokuapp.com/home>

**Gambar 3.31** Tampilan *home* aplikasi literature recommender



**Gambar 3.32** Fitur *recommender system*

Home About Us Our Booklist

# Literature Recommender

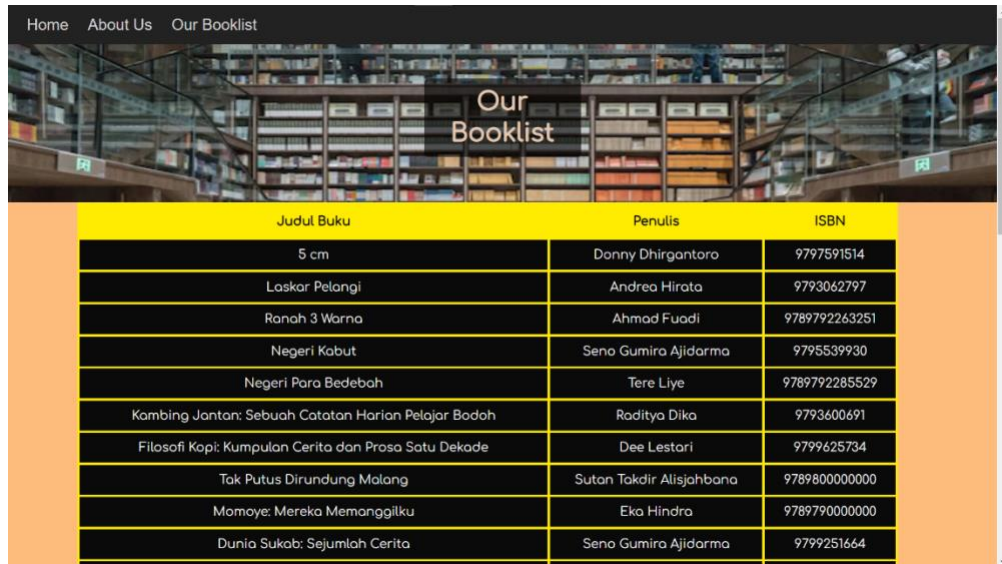
Masukan judul buku

Recommendation for Negeri 5 Menara

Judul Buku	Penulis	Skor Similarity
Ranah 3 Warna	Ahmad Fuadi	1.0
Negeri 5 Menara	Ahmad Fuadi	1.0
Kubah	Ahmad Tohari	0.4216597657539667
Bekisar Merah	Ahmad Tohari	0.4216597657539667
Ronggeng Dukuh Paruk	Ahmad Tohari	0.4216597657539667

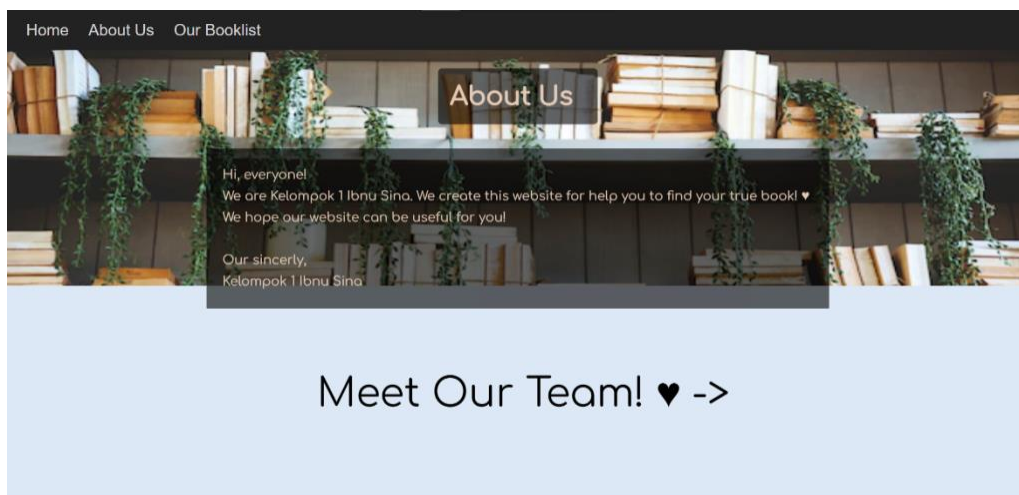


**Gambar 3.33 Tampilan *our booklist***



Judul Buku	Penulis	ISBN
5 cm	Donny Dhirgantoro	9797591514
Laskar Pelangi	Andrea Hirata	9793062797
Ranah 3 Warna	Ahmad Fuadi	9789792263251
Negeri Kabut	Seno Gumira Ajidarma	9795539930
Negeri Para Bedebah	Tere Liye	9789792285529
Kombing Jantang: Sebuah Catatan Harian Pelajar Bodoh	Raditya Dika	9793600691
Filosofi Kopi: Kumpulan Cerita dan Prosa Satu Dekade	Dee Lestari	9799625734
Tak Putus Dirundung Malang	Sutan Takdir Alisjahbana	9789800000000
Momoye: Mereka Memanggilku	Eka Hindra	9789790000000
Dunia Sukob: Sejumlah Cerita	Seno Gumira Ajidarma	9799251664

**Gambar 3.34 Tampilan *about us***



### **Kelebihan dan Kelemahan Website**

#### **Kelebihan :**

- Sistem dapat merekomendasikan buku yang belum pernah dibaca.
- Memudahkan pengguna untuk mencari buku yang ingin dibaca selanjutnya dengan memasukan judul buku kesukaan.

- Sistem rekomendasi buku otomatis dan sangat langka atau jarang ada di Indonesia terutama pada perpustakaan.

#### **Kekurangan:**

- Sistem pencarian masih tergolong Case-Sensitive jadi kesalahan dalam penulisan judul buku, seperti huruf kapital dan spasi dapat mempengaruhi sistem dalam memberikan rekomendasi
- Jumlah data buku yang sedikit karena alasan waktu pengerjaan yang terbatas maka sistem, dalam memproses rekomendasi masih bisa dikatakan terbatas.
- Karena alasan waktu, pengerjaan presisi pada tahap evaluasi sistem dalam merekomendasi masih terbilang rendah.

#### **Rencana Pengembangan Website**

Jika kami memiliki kesempatan untuk mengembangkan website ini, kami akan melakukan pengembangan website yaitu dengan menambahkan model pada pemodelan. Misalnya menggunakan dua model, yaitu *Content Based* dan *Collaborative Filtering*. Tetapi, untuk saat ini website kami hanya menggunakan satu model yaitu *Content Based* saja karena terdapat kendala seperti terbatasnya waktu untuk pengerjaan Proyek Akhir ini. Lalu, kami juga bisa menambahkan jumlah entri buku agar daftar buku rekomendasi pada website kami memiliki jumlah yang lebih banyak serta memiliki beragam jenis buku yang sesuai dengan kebutuhan pengguna. Dan kami akan melakukan pengembangan agar sistem rekomendasi kami tidak hanya berupa website tapi bisa juga berupa aplikasi (*web app*) pada mobile phone.

