**PROPOSAL SKRIPSI**

**PENERAPAN *STATE BASED CODE EDITOR* PADA**

**SISTEM E – LEARNING BERBASIS WEB**



**Oleh :**

Nama : Satria Efriyadi

NPM : G1A017069

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS BENGKULU**

**2021**

## Judul Penelitian

Penerapan *state based code editor* pada sistem e – learning berbasis web.

## Bidang Ilmu

Bidang ilmu yang penulis akan teliti adalah *sistem e – learning* dan *learning design*.

## Latar Belakang

Konsep Pemrograman adalah hal yang selalu di ajarkan ketika mempelajari hal – hal yang berkaitan dengan dunia teknologi informasi dan sistem informasi, baik melalui pendidikan secara formal maupun nonformal, hal ini tentu saja hal yang lumrah mengingat konsep pemrograman adalah dasar untuk mempelajari bahasa apapun.

Dalam mempelajari suatu bahasa pemrograman, selain mempelajari konsepnya juga perlu mempelajari dasar – dasar fundamental dari bahasa pemrograman tersebut, biasanya bagian ini hanyalah perpanjangan dari konsep pemrograman yaitu membahas bagaimana konsep tersebut di terapkan dalam bahasa pemrograman yang sedang di pelajari.

Seiring dengan perkembangan teknologi, banyak bahasa pemrograman bermunculan sesuai dengan makin dibutuhkannya suatu bidang IT, namun hal ini malah menyebabkan “Demand” akan programmer yang terlalu tinggi di banding jumlah programmer yang tersedia.

Hal ini berdasarkan data dari Peta Okupasi nasional di bidang teknologi informasi dan komunikasi ( TIK ) yang di sampaikan oleh bapak Rudiantara pada tahun 2017 selaku menteri komunikasi dan informatika pada saat itu, Pada tahun 2020 di ajang BAPAREKRAF Developer Day 2020 hal ini kembali di sampaikan kembali bahwa Indonesia masih kekurangan Praktisi IT.

Dapat disimpulkan bahwa sejak 2017 – 2020 jumlah tenaga kerja baru yang bergerak di bidang IT tidak dapat mengimbangi tingginya kebutuhan akan Praktisi IT di indonesia, hal ini dapat di sebabkan oleh berbagai macam faktor dan salah satunya adalah lulusan yang bekerja tidak sesuai dengan bidang yang di ambilnya semasa kuliah.

Penyebab dari lulusan yang bekerja tidak sesuai dengan bidang yang di pelajari di masa kuliah terutama yang mengambil jurusan yang berkaitan dengan informasi dan teknologi bisa dari berbagai macam faktor,contohnya tidak percaya diri dengan skill yang telah di pelajari semasa perkuliahan.

Saat ini banyak sekali layanan E – Learning yang menyajikan materi yang berkaitan dengan teknologi informasi, dengan berbagai jenis pembayaran seperti dari berbayar sampai ke yang gratis, dengan pendekatan metode belajar yang berbagai macam seperti *Passive* *learnin*g, *Active Learning, Adaptive Learning* dan lain sebagainya.

Namun keberagaman metode tersebut tidak menjamin kesuksesan pengguna E – Learning tersebut dikarenakan pada akhirnya ketika sebuah sistem E – Learning telah di desain sedemikian rupa itu tidak ada artinya ketika si pengguna malah tidak mempraktekan ilmu yang telah di pelajarinya.

Hal ini berlaku Bagi semua jenis E – Learning yang memanfaatkan media Bacaan atau Modul, Presentasi maupun dalam bentuk Video, ketika materi disajikan dalam bentuk tersebut tidak jarang yang terjadi adalah pengguna malah masuk ke mode *Passive Learning*.

*Passive Learning* bila di artikan kedalam bahasa indonesia adalah belajar secara pasif, metode ini sering dikaitkan dengan cara belajar yang tidak efektif bila bidang yang di pelajari dengan metode ini berkaitan dengan bidang keilmuan atau kemampuan yang memerlukan kemampuan praktikal.

Contoh sederhananya adalah saat seseorang belajar di bangku sekolah dari TK sampai lulus SMA, sebagian besar ilmu yang didapat pada masa tersebut bisa dipastikan sulit di ingat kembali, terkecuali beberapa hal yang memberikan kesan dan pelajaran yang di praktikan di dalam kelas.

Hal ini juga berlaku kepada gamer E - Sports, karena game pada dasarnya juga memerlukan kemampuan untuk tampil baik dalam permainan yang di mainkan, contohnya adalah seorang pemain game bergenre FPS ( *First Person Shooter )* bernama Valorant, pemain Valorant cenderung melihat berbagai tutorial, Tips dan trik Valorant untuk meningkatkan performa bermainnya.

Namun, yang menjadi pembeda antara seorang pemain yang memiliki peringkat tinggi di valorant dan pemain yang peringkatnya rendah atau setidaknya tidak kunjung mengalami peningkatan adalah seorang pemain berperingkat tinggi tidak hanya menguasai konsep, tapi konsep tersebut di terapkan, di eksplorasi, dan dilatih untuk meningkatkan performa bermainnya.

Kalau tidak di praktekkan, batas pemahaman dari ilmu yang di pelajari sulit di tentukan, permasalahan yang mungkin terjadi saat di terapkan juga menjadi tidak di ketahui, dan pada akhirnya tidak bisa dikatakan menguasai suatu bidang ilmu bila hanya paham konsep tapi minim kemampuan menerapkan konsep tersebut.

Dari analogi permasalahan di atas, diperlukan suatu *Learning Design* yang dapat mendorong murid untuk melakukan praktek ketika mempelajari suatu bahasa pemrograman, sehingga pemahaman yang di dapatkan dari materi dapat terserap dengan efisien.

