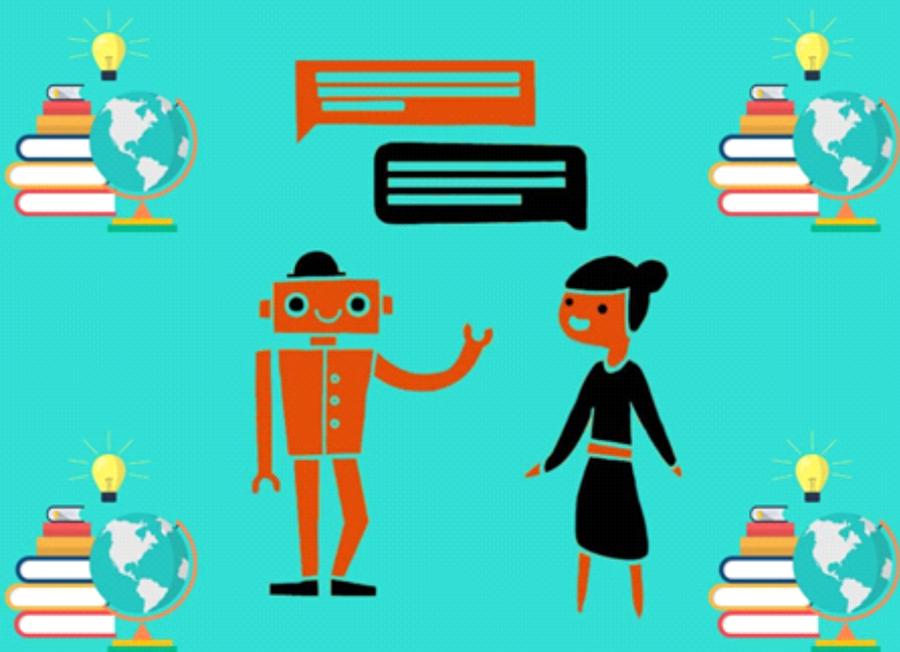


Aplikasi Chatbot (SIBOT FTI) Berbasis Artificial Intelligence



Penulis : Putri Taqwa Prasetyaningrum, S.T.,M.T |
M Hamzah Alfani Suri | Hasnidar | Haya Azzura
Rassya | Mochamad Rizki D | Miftakhul Khasan |
Mohammad Trilaksono S. | Ardiansyah Fahrul A |
Fuji Handayani

APLIKASI CHATBOT (SIBOT FTI)
BERBASIS
ARTIFICIAL INTELLIGENCE

Putri Taqwa Prasetyaningrum, S.T.,M.T | Ardiansyah
Fahrul A | Fuji Handayani | Haya Azzura Rassya |
Hasnidar | M.Hamzah Alfani Suri | Mochamad Rizki D |
Miftakhul Khasan | Mohammad Trilaksono S.



FTI UMBY

Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta

Aplikasi Chatbot (SIBOT FTI) Berbasis Artificial Intelligence

Hak Cipta dilindungi Undang-Undang
All Rights Reserved
85 hal, 14,8 cm x 21 cm
ISBN : 978-623-95744-6-8

Penulis :

Putri Taqwa Prasetyaningrum, S.T.,M.T. | Ardiansyah Fahrul A;
Fuji Handayani | Haya Azzura Rasya | Hasnidar | M.Hamzah
Alfan Suri | Mochamad Rizki D. | Miftakhul Khasan |
Mohammad Trilaksono S.

Editor :

Putri Taqwa Prasetyaningrum, S.T.,M.T.

Tata Isi :

Haya Azzura Rasya & Hasnidar

Tata Sampul :

M. Hamzah Alfan Suri

Cetakan Pertama, Juni 2021

Diterbitkan Oleh:

Fakultas Teknologi Informasi
Universitas Mercu Buana Yogyakarta
JL. Jembatan Merah, No. 84.C. Gejayan, Yogyakarta
Telp: (0274) 550703
Email: tu.fti@mercubuana-yogya.ac.id

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh

Segala puji bagi Allah SWT yang telah memberikan kami kemudahan sehingga dapat menyelesaikan buku dengan judul “Aplikasi Chatbot (SIBOT FTI) Berbasis Artificial Intelligence” yang merupakan tugas mata kuliah, Kuliah Industri. Berkat tugas kuliah menulis tersebut kami mendapat tambahan wawasan dan ilmu pengetahuan. Tidak lupa kami menyampaikan rasa terimakasih kepada Ibu Putri Taqwa Prasetyaningrum, S.T., M.T.. yang telah memberikan arahan kepada kami sehingga dapat menyelesaikan buku ini dengan tepat waktu. Kami ucapkan pula terima kasih kepada teman – teman yang sudah ikut serta berpartisipasi meluangkan waktunya untuk sekedar membantu kami dalam penyelesaian ini. Dan ucapan terima kasih kami untuk semua yang tak bisa kami sebutkan satu per satu namanya

Penulis menyadari bahwa dalam penulisan buku ini banyak kekurangan, oleh sebab itu kami mengharapkan kritik dan saran yang membangun. Dan semoga dengan selesainya buku ini dapat bermanfaat bagi kami khususnya dan umumnya bagi pembaca.