**IV.1 Kesimpulan**

Berdasarkan dari penjelasan mengenai Proyek Akhir kami diatas, adapun kesimpulan yang didapat adalah:

1. Program Magang dan Studi Independen Bersertifikat adalah sebagai wadah mahasiswa untuk mengembangkan dan meningkatkan skillnya karena *Artificial Intelligence* (AI) dan teknologi sangat dibutuhkan dimasa kini maupun masa depan. Sehingga, kami menuangkan ide dan gagasan kami yang berguna bagi masyarakat serta berhubungan dengan *Artificial Intelligence* (AI) dan teknologi.
2. Kami mengerjakan Proyek akhir melalui beberapa tahapan dalam AI Project Cycle antara lain: tahap problem scoping, tahap data acquisition, tahap data exploration, tahap modelling, tahap evaluation, dan tahap deployment.
3. Proyek Akhir yang kami buat dan kembangkan adalah *Sistem Rekomendasi Buku Perpustakaan* yang berupa website yang bisa diakses oleh siapapun dan dimanapun pengguna berada. Website ini diperuntukkan untuk mereka yang mengalami masalah atau bimbang dalam memilih buku apa yang harus pengguna baca selanjutnya atau buku yang sesuai dengan kebutuhan dan keinginan pengguna. Website kami dapat diakses melalui link sebagai berikut: <https://literature-recommender-ibn-1.herokuapp.com/home>
4. Laporan Akhir sebagai hasil dari pembelajaran selama Studi Independen di Orbit Future Academy. Diharapkan *Sistem Rekomendasi Buku Perpustakaan* ini dapat bermanfaat bagi masyarakat, menjadikan dunia literasi Indonesia dapat berkembang serta minat baca masyarakat Indonesia diharapkan dapat meningkat.



## **IV.2 Saran**

Setelah dilihat dari bab-bab sebelumnya, adapun saran yang diberikan adalah sebagai berikut.

1. Diharapkan website yg kami buat dapat bermanfaat untuk meningkatkan minat baca bagi semua kalangan.
2. Saat melakukan pengujian harus teliti agar tidak terjadi kesalahan seperti hasil output yang error.
3. Pengujian pada model berikutnya dapat dilakukan dengan menggunakan parameter yang berbeda pada model atau algoritma deep learning lainnya , untuk mendapatkan hasil yang maksimal

## Bab V

## Referensi

- [1] A. O. P. Dewi, “Kecerdasan Buatan sebagai Konsep Baru pada Perpustakaan,” *ANUVA*, vol. 4, no. 4, pp. 453–460, 2020, doi: 10.14710/anuva.4.4.453-460.
- [2] K. P. dan K. R. Direktorat Jenderal Pendidikan Tinggi, “Apa itu Studi Independen?,” 2022. <https://kampusmerdeka.kemdikbud.go.id/program/studi-independen/detail> (accessed Jun. 15, 2022).
- [3] F. Yunianto, “Orbit Future Academy Luncurkan Program Pembelajaran Orbit Jago AI,” *Yahoo! Berita*, 2022. [https://id.berita.yahoo.com/orbit-future-academy-luncurkan-program-082845811.html?guccounter=1&guce\\_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce\\_referrer\\_sig=AQAAAse0kDkmBYqIVJfuKAoI9\\_aoaRzq5MERNr62W-e\\_EJfiX4a-uK6AlwQJp\\_FXPA5mhseSy5f7rfEDC1\\_40XIQL020V4d](https://id.berita.yahoo.com/orbit-future-academy-luncurkan-program-082845811.html?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAAse0kDkmBYqIVJfuKAoI9_aoaRzq5MERNr62W-e_EJfiX4a-uK6AlwQJp_FXPA5mhseSy5f7rfEDC1_40XIQL020V4d) (accessed Jun. 15, 2022).
- [4] E. Genesius, E. Devara, T. Rijanandi, and R. B. Riyanto, “Pengembangan Teknologi Rekomendasi Kecerdasan Buatan yang Digunakan pada Perpustakaan,” *J. Informatics Vocat. Educ.*, vol. 4, no. 3, pp. 87–92, 2021.
- [5] D. V. Bagul and S. Barve, “A novel Content-based Recommendation Approach Based on LDA Topic Modeling for Literature Recommendation,” in *Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021*, 2021, pp. 954–961, doi: 10.1109/ICICT50816.2021.9358561.
- [6] M. Nurjannah, Hamdani, and I. Fitri Astuti, “Penerapan Algoritma Term Frequency-Inverse Document Frequency (TF-IDF) untuk Text Mining,” *J. Inform. Mulawarman*, vol. 8, no. 3, pp. 110–113, 2013.
- [7] I. M. Viskha, “Relevansi Koleksi dengan Kebutuhan Informasi Pengguna di UPT Perpustakaan Proklamator Bung Hatta Bukittinggi,” Universitas Sumatera Utara, 2018.

- [8] M. Malaeb, "Recall and Precision at k for Recommender Systems," *Medium*, 2017. [https://medium.com/@m\\_n\\_malaeb/recall-and-precision-at-k-for-recommender-systems-618483226c54](https://medium.com/@m_n_malaeb/recall-and-precision-at-k-for-recommender-systems-618483226c54) (accessed Jun. 15, 2022).
- [9] M. Sidiq, "Software Process," Semarang, 2020. [Online]. Available: [https://repository.dinus.ac.id/docs/ajar/RPL-Software\\_Process.pdf](https://repository.dinus.ac.id/docs/ajar/RPL-Software_Process.pdf).
- [10] A. I. Yunus, "Perancangan Desain User Interface dan User Experience pada Aplikasi SIAKAD dengan Menggunakan Metode User Centered Design (UCD) pada Universitas Islam Negeri Sunan Ampel Surabaya," Institut Bisnis dan Informatika STIKOM Surabaya, 2018.
- [11] T. Kahlert and K. Giza, *Visual Studio Code Tips & Tricks Vol. 1*, 1st ed., vol. 1, no. March. Unterschleißheim: Microsoft, 2016.
- [12] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb, "Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository," in *CSCW '12: Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, 2012, vol. 1, pp. 1277–1286, doi: 10.4018/978-1-4666-2122-0.ch023.
- [13] R. Irsyad, "Penggunaan Python Web Framework Flask Untuk Pemula," *OSF Prepr.*, pp. 1–4, 2018, doi: 10.31219/osf.io/t7u5r.
- [14] Firstsiteguide, *Web Hosting Explained The Beginner's Guide to Small Business Website Hosting*. Los Angeles: firstsiteguide.com, 2015.
- [15] J. Ansaharju, "Improving Software Development with Platform-as-a-Service Product – Using Heroku in Web Application Project," Juuso Ansaharju Improving Software Development with Platform-as-a-Service Product – Using Heroku in Web Application Project Helsinki Metropolia University of Applied Sciences, 2016.
- [16] Atlassian, "Git Bash," *Atlassian*, 2022. <https://www.atlassian.com/git/tutorials/git-bash> (accessed Jun. 12, 2022).

## **Bab VI Lampiran    A. TOR**

### **TERM OF REFERENCE (TOR) STUDI INDEPENDEN BERSERTIFIKAT FOUNDATION OF AI AND LIFE SKILLS FOR GEN-Z DI ORBIT FUTURE ACADEMY**

#### **A. Rincian Program**

Foundation of AI and Life Skills for Gen-Z adalah program pelatihan *Artificial Intelligence* (AI) daring yang bertujuan untuk memperkenalkan teknologi dan perangkat AI kepada pelajar, sehingga diharapkan mereka dapat mengembangkan produk AI yang memiliki dampak sosial. Program ini berfokus pada komponen utama AI, seperti Data Science (DS), Natural Language Processing (NLP), dan Computer Vision (CV). Selain keterampilan AI, pelajar juga akan mendapat *life skills* yang bermanfaat untuk mencari atau menciptakan lapangan kerja.