Berdasarkan paparan diatas, peneliti akan melakukan penelitian dengan topik *“Implementasi State Based Code Editor pada Sistem E – Learning berbasis Web”.* Penelitian ini berfokus pada cara membuat dan mengintegrasikan *State Based Code Editor* ke dalam sebuah *Sistem E – Learning berbasis Website.*

Sistem informasi ini di harapkan dapat meningkatkan kemampuan memahami materi kursus yang di sajikan dalam sebuah Sistem E – Learning dan menyajikan suatu cara menyampaikan materi yang mudah di pahami bagi pengguna sistem.

## Perumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, dapat dirumuskan beberapa permasalahan sebagai berikut:

1. Bagaimana Merancang dan Mengimplementasikan sebuah *State Based Code Editor*.
2. Bagaimana merancang dan mengimplementasikan sebuah *State Based Code Editor* yang dapat terintegrasi dengan Sistem E - Learning.

## Batasan Masalah

Agar aplikasi ini tidak terlalu luas dalam proses pengembangannya, maka peneliti membuat batasan-batasan permasalahan sebagai berikut :

1. Sistem E – Learning yang terintegrasi dengan *State Based Code Editor berbasis Web.*
2. UI/UX Sistem E – Learning di aplikasikan menggunakan HTML,CSS dan JS beserta Library yang bersangkutan.
3. Backend Sistem dibuat menggunakan Node.js serta library Express JS
4. Sistem ini diuji cobakan pada user yang mendaftar pada sistem E - Learning.

## Tujuan Penelitian

Tujuan penelitian yang ingin dicapai dalam penelitian ini adalah merancang dan membangun sistem e - learning yang terintegrasi dengan *state based code editor*.

## Manfaat Penelitian

Manfaat dari penerapan *state based code editor* adalah sebagai berikut :

1. Memudahkan dalam mempraktekkan materi yang sedang di pelajari.
2. Menghindari dari konfigurasi Code editor kompleks yang tidak ramah bagi pemula.
3. Source Code pada Code Editor menjadi dinamis mengikuti sesuai dengan State yang telah di definisikan.
4. Mendukung user untuk mempraktekkan materi yang telah di pelajari.

## Tinjauan Pustaka

### *E - Learning*

*E – Learning* adalah konsep pembelajaran yang memanfaatkan teknologi digital .

### *Passive Learning*

*Passive Learning* adalah metode belajar secara pasif dimana murid belajar dari mendengarkan dan mengobservasi, hal ini memberikan dampak yang buruk karena bagaimanapun juga, menulis kode adalah sebuah kemampuan, dan setiap kemampuan perlu di praktekkan.

### *Active Learning*

*Active Learning* adalah kegiatan belajar apapun dimana setiap murid berpartisipasi atau berinteraksi dengan proses pembelajaran. Metode ini Berlawanan dengan passive learning .

### *Hyperteks Markup Language - Document Object Model (*HTML - DOM*)*

DOM atau *Document Object Model* adalah model standar penulisan sebuah dokumen XML atau HTML*.* DOM yang digunakan sebagai standar penulisan dokumen HTML disebut juga dengan HTML-DOM. HTML-DOM berfungsi untuk mengatur elemen-elemen html disusun memenuhi dan bagaimana untuk mendapatkan, mengubah, menambah, atau menghapus elemen html.

Elemen html adalah kode html yang dimulai dari sebuah *tag* pembuka dan diakhiri dengan *tag* penutup, *tag* itu sendiri adalah kode-kode tertentu yang menjadi pengenal bahwa kode tersebut adalah kode html dan bisa diterjemahkan oleh browser. Ada tiga *tag* utama yang membangun sebuah halaman html, yaitu *tag* html (<html> </html>), *tag head* (<head> </head>) dan *tag body* (<body> </body>) (Mitra et al., 2017).

### *Learning Design*

*Learning Design* adalah kerangka kerja yang mendukung pengalaman belajar, merujuk kepada pilihan yang disengaja tentang apa,kapan,di mana dan bagaimana suatu cara mengajar di terapkan, keputusan yang di perlukan tentang suatu konten ajar, struktur, waktu, strategi pedagogis, urutan kegiatan pembelajaran serta sifat teknologi yang digunakan untuk mendukung suatu pembelajaran.

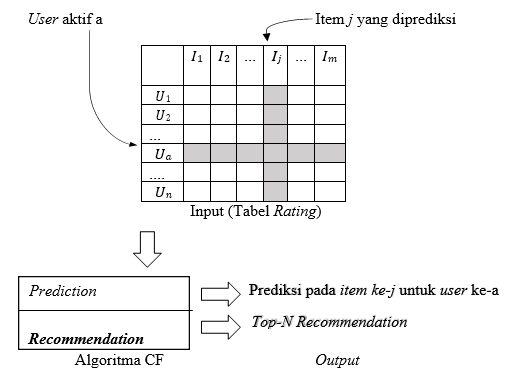
### State Based Code Editor

*State Based Code Editor* adalah Code Editor yang telah didefinisikan *state* atau *keadaan* di dalamnya, dengan adanya state di dalam code editor memungkinkan code editor untuk berubah secara dinamis mengikuti keadaan code editor pada materi suatu course bahasa pemrograman.

### *Collaborative filtering*

*Collaborative Filtering* memberikan rekomendasi berdasarkan kumpulan dari pendapat, minat dan ketertarikan beberapa *user* yang biasanya diberikan dalam bentuk *rating* yang diberikan *user* kepada suatu *item*. Algoritme *collaborative filtering* memiliki dua hal untuk melakukan sistem rekomendasi sebagai berikut (Sabani, 2020) :

1. *Prediction*, merupakan nilai prediksi pada *rating item* yang mungkin disukai oleh *user*.
2. *Recommendation*, memberikan rekomendasi berupa daftar *item* dengan nilai prediksi tertinggi. Dengan catatan bahwa *item* yang direkomendasikan belum pernah dilihat atau di-*rating* oleh user tersebut. Hal ini sering juga disebut *top-N recommendation*.