#### **B. Tujuan Program**

Tujuan yang diharapkan setelah peserta mengikuti program ini:

1. Memiliki wawasan tentang AI dan perkembangannya.
2. Mampu merancang dan mengimplementasikan AI Project Cycle.
3. Mampu menggunakan bahasa pemrograman Python untuk mengembangkan aplikasi AI.
4. Mampu mengembangkan salah satu dari 3 domain AI (DS, NLP, dan CV) hingga tahap *deployment*.
5. Mampu menggunakan *soft skills* dan *hard skills* dalam dunia industri dan lingkungan perusahaan.
6. Mampu mengaplikasikan kiat-kiat yang dibutuhkan seorang wirausahawan yang bergerak di bidang *start-up* dalam mentransformasikan ide ke dalam bentuk produk/jasa sehingga dapat menciptakan peluang bisnis yang terus berinovasi, berevolusi, dan berkelanjutan.

### C. Jadwal dan Tempat Pelaksanaan

Jadwal pelaksanaan tertera dalam tabel berikut:

**Tabel 7.1 Agenda Kelas**

Pukul (WIB)	Durasi (jam)	Aktivitas
08.00 s.d. 11.30	3.5	Kelas Sesi Pagi
13.00 s.d. 16.30	3.5	Kelas Sesi Siang
16.30 s.d. 17.30	1	<i>Self-Study</i>

Kelas akan diselenggarakan secara daring melalui aplikasi *video conference*.

### D. Peserta

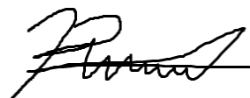
Peserta program ini adalah mahasiswa yang berasal dari Perguruan Tinggi di bawah Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia.

### E. Uraian Tugas Peserta

Selama mengikuti program ini, peserta diharuskan:

1. Mengikuti program dari awal hingga selesai.
2. Mematuhi aturan program.
3. Mematuhi aturan kelas yang dibuat bersama *homeroom* atau *domain coach*.
4. Mengikuti kelas dengan presensi minimal 85%.
5. Membuat laporan harian dan mingguan di *website* Kampus Merdeka.
6. Menyelesaikan Proyek Akhir (PA) beserta laporannya.

Homeroom Coach,



Muhammad Rifal Alfarizy  
NIP: 2201058

Semarang, 16 Juni 2022  
Peserta Program,



Aulia Sava Febiyana  
NIM: 13040119130073

**Bab VII Lampiran      B. Log Activity**

**Tabel 8.1 Aktivitas selama pengerjaan Proyek Akhir**

<b>Minggu/Tgl</b>	<b>Kegiatan</b>	<b>Hasil</b>
25 April 2022 - 27 April 2022	Mencari anggota dan membentuk kelompok untuk Proyek Akhir.	Membentuk Kelompok 1, semua anggota dari kelas Ibnu Sina
28 April 2022 – 8 Mei 2022	Mencari ide dan judul yang tepat untuk Proyek Akhir.	Sistem Rekomendasi Buku Perpustakaan.
9 Mei 2022 - 12 Mei 2022	Membuat dan menyusun problem scoping terkait topik Proyek Akhir.	Problem Scoping berupa Who, What, Where, Why Yang berhubungan dengan topik Proyek Akhir
15 Mei	Melakukan <i>data cleansing</i> dari data yang diperoleh pada <i>data acquisition</i>	Dataset bersih
16 Mei 2022 – 18 Mei 2022	Memberi rating pada masing-masing data buku yang ada Dataset Rating.	Rating untuk setiap data buku yang tersedia.
19 Mei 2022 – 22 Mei 2022	Berdiskusi dengan teman kelompok terkait model yang akan digunakan untuk pemodelan.	Model Content Based yang digunakan untuk Pemodelan.
23 Mei 2022 – 7 Mei 2022	Merancang <i>interface</i> dari aplikasi web <i>recommender system</i>	Tampilan aplikasi web Sistem Rekomendasi Buku Perpustakaan (Literature Recommender) yang menarik.

26 Mei 2022 – 31 Mei 2022	Berkonsultasi dengan Coach Teknikal terkait kendala dalam mengerjakan Proyek akhir serta meminta saran dan sousi untuk Deployment karena terdapat hambatan saat proses pengerjaan bersama kelompok.	Mendapatkan solusi dari masalah yang dihadapi sehingga Kelompok kami bisa melakukan Deployment.
2 Juni 2022 – 15 Juni 2022	Mengerjakan Laporan Proyek Akhir.	Laporan Akhir sudah mulai terisi karena dikerjakan secara konsisten.
16 Juni 2022	Meminta tanda tangan dosen pembimbing kampus maupun mentor dari pihak Orbit Future Academy.	Mendapatkan tanda tangan dosen pembimbing kampus maupun mentor dari pihak Orbit Future Academy.
16 Juni 2022 – 17 Juni 2022	Menyelesaikan dan merevisi Laporan Proyek Akhir.	Laporan Proyek Akhir sudah selesai.

**1. AI Project Cycle**

**a. Problem Scoping**

**Topik yang diambil :**

Penerapan *Recommender System* Buku dalam Merekomendasikan Buku Yang Sesuai Dengan Kebutuhan *User* berdasarkan kemiripan nama penulis buku (Domain: *Natural Language Processing*)

**• WHO - Siapa yang memiliki masalah tersebut?**

- Siapa yang mengalami permasalahan? =  
Pengunjung perpustakaan (pemustaka).
- Siapa sasaran pengguna aplikasi ini?  
Pengunjung perpustakaan (pemustaka) yang mengalami kesulitan dalam menemukan buku yang sesuai dan relevan dengan kebutuhan informasinya.
- Siapa saja pihak yang terlibat dalam proses pembuatan aplikasi ini?  
Perpustakaan, pengunjung perpustakaan (pemustaka), dan pustakawan.

**• WHAT - Apa permasalahan sebenarnya?**

- Apa permasalahan utamanya?
  3. Sulitnya dalam mencari buku yang sesuai dengan kebutuhan.
  4. Kesulitan dalam menentukan pilihan yang tepat sesuai dengan ketertarikan pemustaka.
  5. Pengunjung perpustakaan (pemustaka) kebingungan dalam menentukan buku apa yang akan dibaca.
  6. Tidak adanya petunjuk buku apa saja yang memiliki kualitas yang baik untuk dijadikan pertimbangan pengunjung



perpustakaan (pemustaka) dalam menentukan buku yang akan dibaca.

- Apa saja faktor pendukung permasalahan tersebut?
  4. Pemustaka kurang terampil untuk menentukan buku yang relevan dengan kebutuhan informasinya.
  5. Pemustaka kesulitan dengan proses temu balik pada buku perpustakaan untuk menemukan buku yang sesuai dengan kebutuhannya.

#### **Apa dari peran pihak terkait?**

- Apa peran dari pihak yang terlibat dalam proses pembuatan sistem ini?
  3. Pemustaka merupakan orang yang memiliki masalah dalam melakukan seleksi buku yang ingin digunakan sesuai dengan kebutuhan karena sistem temu balik informasi di perpustakaan yang tidak menunjang mereka untuk mendapatkan bacaan yang baik
  4. Perpustakaan berperan sebagai penyedia jasa layanan informasi harus mampu mengakomodir kebutuhan informasi pemustaka.
  5. Pustakawan sebagai seseorang yang membantu pemustaka dalam menemukan buku yang tepat dan sesuai dengan kebutuhan informasinya.

#### **Apa solusi yang ditawarkan untuk memecahkan masalah tersebut?**

- Apa tujuan dan manfaat dari pembuatan aplikasi ini?
  4. Memudahkan proses temu balik informasi di perpustakaan.
  5. Pemustaka mendapatkan informasi baru dan saran terkait buku apa saja yang memiliki kualitas yang baik dan layak untuk dibaca.