Gambar 2 Proses Collaborative Filtering (Sabani, 2020)

Gambar 2 menunjukkan skema kerja dari sistem rekomendasi yang menggunakan *collaborative filtering* berbentuk *user-item rating* matriks berukuran m x n, m adalah jumlah *user* U = {u1,u2,u3,...,um}, dan n adalah jumlah *item* I = {i1,i2,i3,...in}. Setiap *user (*ui) mempunyai daftar *item* Iui yang merupakan *rating* dari *user* pada setiap *item*.

Untuk memperoleh data *rating* dari *user* yang digunakan dalam sistem rekomendasi, dibedakan menjadi dua cara yaitu (Wijaya & Alfian, 2018) :

1. Secara *Eksplisit*, yaitu proses pengumpulan data saat user memberikan data secara sadar atau sengaja. Contoh seperti pemberian *rating*, pengisian form tentang data *user*, dan komentar *user* pada suatu *item*.
2. Secara *Implisit*, yaitu proses pengumpulan data saat user tidak menyadari bahwa ia telah memberikan masukan terhadap sistem. Contoh seperti catatan *item* yang diunduh, *item* yang diklik, dan *item* yang dicari *user*.

*Collaborative filtering* pada dasarnya dibagi menjadi dua metode yaitu *user-based collaborative filtering* disebut juga *memory-based*, dan *item-based collaborative filtering* yang disebut juga *model-based*. Pada metode *user based* sistem memberikan rekomendasi kepada *user, item-item* yang disukai atau di-*rating* oleh *user* lain. Misalnya, *user* a menyukai atau me-*rating item* 1 2 dan 3, kemudian *user* b menyukai *item* 1 2 dan 4 maka sistem akan merekomedasikan *item* 3 kepada *user* b dan *item* 4 kepada *user* a. Sedangkan metode *item based* memberikan rekomendasi berdasarkan kemiripan antar *item.* Metode *item based* merupakan metode rekomendasi yang dengan adanya kesamaan antara pemberian *rating* terhadap suatu *item* dengan *item* yang pernah di-*rating user* (Wijaya & Alfian, 2018).

Menurut penelitian yang dilakukan oleh Christanti (2013) dalam melakukan perbandingan metode *user-item based* dan *item-based collaborative filtering* pada studi kasus sistem rekomendasi tempat wisata untuk wilayah Solo Dan Yogyakarta. Menyimpukan bahwasanya metode *item-based* dapat menghasilkan rekomendasi yang lebih baik dari *user-based.* Hal ini dikarenakan pada perhitungan prediksi dari metode *user-based* banyak ditemukan nilai prediksi di luar *range,* selain itu semakin bertambah *item* semakin meningkat nilai MAE dan NMAE yang dihasilkan dan proses perekomendasian akan semakin lama. Sedangkan pada metode *item-based* tidak ditemukan nilai prediksi di luar *range* dan memiliki tingkat akurasi yang lebih baik dari metode *user-based* (Christanti, 2013).

Penelitian ini mengguanakan metode *item-based collaborative filtering*, karena *item-based* membutuhkan perhitungan yang lebih sedikit tetapi mempunyai kualitas yang sama dibandingkan dengan metode *user-based*. Pada pengambilan *rating* dilakukan secara *implisit* untuk mengumpulkan data *rating* tanpa disadari oleh *user*. *Rating* diperoleh secara langsung pada saat pengguna melakukan pencarian, melakukan klik dan mengunduh suatu artikel*.* Dengan banyaknya *rating* yang terkumpul melalui aktifitas *user* akan sangat berguna pada *item-based* *collaborative filtering,* karena metode ini membutuhkan banyak *rating* agar rekomendasi yang dihasilkan semakin berkualitas.

### *Item-Based Collaborative filtering*

Metode *item-based collaborative filtering* merupakan metode rekomendasi dengan menghitung kemiripan *item* yang sudah di *rating* dengan *item* lain, kemudian dipilih sekelompok *item* yang mempunyai nilai kemiripan dengan *item* yang telah di *rating*. *Item* yang telah di *rating* oleh *user* akan menjadi patokan untuk mencari sejumlah *item* lainnya yang berkorelasi dengan *item* yang telah di *rating* *user* (Jaja et al., 2020).

Untuk membuat sistem rekomendasi menggunakan metode *item-based collaborative filtering* ada dua langkah yang harus dilakukan, sebagai berikut (Setiawan et al., 2019) :

1. Menghitung *Similarity*

Menghitung kemiripan (*Similarity)* antara satu *item* dengan *item* lain. *Cosine similarity* merupakan metode yang sering digunakan untuk menghitung kesamaan *item*, tetapi metode ini memiliki satu kekurangan. Perbedaan skala *rating* antara berbagai pengguna akan menghasilkan *simarity* yang sangat berbeda. *Adjusted cosine similarity* mengatasi kelemahan dari *cosine similarity*. Metode *adjusted cosine similarity* dapat ditunjukkan oleh Persamaan (Pamuji, 2017).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | (1) |
| Keterangan : | | | |
| *sim(i,j)*  *ΣuϵU*  *ru,i*  *ru,j*  *ru* | =  =  =  =  = | nilai kemiripan antara artikel *i* dan artikel *j*,  himpunan *user* u yang memberikan *rate* pada artikel *i* dan artikel *j*,  *rating user u* pada artikel *i*,  *rating user u* pada artikel *j*,  rata-rata *rating user u* | |

Dari perhitungan kemiripan tersebut akan mengurutkan *item*-*item* berdasakan nilai *similarity*, *item*-*item* yang mempunyai *similarity* besar akan berada pada urutan atas dan sebalikanya. Hasil dari pada persamaan *cosine similarity* adalah berkisar antara-1 s/d 1. Jika nilai *similarity* antara kedua *item* mendekati +1, maka kedua *item* dianggap dianggap semakin berkolerasi. Sebaliknya, jika nilai *similarty* mendekati -1, maka kedua *item* itu akan semakin tidak berkolerasi.

1. Menghitung prediksi *rating*

Menghitung prediksi *rating* dari *item-item* tersebut dengan membandikan *rating* yang pernah diberikan pengguna pada suatu *item* dengan kemiripan antara *item* tersebut dengan *item* lainnya. Metode yang digunakan merupakan metode *Weighted Sum.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | (2) | |
| Keterangan : | | | |
| *P(u,j)*  *iϵI*  *Ru,I*  *Si,j* | =  =  =  = | Prediksi *rating* pada artikel *j* oleh *user u*,  himpunan artikel yang mirip dengan artikel *j*,  *rating user u* pada artikel i*,*  nilai kemiripan antara artikel *i* dan artikel *j* | |

* 1. **Skala Penilaian**

Skala penilaian sering digunakan untuk mengetahui pendapat pengguna terhadap penelitian yang dilakukan. Peneliti meminta pengguna untuk memberikan penilaian terhadap sekumpulan *item*. Hal ini dilakukan untuk mengetahui seberapa besar ketertarikan responden terhadap *item* tersebut. Dalam penelitian ini menggunakan skala *Rating* (*Rating Scale*).

Menurut (Wimmer, Roger D and Dominick, 2013) *Rating scale* merupakan skala penilaian yang lebih fleksibel, skala penilaian ini tidak hanya untuk mengukur sikap tetapi dapat juga digunakan untuk mengukur persepsi responden terhadap fenomena lingkungan, seperti skala untuk mengukur status sosial, ekonomi, pengetahuan, kemampuan dan lain-lain. penilaian *rating scale* merupakan keputusan peneliti yang menentukan skala mana yang digunakan untuk penilaian yang digunakan, apakah 1 sampai 3, 1 sampai 5, 1 sampai 7, 1 sampai 10 atau 1 sampai 100.

### Flask

Flask adalah sebuah *web framework* yang ditulis dengan bahasa Python dan tergolong sebagai jenis *microframework*. Flask berfungsi sebagai kerangka kerja aplikasi dan tampilan dari suatu web. Dengan menggunakan Flask dan bahasa Python, pengembang dapat membuat sebuah web yang terstruktur dan dapat mengatur *behaviour* suatu web dengan lebih mudah (Irsyad, 2018).

## Metode Penelitian

Berdasarkan tujuan penelitian yang telah dijelaskan sebelumnya, metodologi penelitian yang digunakan peneliti dalam tugas akhir ini terdiri dari:

* 1. **Metode Pengumpulan Data**

1. Studi Pustaka

Metode ini dilakukan dengan cara mengumpulkan data yang diperoleh dari berbagai literatur, seperti buku, jurnal dan internet yang membahas tentang metode *web scraping* dan *item-based colaborative filtering*, sehingga dapat membantu proses pengerjaan tugas akhir.

Penelitian terdahulu yang mendasari pada penelitian ini, antara lain :

* + - * 1. *A Survey on Collaborative Filtering Based Recommendation System,* 2016 (G. Suganeshwari and S.P. Syed Ibrahim).

Penelitian ini memuat analisis dan survei mengenai metode pada sistem rekomendasi yang membandingkan 3 jenis metode yaitu *content based, collaborative filtering* dan *hybrid method*. Ketiga metode memiliki kelebihan dan batasan tersendiri. Pada artikel ini diusulkan untuk menggabungkan LARS dan KARS. LARS yang merupakan *item based collaborative filtering* yang mengurangi kompleksitas dengan mengurangi jumlah *user* dan *item* dengan mempertimbangkan sifat spasial mereka. KARS adalah *user based collaborative filtering* yang menangani data (*review*) yang tidak terstruktur yang diberikan oleh *user*. Hasil dari menggabungkan kedua metode telah terbukti bahwa rekomendasi yang dihasikan lebih akurat daripada teknik aslinya (Ibrahim, 2016).

* + - * 1. *Scientific Paper Recommendation: A Survey,* 2019 (Xiaomei Bai, Mengyang Wang, Ivan Lee, Zhuo Yang, Xiangjie Kong dan Feng Xia)

Penelitian ini memuat analisis teknik rekomendasi dapat dibagi menjadi empat kategori utama: *content-based filtering* (CBF), *collaborative filtering* (CF), *graph-based method* (GB) dan *hybrid method*. Dalam paper ini, membahas mengenai metode rekomendasi dan metode evaluasi sistem rekomendasi artikel ilmiah. Serta beberapa masalah dan tantangan dari sistem rekomendasi, sepeti *cold start, sparsity, scalability, privacy, serendipity* dan standar data.

1. Kuesioner

Kuesioner diberikan kepada mahasiswa yang mengerjakan tugas akhir di Fakultas Teknik Universitas Bengkulu. Hasil dari kuesioner diharapkan dapat digunakan sebagai data yang menunjang penelitian dan hal-hal yang diperlukan untuk membangun sistem ini

* 1. **Metode Pengembangan Sistem**

Metode Analisis penerapan *web scraping* dan *collaborative filtering* pada sistem rekomendasi artikel ilmiah berbasis *web* berdasarkan keterkaitan topik skripsi ini adalah metode Incremen (*Incremental development model*). *Incremental model* dipilih karena metode ini dapat meminalisir ketidak sesuaian dalam pengembangan perangkat lunak. Pada metode incement, setiap tahapan yang ada dalam metodologi terdapat masukan (*input*) dan keluaran (*output*). *Output* dari *increment* akan dijadikan masukan (*input*) increment selanjutnya­ (Shalahuddin, 2018).