- **WHERE - Dimana/pada saat apa permasalahan ini muncul?**

- Di mana saja calon '*user*' merasakan masalah ini?  
Calon '*user*' merasakan masalah ini ketika mereka berada di perpustakaan.
- Pada kondisi seperti apa calon '*user*' mengalami masalah ini?  
Calon '*user*' mengalami masalah tersebut ketika mereka sedang melakukan temu balik informasi pada koleksi buku milik perpustakaan dan merasa kebingungan untuk menentukan buku mana yang sesuai dengan kebutuhan informasinya.

- **WHY - Mengapa kamu yakin masalah ini sangat penting dibahas?**

- Mengapa permasalahan ini sangat penting?  
Salah satu indikator keberhasilan perpustakaan dari dilihat dari pemanfaatan koleksinya yang sudah sesuai dengan kebutuhan informasi pengguna atau pemustakanya (Viskha, 2018). Oleh karena itu, perpustakaan harus memiliki fasilitas penunjang untuk membantu pemustaka agar dapat mempertemukan buku yang relevan dengan kebutuhan informasinya.
- Apa nilai utama dari permasalahan ini?  
Nilai utama dari permasalahan ini terdapat dalam pembukaan UUD, yaitu memiliki tujuan sesuai dengan poin mencerdaskan kehidupan bangsa, yang mana merupakan kedudukan perpustakaan dalam menyelenggarakan kegiatannya sebagai penyedia layanan informasi tanpa memandang SARA dan status sosial pemustakanya.
- Bagaimana aplikasi ini mampu mengatasi permasalahan tersebut?
  3. Mengembangkan teknologi, misalnya dengan membuat *recommender system* untuk koleksi buku perpustakaan dan mengintegrasikannya dengan OPAC (alat telusur koleksi

perpustakaan) agar memudahkan pemustaka dalam mencari serta menentukan buku yang sesuai dan relevan dengan kebutuhan informasinya.

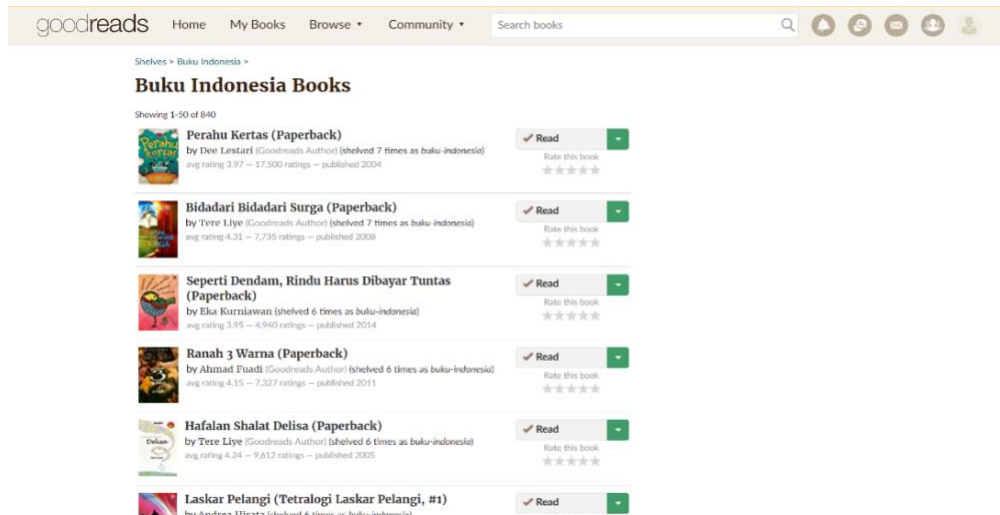
4. Melakukan edukasi mengenai literasi informasi kepada pemustaka dari tahap identifikasi kebutuhan informasi pemustaka hingga tahap pemanfaatan buku yang sesuai tanpa melanggar kaidah hak cipta penerbitan buku tersebut.

#### **a. Data Acquisition**

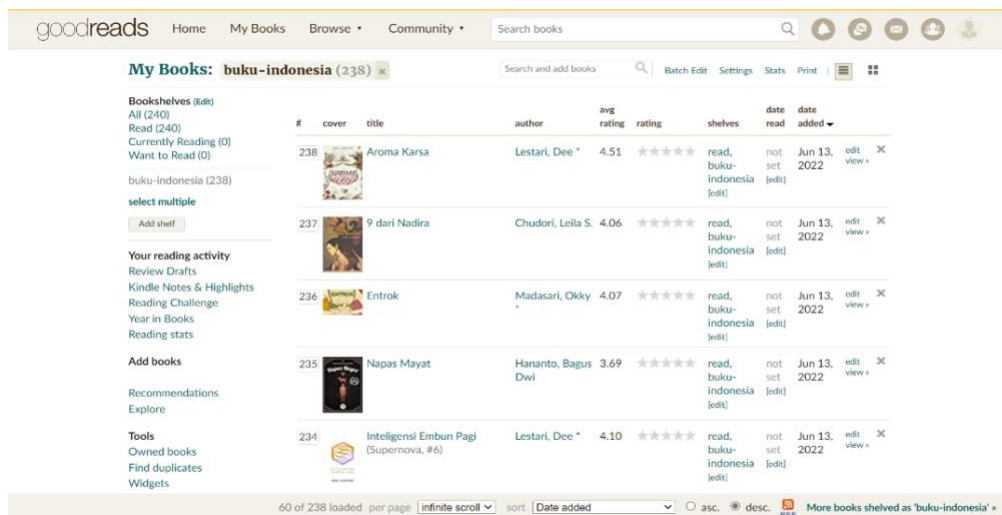
Pada tahap ini, kami mengambil data buku untuk proyek akhir ini dari website Goodreads dengan cara pengambilan manual dimana kami membuat akun pengguna terlebih dahulu untuk memakai fitur yang disediakan oleh goodreads dengan cara menambahkan data buku kedalam *bookshelves* dan mengekspor nya menjadi format CSV (*Comma Seperated Values*) guna keperluan untuk membuat dataset serta dengan menggunakan web-scrapping yaitu memakai ekstensi *Web Scrapper* yang ada di internet browser Google chrome dan Microsoft Edge.

Kami mengambil sekitar 200 data buku Indonesia terutama buku sastra seperti penulis Pramoedya Ananta Toer hingga penulis buku novel karangan Tere liye. Format data yang kami ambil dimulai dari Judul, Penulis, ISBN hingga Rating per-buku.

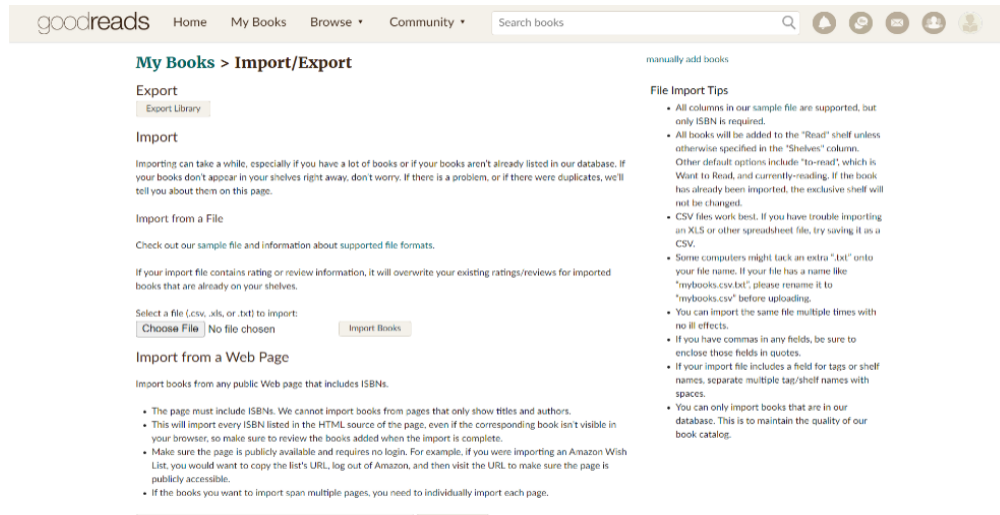
**Gambar 8.1 Tampilan Goodreads dengan target dataset**



**Gambar 8.2 Tampilan Website Goodreads (Book-Shelves)**



**Gambar 8.3 Tampilan Import/Export**



**Gambar 8.4 Tampilan Dataset yang digunakan (.CSV)**



## b. Data Exploration

Pada tahap ini, dataset yang berjumlah 200 data buku akan diproses agar bisa digunakan pada *Sistem Rekomendasi Buku Perpustakaan* yang akan kami buat. Berikut merupakan visualisasi dari dataset yang telah kami proses.

**Gambar 8.5 Tampilan jumlah total dataset**

```
Jumlah ISBN: 890
Jumlah Book-rating 1001

Jumlah judul buku: 987

judul buku:

['Karmila' 'Orang Miskin Dilarang Sekolah!' 'Jendela-Jendela'
'Negeri Kabut' 'Angin Pun Berbisik' 'Tesaurus Bahasa Indonesia'
'O Amuk Kapak' 'Tak Putus Dirundung Malang' 'Momoye: Mereka Memanggilku'
'Dunia Sukab: Sejumlah Cerita'
'Pergolakan Pemikiran Islam: Catatan Harian Ahmad Wahib' 'Perantau'
'Cado-cado: Catatan Dodol Calon Dokter'
'Tuhan & Hal-Hal yang Tak Selesai' 'Jazz, Parfum, dan Insiden'
'Arok Dedes' 'Moslem Millionaire'
'Perang Eropa Jilid I (Perang Eropa, #1)' 'Mencari Telur Garuda'
'Azab dan Sengsara' 'Bibir dalam Pispot' 'Katastrofa Cinta'
'Kitab Omong Kosong'
'Masyarakat Sipil Untuk Transformasi Sosial: Pergolakan Ideologi LSM Indonesia'
'Miss Jutek' 'The Da Peci Code' 'Politik Hati Nurani' 'Kun Fayakun'
'Salah Pilih' 'Dian Yang Tak Kunjung Padam' 'Di Bawah Kibaran Sarung'
'Menalar Tuhan' 'Ketika Jurnalisme Dibungkam Sastra Harus Bicara'
'Jomblo: Sebuah Komedi Cinta'
'Bulan Mati di Javasche Oranje (Johana Rijkaard, Buku 1)' 'Cala Ibi'
```

**Gambar 8.6 Tampilan dataset buku**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1001 entries, 0 to 1000
Data columns (total 8 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   ISBN                                  1001 non-null   object
1   Book-Title                           1001 non-null   object
2   Book-Author                          1001 non-null   object
3   Year-Of-Publication                  997 non-null    float64
4   Publisher                            996 non-null    object
5   Image-URL-S                          1001 non-null   object
6   Image-URL-M                          1001 non-null   object
7   Image-URL-L                          1001 non-null   object
dtypes: float64(1), object(7)
memory usage: 62.7+ KB
```

**Gambar 8.7 Tampilan dataset detail buku**

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
0	9789796623051	Karmila	Marga T.	1990.0	Gramedia Pustaka Utama
1	9793457171	Orang Miskin Dilarang Sekolah!	Eko Prasetyo	2004.0	INSISTPress
2	9796953870	Jendela-Jendela	Fira Basuki	2001.0	PT Gramedia Widiasarana Indonesia (Grasindo)
3	9795539930	Negeri Kabut	Seno Gumira Ajidarma	1999.0	Grasindo
4	9789791685238	Angin Pun Berbisik	Irwan Dwi Kusnanto	2008.0	Mitra Netra & VHRbook

**Gambar 8.8 Tampilan dataset rating dengan detail**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1001 entries, 0 to 1000
Data columns (total 3 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   User-ID         1001 non-null   int64
1   ISBN            1001 non-null   object
2   Book-Rating     1001 non-null   int64
dtypes: int64(2), object(1)
memory usage: 23.6+ KB
```

	User-ID	ISBN	Book-Rating
0	276001	9789796623051	7
1	276001	9793457171	7
2	276001	9796953870	7
3	276001	9795539930	8
4	276001	9789791685238	8

**Gambar 8.9 Tampilan Error cek pada dataset buku**

```
1 # mengecek jika ada error pada dataset
2 book.isnull().sum()
```

...	ISBN	0
	Book-Title	0
	Book-Author	0
	Year-Of-Publication	4
	Publisher	5
	Image-URL-S	0
	Image-URL-M	0
	Image-URL-L	0
	dtype: int64	



**Gambar 8.10 Tampilan Error cek pada dataset rating**

```
1 # mengecek jika ada error pada dataset
2 rating.isnull().sum()

[7]
... User-ID      0
     ISBN        0
     Book-Rating  0
     dtype: int64
```

Pada tahap Pre-Processing, kami memproses ulang kembali data yang sudah di ambil dengan metode manual dan webscraping sehingga siap digunakan untuk untuk tahap *modelling*.

**Gambar 8.11 Preprocessing Data**

```
10
11 ##### Data Preprocessing
12 # membersihkan kolom semua_buku dengan definisikan kepada berishkan_buku
13 bersihkan_buku = semua_buku.dropna()
14 # mengecek keadaan kolom yang sudah dibersihkan dan mereset urutan
15 bersihkan_buku.reset_index(drop=True,inplace=True)
16 # mensortir nilai pada kolom ISBN pada variabel berishkan_buku
17 persiapandata = bersihkan_buku.sort_values('ISBN',ascending=False)
18 # membersihkan duplikat data pada kolom ISBN
19 dataduplikat = persiapandata.drop_duplicates('ISBN')
20 # mengkonversi row rating menjadi list rating_buku
21 rating_buku = dataduplikat['Book-Rating'].tolist()
22 # mengkonversi row ISBN menjadi list id_buku
23 id_buku = dataduplikat['ISBN'].tolist()
24 # mengkonversi row Book-Author menjadi list author_buku
25 author_buku = dataduplikat['Book-Author'].tolist()
26 # mengkonversi row Book-Title menjadi list judul_buku
27 judul_buku = dataduplikat['Book-Title'].tolist()
28 # membuat key dictionary untuk data id_buku, author_buku, judul_buku
29 buku_baru = pd.DataFrame({
30     'id' : id_buku , #ISBN
31     'judul' : judul_buku ,
32     'author' : author_buku,
33     'penilaian': rating_buku
34 })
```



Pada tahap ini kami mencoba untuk melakukan preprocessing data dengan melakukan pembersihan dataset buku dari index kosong, membersihkan data duplikasi hingga dataset dimuat kembali kepada Pandas Dataframe dengan metode key dictionary yang berfungsi untuk mempermudah penamaan kolom dataset.