Adapun tahapan tahapan dalam metode *incremental* adalah:

1. **Analisis Kebutuhan Perangkat Lunak**

Pada tahap ini dilakukan analisa terhadap kebutuhan sistem. Tujuan dari tahap ini adalah untuk menggali informasi sebanyak-banyaknya dari *user* untuk menciptakan suatu sistem informasi yang dapat melakukan tugas yang diinginkan *user* dan dapat dimanfaatkan dengan baik.

Pada tahap ini, yang dilakukan adalah :

* 1. Mengumpulkan dan menganalisis kebutuhan pengguna pada sistem rekomendasi artikel ilmiah.
  2. Membuat alur untuk dalam proses penerapan *web scraping* dan *item-based* *collaborative filtering* pada sistem rekomendasi artikel ilmiah berbasis web berdasarkan keterkaitan topik skripsi.
  3. Membuat *Interface*, yang dapat memberikan kemudahan bagi pengguna untuk berinteraksi dengan sistem informasi ini (User Friendly).
  4. Pengkodean sistem, memberikan komputasi pada sistem dengan algoritme *web scraping* dan *item-based* *collaborative filtering*.
  5. Pengujian, merupakan proses menguji kelayakan sistem dan akurasi yang dihasilkan dari algoritme *web scraping* dan *item-based* *collaborative filtering*.

Adapun kebutuhan perangkat keras dan perangkat lunak untuk pembuatan sistem ini adalah :

* + - * 1. Perangkat Lunak

Beautiful soup 4

Framewrok Flask

Browser

Microsoft Office 2019

Astah Community

Sistem Operasi Windows 10

Python 3.9.1

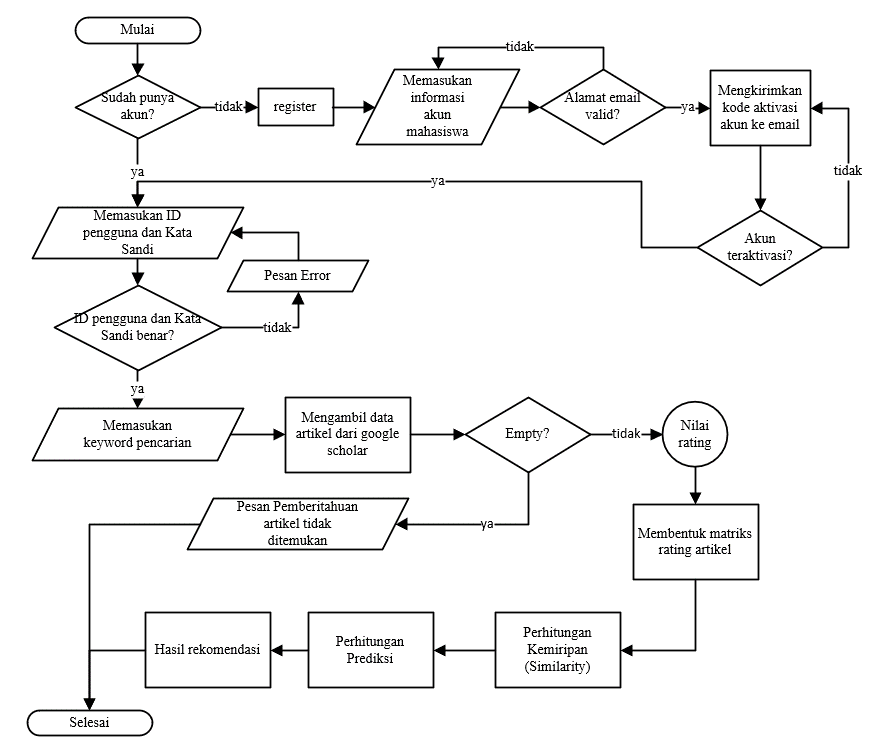
Visual Studio Code

Microsoft Visio 2019

* + - * 1. Perangkat Keras
  1. Processor AMD Ryzen™ 5 3500U
  2. RAM 8 GB

1. **Desain**

Desain perangkat lunak adalah proses yang berfokus pada desain atau perancangan dalam pembuatan perangkat lunak. Pada tahap ini, langkah yang dilakukan adalah penetapan struktur data, alur program, dan desain *interface*. Gambar 3 merupakan diagram alur penelitian sistem rekomendasi yang telah dibuat.



Gambar 3 Diagram Alur Penelitian

Gambar 3 adalah diagram yang menggambarkan proses *web scraping* dan *item-based* *collaborative filtering* pada sistem rekomendasi artikel ilmiah. Untuk mengakses sistem memerlukan akun mahasiswa, untuk menyimpan tingkah laku atau aktivitas mahasiswa terhadap sistem. Aktivitas mahasiswa digunakan sebagai *rating* yang menjadi pedoman dalam perekomendasian artikel.

Untuk membuat akun terdapat beberapa tahapan mahasiswa diminta untuk memasukan data seperti nama, NPM, email, kata sandi dan konfirmasi kata sandi. Selanjutnya sistem akan menvalidasi email yang didaftarkan, jika email valid sistem memberikan kode aktivasi pada email pengguna. Jika akun telah teraktivasi pengguna dapat melakukan *login* untuk melakukan pencarian artikel ilmiah.

Pada saat melakukan pencarian sistem mendapatkan data artikel dari halaman *google scholar*. Pencarian dapat dilakukan dengan memasukan *keyword* dapat berupa judul, penulis atau abstrak artikel. Jika *google scholar* menyediakan artikel berdasarkan *keyword*  yang dicari mahasiswa, maka sistem akan mengambil data artikel berupa judul, tahun,abstrak, sitasi dan data lainnya. Namun jika data artikel tidak ditemukan, sistem akan menampilkan pemberitahuan artikel tidak ditemukan dan proses selesai.