### c. Modelling

Pada tahap modelling pada awalnya kami menggunakan 2 permodelan yaitu Vektorisasi TF-IDF dengan cosine similarity sebagai metriknya dan permodelan DNN (Deep Neural Network) yang menggunakan modul Tensorflow untuk membuat permodelannya. Namun karena ada kendala dalam proses deployment, kami memutuskan untuk menggunakan 1 permodelan saja yaitu menggunakan TF-IDF (*Term Frequencies – Inverse Document Frequencies*) dengan metrik Cosine similarity. Permodelan dilakukan sepenuhnya pada aplikasi Vscode menggunakan ekstensi jupyter notebook sebagai IDE python.

### d. Evaluation

Pada tahap evaluasi, pada dasarnya kami menggunakan metrik skor cosine similarity namun untuk memperjelas kami menambahkan metrik Precision Top-N dimana rumusnya sebagai berikut [8]:

$$\text{Precision} = ((K \text{ dari Rekomendasi Relevant}) / (K \text{ dari item yang direkomendasikan})) \times 100\%$$

Pada permodelan ini kita hanya memanggil sekitar 5 skor similarity atau 5 judul buku dari function `get_title()` dimana rumusnya menjadi =

$$\text{Precision} = (3) / (5) \cdot 100\% = 60\%$$

kita mendapatkan 60% karena model yang saya buat telah diatur untuk 5 buku dari rekomendasi berdasarkan kedekatan nama penulis dengan input

judul buku yang dicocokkan dengan penulis buku [8]. Serta mengapa K dari Rekomendasi Relevant saya masukan 3 karena pada proses nya item dengan skor similarity tertinggi ada di tiga teratas.

**Gambar 8.12 Skor Similarity 3 dari 5 buku yang direkomendasikan dimana terdapat nama judul dari hasil pencarian.**

Penulis	Skor Similarity
Ahmad Fuadi	1.0
Ahmad Fuadi	1.0
Ahmad Tohari	0.4216597657539667
Ahmad Tohari	0.4216597657539667
Ahmad Tohari	0.4216597657539667

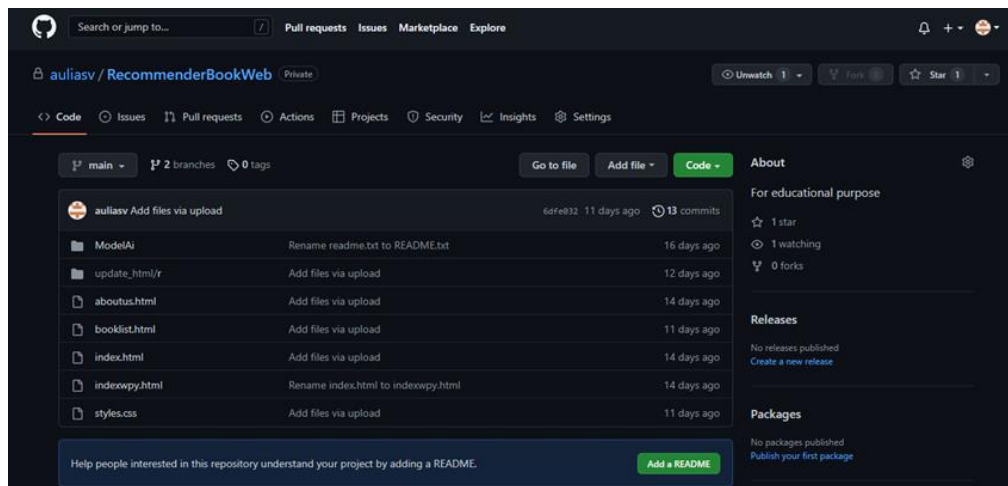
#### e. Deployment

Tahap selanjutnya setelah menyelesaikan permodelan dan evaluasi dari model yang ada, kita mencoba untuk mengintegrasikan model yang kita buat menjadi aplikasi berbasis *web*, yaitu *Deployment*. Tujuan dari tahap *deployment* ini adalah untuk mengelola yang berkaitan dengan aksesibilitas dan ketersediaan aplikasi *web* yang sudah dibuat untuk pengguna melalui internet [9]. Selain itu, tahap ini merupakan salah satu kemampuan penting di bidang teknologi yang tercantum pada Strategi Nasional Kecerdasan Artifisial Indonesia Tahun 2020 - 2045 yang dicanangkan oleh Badan Pengkajian dan Penerapan Teknologi Indonesia. Oleh karena itu, tahap *deployment* merupakan tahap penting, bukan hanya dalam *AI cycle* saja, tetapi dalam bidang pengembangan teknologi, khususnya pada aplikasi *software*.

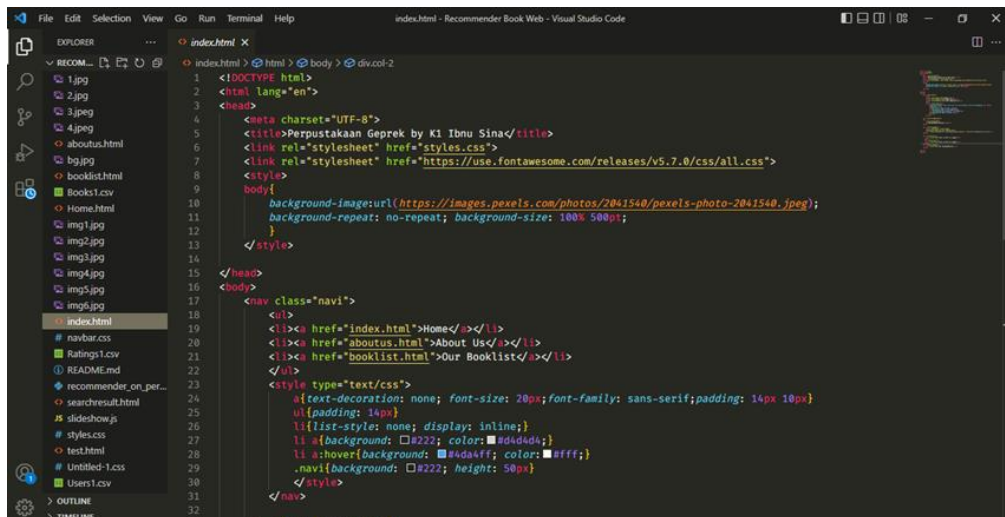
Pada tahap *deployment project* ini, kami terlebih dahulu melakukan perencanaan desain *interface* dengan melakukan diskusi dengan teman sekelompok untuk menentukan tema aplikasi, *tools* yang ingin ditampilkan, menu yang disediakan. Menurut Leavitt & Shneiderman [10], rancangan

desain *user interface* bertujuan untuk memastikan hasil terbaik kepada pengguna. Jika rancangan desain sudah disepakati antar anggota kelompok, kemudian anggota yang bertugas untuk melakukan *web programming* aplikasi melakukan tugasnya untuk membuat halaman *web* untuk aplikasi AI yang sudah kami buat. Tidak ada kendala yang berarti dalam proses ini, tetapi dalam proses pembuatannya terdapat beberapa bagian yang harus diperbaiki ulang agar dapat menyesuaikan dengan konten aplikasi *recommender system* yang kami sudah kembangkan, sehingga pengerjaannya pun relatif lancar dan memakan waktu sekitar  $\pm 7$  hari saja. *Tools* yang kami gunakan pada tahap ini adalah Visual Studio Code dan GitHub. Visual Studio Code merupakan *tools text editor* berbasis *open source* yang digunakan untuk mengembangkan aplikasi dan didukung oleh ada 30 bahasa pemrograman, *markup*, hingga basis data yang berbeda [11]. Kami menggunakan Visual Studio Code untuk mengembangkan *web* dengan bahasa *markup*, yaitu HTML (HyperText Markup Language) sebagai *templating* dari *web* dan CSS (Cascading Style Sheets) untuk *styling* dari *templating web* yang dibuat menggunakan HTML. Selain itu, kami menggunakan GitHub untuk menyimpan kode HTML & CSS serta aset yang telah kami buat menggunakan Visual Studio Code. Selain itu, GitHub juga memberikan lingkungan dinamis dan kolaboratif kepada *developer*, sehingga disebut dengan *social coding platform* karena selain untuk penyimpanan kode, juga mendukung *peer-review*, komentar, hingga diskusi dan kolaborasi antar *developer* [12]. Oleh karena itu, *tools* tersebut sangat cocok untuk mendukung kerja tim untuk pembuatan *project* akhir kelompok kami.

**Gambar 8.13 Tampilan Repository GitHub**



**Gambar 8.14 Tampilan Text Editor Visual Studio Code**



Jika tahap pengembangan *web* selesai dilakukan, maka tahap selanjutnya adalah tahap pengintegrasian antara *user interface* laman aplikasi yang berisikan kode HTML & CSS dengan aplikasi *recommender system* yang telah dibuat pada tahap *modelling* yang berisi *code* Python yang merupakan konten utama dalam halaman *web* yang telah dikerjakan.

Pada tahap pengntegrasian ini, kami menggunakan Flask, *micro-framework* berbasis Python untuk membantu pengintegrasian sistem yang telah kami

buat. Flask merupakan *tools* yang berperan sebagai *framework* dari aplikasi dan tampilan *web*, sehingga pengembang dapat membuat *web* yang terstruktur dan mengatur *behavior web* dengan mudah [13]. Kendala yang kami jumpai pada tahap ini adalah adanya usaha, kemampuan, serta pemahaman yang lebih untuk mempelajari lebih rinci mengenai dasar-dasar mekanisme kerja *web development* yang tidak begitu diulas mendalam pada pembelajaran di program ini. Dengan memperhatikan kendala kami tersebut, maka kami memutuskan untuk menggunakan Flask yang merupakan *micro-framework* berbasis Python yang cocok untuk pemula *web development* karena proses instalasi, penggunaan, dan fitur yang disediakan sudah cukup mengakomodir kebutuhan kami untuk melakukan pengintegrasian sistem aplikasi *recommender system*.

### Proses pengintegrasian Model AI kepada Flask:

Kami menggunakan file `app.py` untuk membuat struktur awal framework dengan memasukan logika python untuk menampilkan halaman dan menampilkan hasil dari permodelan menggunakan metode request POST dan Iterasi.

**Gambar 8.15 Framework Flask yang sudah di integrasikan dengan logika python**

```
1 from flask import Flask, render_template, request, url_for
2
3 from content import content
4 from content import semua_judul, get_title
5
6 app = Flask(__name__)
7
8 app.register_blueprint(content)
9
10 #inisialisasi aplikasi dan logik output dengan metode GET dan POST
11 @app.route('/', methods=['GET', 'POST'])
12 @app.route('/home', methods=['GET', 'POST'])
13
14 def home():
15
16     if request.method == 'GET':
17         return render_template('index.html')
18
19     if request.method == 'POST':
20         buku_title = request.form['tit']
21
22         if buku_title not in semua_judul:
23             gakada = "Book not found"
24
25             return render_template('notfound.html', user_input=buku_title, notfound=gakada)
26
27         else:
28             hasil_cari = get_title(buku_title)
29             namabuku = []
30             penulis = []
31             penilaian = []
32             isbn = []
33             skor = []
34             for i in range(len(hasil_cari)):
35                 namabuku.append(hasil_cari.iloc[i][0])
36                 penulis.append(hasil_cari.iloc[i][1])
37                 penilaian.append(hasil_cari.iloc[i][2])
38                 isbn.append(hasil_cari.iloc[i][3])
39                 skor.append(hasil_cari.iloc[i][4])
40
41             return render_template('found.html', namabuku=namabuku, penulis=penulis, user_input=buku_title, skor=skor)
```

Permodelan Ai yang dimasukan kedalam Content.py guna untuk menjadikan model sebagai API dengan metode Blueprint dari modul flask

**Gambar 8.16 Memasukan penulisan kode python kedalam content.py sebagai API untuk app.py**

```
content.py X app.py
1 from flask import Blueprint
2
3 content = Blueprint('content', __name__)
4
5
6 import pandas as pd
7 import numpy as np
8 # library cosine_similarity
9 from sklearn.metrics.pairwise import cosine_similarity
10 from sklearn.feature_extraction.text import TfidfVectorizer
11 #Load Dataset
12 books = pd.read_csv('dataset/Books1.csv',encoding='latin1')
13 ratings = pd.read_csv('dataset/Ratings1.csv')
14 #diambil dari index 0 sampai 1000
15 book = books[0:1001]
16 rating = ratings[0:1001]
17 ### Data Preparation (Content Based)
18 book = book.drop(columns=['Image-URL-S','Image-URL-M','Image-URL-L'], axis=1)
19 rating.groupby('ISBN').sum()
20 semua_buku = pd.merge(rating, book[['ISBN','Book-Title','Book-Author']], on='ISBN', how='left')
21 ### Data Preprocessing
22 bersihkan_buku = semua_buku.dropna()
23 bersihkan_buku.reset_index(drop=True,inplace=True)
24 persiapandata = bersihkan_buku.sort_values('ISBN',ascending=False)
25 dataduplikat = persiapandata.drop_duplicates('ISBN')
26 rating_buk = dataduplikat['Book-Rating'].tolist()
27 id_buku = dataduplikat['ISBN'].tolist()
28 author_buku = dataduplikat['Book-Author'].tolist()
29 judul_buku = dataduplikat['Book-Title'].tolist()
30 buku_baru = pd.DataFrame({
31     'id' : id_buku , #ISBN [0]
32     'judul' : judul_buku , # [1]
33     'author' : author_buku , # [2]
34     'nilai' : rating_buk # [3]
35 })
36
37 ### CONTENT-BASED
38 # mendefinisikan book_author sebagai dasar konstruk matrix
39 book_author = buku_baru['author']
40 # inialisasi tfidfvectorisasi
41 tfidf = TfidfVectorizer()
42 # membangun matrix TF-IDF
43 tfidf_matrix = tfidf.fit_transform(book_author)
```

Tahap ini dapat dikatakan selesai jika konten dari *web* yang sudah dibuat akan menampilkan aplikasi *recommender system* yang telah dikerjakan pada tahap *modelling*. Tahap selanjutnya adalah melakukan *web hosting*. *Web hosting* merupakan suatu kegiatan untuk menyediakan ruang penyimpanan laman *web* untuk individu atau organisasi agar dapat diakses melalui *World Wide Web* (www) [14].