Bila artikel ditemukan dilanjutkan pada proses nilai *rating. Rating* yang digunakan pada sistem rekomendasi artikel ilmiah diperoleh secara *implisit*, dimana mahasiswatidak menyadari bahwa ia telah memberikan masukan terhadap sistem. Setiap pencarian yang dilakukan mahasiswaakan dihitung kemudian dijadikan sebagai *rating*, selain itu penilaian juga diambil ketika mahasiswa mengklik *link* pada suatu artikel dan mengunduh artikel.

Untuk skala pemberian *rating* dilakukan berdasarkan *rating scale* yaitu nilai 1-5, data *rating* dibagi menjadi 2 yaitu skala1-3 untuk data pencarian dan skala 4-5 untuk mengunduh artikel. Berdasarkan *rating* 1-5 dilakukan pembagian *rating* sebagai berikut:

1. Nilai *rating* 1 sampai 3 diberikan ketika mahasiswamencari dan melakukan klik *link* pada artikel*.* Jika artikel tersebut diklik sebanyak 1 kali, maka nilai *rating user* terhadap artikel tersebut adalah 1. Jika mahasiswamelakukan klik sebanyak 2 kali, nilai *rating user* terhadap artikel di-*update* menjadi 2. Begitu seterusnya sampai nilai *rating* menjadi 3. Ketika *user* melakukan klik lebih dari 3 kali, nilai *rating* tetap 3 karena nilai *rating* maksimal yang diperoleh melalui klik atau pencarian adalah 3.
2. Nilai *rating* 4 sampai 5 diberikan ketika mahasiswamelakukan mengunduh suatu artikel. Dimana ketika mahasiswamengunduh artikel 1 kali, nilai *rating user* terhadap artikel tersebut adalah 4. Ketika mahasiswamengunduh kembali artikel tersebut nilai *rating* di-*update* menjadi 5. Jika nilai *rating* mahasiswa terhadap artikelsudah mencapai 5 sebagai nilai maksimum pada penilaian, sehingga ketika user mengunduh kembali maka nilai *rating* tetap.

Tahap selanjutnya sistem akan membuat matriks *rating* artikel, sebagai contoh diberikan 4 buah *sample* artikel dan 4 mahasiswa*,* pada tabel 1 dapat dilihat nilai *rating* yang diberikan mahasiswapada setiap artikel.

Tabel 1 matriks Nilai *Rating* mahasiswaterhadap artikel

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Artikel**  **Mhs** | **A** | **B** | **C** | **D** | ***Ru*** |
| **𝑈1** | 3 | - | 1 | 4 | **2,67** |
| **𝑈2** | 2 | 5 | - | - | **3,5** |
| **𝑈3** | 2 | 2 | 1 | - | **2,5** |
| **𝑈4** | 2 | 3 | 2 | 1 | **2** |

Pada Tabel 1 Diasumsikan mahasiswayang sedang *login* adalah 𝑈1. Untuk menghitung nilai kemiripan menggunakan *adjusted cosine similarity,* terlebih dahulu ditentukan pasangan artikel yang akan dihitung nilai kemiripannya. Nilai kemiripan yang dihitung adalah antara artikel yang sudah pernah di-*rating* mahasiswadan artikel yang belum pernah di-*rating.* Untuk kasus 𝑈1 pada tabel 1, artikel yang akan dihitung nilai kemiripannya yaitu A-B, C-B, dan D-B.

Menghitung nilai *similarity* antara artikel A dan B dapat dilihat pada perhitungan dibawah ini.

Menghitung nilai *similarity* antara artikel C dan B dapat dilihat pada perhitungan dibawah ini.

Perhitungan nilai kemiripan dilakukan jika terdapat 2 atau lebih *rating* dari *user* lain pada kedua artikel tersebut. Seperti perhitungan 𝑆𝑖𝑚(b,c), *user* yang melakukan *rating* adalah 𝑈3 dan 𝑈4. Jika *rating user* lain pada artikel kurang dari 2 *user* maka perhitungan nilai kemiripan tidak dapat dilakukan seperti perhitungan nilai kemiripan antara artikel D dan B tidak dapat dihitung karena *user* yang me-*rating* kedua artikel tersebut hanyalah 𝑈4.

Hasil dari perhitungan nilai kemiripan (*similarity*) disimpan ke tabel *similarity* yang digunakan pada saat proses perhitungan prediksi, tabel kemiripan (*similarity*) terhadap *user* 𝑈1 yang dapat dilihat pada Tabel 2.

Tabel 2 Nilai Kemiripan Antar *Item*

|  |  |
| --- | --- |
| Artikel | Nilai Kemiripan |
| *Sim*(𝐴,𝐵) | -0.6761 |
| *Sim*(𝐶,B) | 0.4472 |
| *Sim*(D,B) | - |

Setelah didapatkan nilai kemiripan langkah selanjutnya adalah menghitung prediksi. Pertama sekali yang dilakukan dalam menghitung prediksi adalah membaca tabel *similarity,* kemudian data diurutkan secara *descending* dari data yang besar ke yang kecil. Pengurutan atau perangkingan ini adalah untuk mengambil sebanyak *n* data yang nilai *similarity* tertinggi. Setelah itu hitung prediksi untuk masing-masing artikel yang belum pernah di-*rating user* dengan algoritme *weight sum.*  Tahapan ini memperkirakan *rating* yang akan diberikan oleh seorang *user* pada suatu artikel yang belum pernah diberi *rating* oleh *user* tersebut.Berikut ini contoh perhitungan prediksi *rating* yang akan diberi 𝑈1 terhadap artikel B.

Dari perhitungan *weight sum* perkiraan *rating* yang akan diberikan oleh mahasiswa 𝑈1 pada suatu artikel B adalah satu.