Setelah selesai melakukan pengintegrasian menggunakan Flask dan semua kode serta aset disimpan dalam repositori GitHub, maka kita membutuhkan *tools* untuk melakukan *web hosting* agar *web* yang sudah dikembangkan dapat diakses melui internet. *Tools* yang kami gunakan dalam melakukan *web hosting* adalah Heroku dan Git Bash. Pada dasarnya, Heroku adalah

*platform* untuk *web developer* yang membantu penggunaanya dalam mengelola aplikasi *web* dengan mudah, tanpa terganggu oleh perangkat keras atau server [15]. *Deployment* pada tahap *web hosting* dengan Heroku dapat menggunakan Git Bash. Git Bash adalah aplikasi untuk sistem operasi Windows yang menyediakan emulasi untuk perintah Git dan merupakan paket instalasi Bash, utilitas Bash umum, yang berguna untuk berinteraksi dengan sistem operasi melalui perintah tertulis [16]. Kendala yang dihadapi pada tahap ini adalah adanya limitasi terhadap ukuran *file* yang dapat diunggah pada repositori (maks. 25MB), sehingga kelompok kami harus mencari solusi dan akhirnya kami memutuskan untuk menggunakan *tools* tambahan berupa Git Bash untuk dapat melakukan *deployment* dengan Heroku. Tahap *web hosting* merupakan tahap terakhir dari *AI project cycle* yang dilakukan oleh kelompok kami. Aplikasi yang kami kembangkan sudah dapat dikatakan selesai, jika sudah dapat diakses pada *World Wide Web* (www) dengan pranala berikut: <https://literature-recommender-ibn-1.herokuapp.com/>

## **2. Profil Tim dan Deskripsi Pembagian Tugas**

- Data akuisisi : Tanty Lianasondari  
Yaitu melakukan pengambilan sumber data menggunakan cara atau metode *scrapping*. Data yang diambil berisi judul buku, penulis, dan ISBN. Data tersebut akan digunakan untuk tahapan proses pembuatan model AI.
- Modelling AI : Rifky Abdul Hakim  
Yaitu melakukan pemodelan, dengan menggunakan satu model (*Content Based*). Dimana sistem merekomendasikan buku dari nama penulis dengan memasukan judul buku untuk permudah pengguna dalam mencari rekomendasi. Kami menggunakan library scikit-learn yaitu model TF-IDF (TfidfVectorizer) dan menggunakan metrik Cosine similarity sebagai parameter untuk menentukan kesamaan skor kedekatan vektor dari satu nama penulis ke nama penulis lainnya yang berada dalam index dataset yang kami gunakan.

- Deployment : Aulia Sava Febiyana & Shofa Mumtazah

Pada tahap *deployment project* ini, kami terlebih dahulu melakukan perencanaan desain *interface* dengan melakukan diskusi dengan teman sekelompok untuk menentukan tema aplikasi, *tools* yang ingin ditampilkan, menu yang disediakan. *Tools* yang kami gunakan pada tahap ini adalah Visual Studio Code dan GitHub.

- Laporan : Shofa Mumtazah & Putri Anggi Nauli

Yaitu menyusun laporan akhir kelompok terkait pengerjaan proyek akhir.

### 3. Deskripsi Aplikasi

#### a. Nama dan Fungsi Aplikasi

Proyek Sistem Rekomendasi Buku Perpustakaan, bertujuan untuk memberikan buku bacaan baru kepada pengunjung perpustakaan yang berdasarkan nama penulis dari judul buku yang disukai.

Pengguna memasukan judul buku yang diminatinya lalu sistem secara otomatis akan memberikan buku dari nama penulis yang serupa atau berbeda.

Kami menamakan *website* kami dengan *Literature Recommender* agar mempermudah dan memperjelas pengunjung perpustakaan yang mengunjungi web kami.

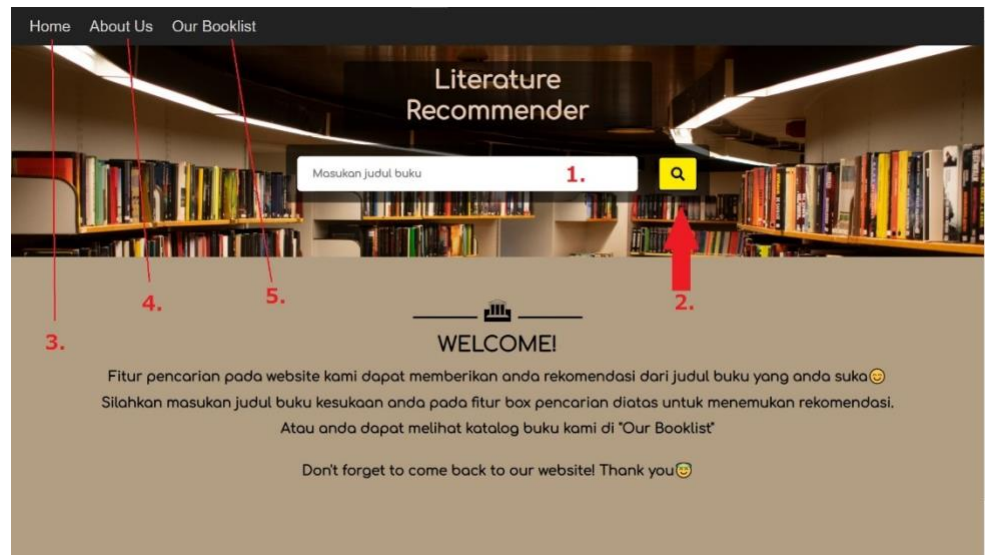
#### b. Jenis Aplikasi dan *Specific Requirement*

Aplikasi kami berjenis *WebApp* dimana aplikasi diintegrasikan dengan halaman website agar bisa diakses dimana saja oleh siapa saja dengan berbagai macam perangkat. Namun, ketika laporan ini dibuat aplikasi masih teroptimalisasi dengan perangkat laptop dan komputer desktop.

#### c. User Interface

**Gambar 8.17 *User interface* web Literature Recommender**





1. = SearchBar, guna untuk mencari rekomendasi dari judul buku yang diminati.
2. = SearchButton, guna untuk melakukan pencarian rekomendasi dari judul yang dimasukan oleh pengguna
3. = NavBar Home, guna untuk kembali kepada halaman utama
4. = NavBar AboutUs, guna untuk menunjukan tim pembuat website
5. = NavBar OurBooklist, guna untuk menunjukan list buku sebagai patokan bagi pengguna untuk mengetahui katalog buku pada web kami.

#### **Kelebihan :**

- Sistem dapat merekomendasikan buku yang belum pernah dibaca.
- Memudahkan pengguna untuk mencari buku yang ingin dibaca selanjutnya dengan memasukan judul buku kesukaan.

#### **Kekurangan:**

- Sistem pencarian masih tergolong Case-Sensitive jadi kesalahan dalam penulisan judul buku, seperti huruf kapital dan spasi dapat mempengaruhi sistem dalam memberikan rekomendasi

- Jumlah data buku yang sedikit karena alasan waktu pengerjaan yang terbatas maka sistem, dalam memproses rekomendasi masih bisa dikatakan terbatas.
- Karena alasan waktu, pengerjaan presisi pada tahap evaluasi sistem dalam merekomendasi masih terbilang rendah.

### **Rencana Pengembangan Website**

Jika kami memiliki kesempatan untuk mengembangkan website ini, kami akan melakukan pengembangan website yaitu dengan menambahkan model pada pemodelan. Misalnya menggunakan dua model, yaitu *Content Based* dan *Collaborative Filtering*. Tetapi, untuk saat ini website kami hanya menggunakan satu model yaitu *Content Based* saja karena terdapat kendala seperti terbatasnya waktu untuk pengerjaan Proyek Akhir ini. Lalu, kami juga bisa menambahkan jumlah entri buku agar daftar buku rekomendasi pada website kami memiliki jumlah yang lebih banyak serta memiliki beragam jenis buku yang sesuai dengan kebutuhan pengguna. Dan kami akan melakukan pengembangan agar sistem rekomendasi kami tidak hanya berupa website tapi bisa juga berupa aplikasi (*web app*) pada mobile phone.