Tahap terakhir adalah tahap melakukan perbandingan sitasi atau kutipan (pada artikel yang telah *scraping* di *google scholar*) dengan nilai-nilai prediksi yang dihasilkan. Proses terakhir ini tidak termasuk dalam perhitungan *item-based* *collaborative filtering,* namun proses tersebut dilakukan dengan tujuan untuk mendapatkan hasil rekomendasi yang lebih akurat dengan menggabungkan tingkat kemiripan artikel berdasarkan penilaian pengguna pada *google scholar* dan pengguna sistem yang akan dibangun.

1. **Pembuatan Kode Program**

Tahap ini adalah mentranslasikan desain yang telah dibuat kedalam kode-kode program. Dalam hal ini, bahasa pemrograman yang digunakan adalah Python dengan bantuan *framework* Flask.

1. **Pengujian**

Pengujian program dilakukan menggunakan dua metode, yaitu *black box* dan *white box.* Pengujian ini dilakukan untuk melihat apakah program sudah berjalan sesuai rancangan atau belum.

1. **Pendukung atau pemeliharaan**

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan oleh *user*. Tahapan ini mengantisipasi jika ada ketidaksesuaian sistem setelah dilakukan pengujian. Langkah pada tahapan ini mengulangi tahapan-tahapan sebelumnya.

## Penelitian Terkait

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Peneliti | Judul | Tahun | Metode/ Objek | Hasil | Perbedaan |
| 1 | Restu Meifitrah, Irfan Darmawan, dan Oktariani Nurul Pratiwi | Sentiment analysis of tokopedia application review to service product recommender system using neural collaborative filtering for marketplace in Indonesia | 2020 | Metode:  Web scraping dan Naive Bayes  Objek:  Review / ulasan aplikasi Tokopedia di Google Play | Diperoleh 1172 komentar dengan melakukan *web scraping* review aplikasi Tokopedia di Google Play. 601 komentar positif, 533 komentar negatif, dan 38 komentar netral. Dari hasil yang diperoleh ulasan pengguna cenderung menunjukkan komentar positif, dengan tingkat akurasi 76%.  Dapat dilihat bahwa ada perbedaan yang cukup besar antara data hasil perhitungan manual dan hasil prediksi sistem. Perbedaan terbesar pada data dengan sentimen netral. Ini terjadi karena salah tafsir kalimat, misalnya dalam kalimat netral ada kata-kata positif, menyebabkan sistem cenderung membaca data positif. | Metode yang digunakan adalah *Web scraping* dan *Naive Bayes*.  Penelitian yang akan dilakukan menggunakan metode *web scraping* dan *item-based* *collaborative filtering.* |
| 2 | Fitrianti, Betharia Sri  Fachurrozi, Muhammad  Yusliani, Novi | Sistem Rekomendasi Artikel Ilmiah Berbasis Web Menggunakan Content-based Learning dan Collaborative Filtering | 2018 | Metode:  Content-based Learning dan Collaborative Filtering  Objek:  Artikel ilmiah ber-bahasa Inggris | Pengujian performa sistem rekomendasi dilakukan dengan mengukur nilai kesalahan rata-rata absolut (Mean Absolute Error) terhadap perbandingan antara rating yang diprediksi dan rating nyata yang menggunakan 100 sampel data yang terdiri dari 60 data training dan 40 data testing, serta melibatkan 35 pengguna. Pengukuran akurasi dilakukan dengan menghitung Mean Absolute Error antara rating yang diprediksi oleh rumus dan rating nyata yang diberikan oleh pengguna. Hasil pengujian menggunakan Recall pada metode Content-based Learning adalah 91,6%. Hasil pengujian menggunakan Mean Absolute Error Hybrid Recommendation (Content-based Learning dan Collaborative Filtering) adalah 0.85. | Metode yang digunakan adalah *Content-based Learning* dan *Collaborative Filtering*. Program menggunakan bahasa pemrograman PHP.  Penelitian yang akan dilakukan menggunakan metode *web scraping* dan *item-based* *collaborative filtering.* Program menggunakan bahasa pemrograman Python |
| 3 | Muhammad Fadelillah | Sistem Rekomendasi  Artikel Jurnal Indonesia Menggunakan Metode Jaccard's Coefficient | 2017 | Metode:  Jaccard’s Coefficient  Objek:  Portal Garuda IPI  (Indonesian Publication Index) | Sistem Rekomendasi yang dibuat mampu menampilkan rekomendasi artikel lain yang ada dalam database dengan performa rata- rata precision sebesar 1 dan recall 0.87. Sedangkan pengujian pada fungsi LIKE dalam database My SQL hanya mendapatakan rata rata performa precision sebesar 0,66 dan recall 0.13, sehingga dapat disimpulkan bahwa sistem rekomendasi yang dibuat menggunakan metode jaccard’s coefficient mampu menampilkan data yang lebih valid dibandingkan dengan fungsi LIKE yang ada pada database MySQL. | Metode yang digunakan adalah *Jaccard’s Coefficient.* Program menggunakan bahasa pemrograman PHP.  Penelitian yang akan dilakukan menggunakan metode *web scraping* dan *item-based* *collaborative filtering.* Program menggunakan bahasa pemrograman Python |
| 4 | Young Jun Kim, Jeong Min Park, Sung Taek Chung dan Jeong Joan Kim | Keyword-Based Collaborative Filter Recommendation System Using Scraping | 2018 | Metode:  Collaborative Filtering dan scraping  Objek:  Rekomendasi film | Kesamaan antar film dihitung melalui kata kunci film yang diekstrak. Berdasarkan kesamaan, tetangga terdekat dari setiap film terbentuk dan ketika pengguna memilih film tertentu, itu ditujukan untuk merekomendasikan film serupa. Kata kunci yang diekstrak dalam penelitian adalah film, memiliki peran penting dalam menilai kesamaan film. | Metode yang digunakan adalah *web scraping* dan *collaborative filtering*.  Penelitian yang akan dilakukan menggunakan metode *web scraping* dan *item-based* *collaborative filtering.* Program menggunakan bahasa pemrograman Python |
| 5 | Lailatul Sabani | Sistem rekomendasi peminjaman buku perpustakaan menggunakan metode item-based collaborative filtering | 2020 | Metode:  Item-based collaborative filtering  Objek:  Peminjaman Buku Perpustakaan | Pengujian metode item-based Collaborive Filtering terhadap rekomendasi peminjaman buku tingkat error rating prediksi yang didapat sebesar 50% atau 0,5 berdasarkan 30 data buku dan 30 data mahasiswa UIN Maulana Malik Ibrahim Malang. | Metode yang digunakan adalah *Item-based collaborative filtering.* Program menggunakan bahasa pemrograman PHP.  Penelitian yang akan dilakukan menggunakan metode *web scraping* dan *item-based* *collaborative filtering.* Program menggunakan bahasa pemrograman Python |

## Waktu Pelaksanaan

Jadwal dan waktu pelaksanaan penelitian ini, yaitu :

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Bulan | | |  |  |  |
| Jan 2020 | Feb 2020 | Mar 2020 | April 2020 | Mei 2020 | Juni 2020 |
| 1 | Studi Kepustakaan |  |  |  |  |  |  |
| 2 | Penerimaan Proposal Skripsi |  |  |  |  |  |  |
| 3 | Pengumpulan dan Analisis *Data* |  |  |  |  |  |  |
| 4 | Pembuatan Sistem/Program |  |  |  |  |  |  |
| 5 | Pengujian Sistem/Program |  |  |  |  |  |  |
| 6 | Penyelesaian Laporan Akhir |  |  |  |  |  |  |

## Daftar Referensi

A. Yani, D. D., Pratiwi, H. S., & Muhardi, H. (2019). Implementasi Web Scraping untuk Pengambilan Data pada Situs Marketplace. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, *7*(4), 257. https://doi.org/10.26418/ justin.v7i4.30930

Christanti, Y. E. (2013). *Perbandingan metode user-item based dan item-based collaborative filtering pada studi kasus sistem rekomendasi tempat wisata untuk wilayah Solo Dan Yogyakarta*. UNS.

Flores, V. A., Permatasari, P. A., & Jasa, L. (2020). Penerapan Web Scraping Sebagai Media Pencarian dan Menyimpan Artikel Ilmiah Secara Otomatis Berdasarkan Keyword. *Majalah Ilmiah Teknologi Elektro*, *19*(2), 157. https://doi.org/10.24843/mite.2020.v19i02.p06

Ibrahim, G. S. and S. P. S. (2016). A Survey on Collaborative Filtering Based Recommendation System. *Smart Innovation, Systems and Technologies*, *49*, v–vii. https://doi.org/10.1007/978-3-319-30348-2

Irsyad, R. (2018). *Penggunaan Python Web Framework Flask Untuk Pemula*. https://doi.org/10.31219/osf.io/t7u5r

Jaja, Y. V. L., Susanto, B., & Sasongko, L. R. (2020). Penerapan Meode Item-Based Collaborative Filtering Umtuk Sistem Rekomendasi Data MovieLens. *Jurnal Matematika Dan Aplikasi*.

Jepriana, I. W., & Hanief, S. (2020). Analisis dan Implementasi Metode Item-based Collaborative Filtering untuk Sistem Rekomendasi Konsentrasi di STMIK Stikom Bali. *Jurnal Nasional Pendidikan Teknik Informatika : JANAPATI*, *9*(2), 171–180.

Maria Rosario B, Yovi Pratama, F. (2017). Penerapan Web Scraping Pada Website Company Profile. *Kntia*, *4*(4), 37–43.

Mitra, V., Sujaini, H., & Negara, A. (2017). Rancang Bangun Aplikasi Web Scraping untuk Korpus Paralel Indonesia - Inggris dengan Metode HTML DOM. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, *5*(1), 1–6.

Muhammad Fadelillah. (2017). *SISTEM REKOMENDASI PENCARIAN ARTIKEL JURNAL INDONESIA MENGGUNAKAN METODE JACCARD’S COEFFICIENT* [UNISSULA]. http://repository.unissula.ac.id/9839/

Pamuji, A. (2017). Sistem Rekomendasi Kredit Perumahan Rakyat Dengan Menggunakan Metode Collaborative Filtering. *Faktor Exacta*, *10*(1), 1–9.

Panduan, B., & Skripsi, P. (2018). *Buku panduan penyusunan skripsi*. FAKULTAS EKONOMI DAN BISNIS UNIVERSITAS BENGKULU.

Sabani, L. (2020). *PERPUSTAKAAN MENGGUNAKAN METODE ITEM-BASED COLLABORATIVE FILTERING*.

Setiawan, Y., Nurwanto, A., & Erlansari, A. (2019). Implementasi Item Based Collaborative Filtering Dalam Pemberian Rekomendasi Agenda Wisata Berbasis Android. *Pseudocode*, *6*(1), 13–20. https://doi.org/10.33369/ pseudocode.6.1.13-20

Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak*. Informatika Bandung.

Susrini, N. K. (2009). *Google: Mesin Pencari yang Ditakuti Raksasa Microsoft*. Bentang-B First.

Wijaya, A. E., & Alfian, D. (2018). Sistem Rekomendasi Laptop Menggunakan Collaborative Filtering Dan Content-Based Filtering. *Jurnal Computech & Bisnis*, *12*(1), 11–27.

Wimmer, Roger D and Dominick, J. R. (2013). *Mass media research*. Cengage learning.