UNTUK PARA PEMBACA

Di dalam penulisan buku ini, kami berusaha memanfaatkan kemudahan yang ada secara optimal, kesulitan dan keterbatasan yang menghambat proses penyusunan buku ini dapat diatasi dengan bantuan dari berbagai pihak baik secara langsung maupun tidak langsung.

“Tiada gading yang tak retak”,

Kami sadar buku ini sangat jauh dari kesempurnaan dan harapan. Oleh sebab mungkin buku ini kurang memenuhi kebutuhan pengguna atau pembaca serta mengalami kesulitan setelah membaca buku ini. Mohon maaf jika ada salah penulisan dan kata-kata.

DAFTAR ISI

KATA PENGANTAR.....	i
UNTUK PARA PEMBACA	ii
DAFTAR ISI	iii
BAB I.....	1
PENDAHULUAN.....	1
1.1 Sejarah <i>Chatbot</i>	1
1.2 Sejarah <i>Rapid Application Development (RAD)</i> .	2
1.3 Sejarah Fakultas Teknologi Informasi	4
BAB II	9
DASAR TEORI.....	9
2.1. Konsep Dasar Rapid Application Development ..	9
2.2. Pengertian <i>Rapid Application Development</i>	10
2.3. Tahapan <i>Rapid Application Development (RAD)</i>	12
2.4. Unsur-unsur <i>Rapid Application Development (RAD)</i>	14
2.5. Waktu yang tepat untuk menggunakan <i>Rapid</i> <i>Application Development (RAD)</i>	19
BAB III.....	22
PENGENALAN SISTEM	22
3.1 HTML5	22

3.1.1	Sejarah HTML5.....	24
3.1.2	Keunggulan HTML5	26
3.1.3	Kekurangan HTML5	28
3.2	PHP	28
3.2.1	Sejarah PHP	29
3.2.2	Keunggulan PHP	31
3.2.3	Kekurangan PHP	35
3.3	CSS	36
3.3.1	Sejarah CSS	38
3.3.2	Keunggulan CSS.....	40
3.3.3	Kekurangan CSS.....	41
BAB IV	43
<i>METODE RAPID APPLICATION DEVELOPMENT</i>		
(RAD)	43
4.1	Pengertian <i>Rapid Application Development</i> (RAD)	43
4.2	Perencanaan Persyaratan.....	45
4.3	Pemodelan.....	46
4.4	Implementasi.....	46
4.4.1	Implementasi Interface	47
4.5	Implementasi Aplikasi	52
4.5.1	Implementasi Interface	52
BAB V	56

DEMO SIBOT FTI	56
5.1 <i>Source Code</i>	56
5.2 Demo SIBOT FTI	61
5.2.1 Langkah-Langkah Uji Coba	61
5.2.2 Hasil Demo SIBOT FTI	64
5.2.3 Pembahasan	65
BAB VI.....	66
KESIMPULAN.....	66
DAFTAR PUSTAKA.....	68
BIODATA PENULIS.....	71

BAB I

PENDAHULUAN

1.1 Sejarah *Chatbot*

Perkembangan *chatbot* dimulai pada tahun 1950 pada saat Alan Turing merilis artikel yang bernama “*Computing Machinery and Intelligence*” dan juga menciptakan *Turing test* sebagai salah satu standar kecerdasan buatan. Sehingga dengan melalui tes itu, Turing akhirnya melakukan percobaan pada mesin untuk melihat apakah mesin itu bisa memiliki kemampuan seperti layaknya manusia. Sejarah dari *chatbot* bermula dari pengembangan aplikasi chat ELIZA pada tahun 1966 dan PARRY pada tahun 1972. Pada waktu itu, pengembangan ELIZA dan PARRY digunakan untuk menstimulasi percakapan tertulis. Salah satu penelitian penting ini di bidang kecerdasan buatan (AI) adalah pemrosesan bahasa alami (*Natural Language*

Processing). Biasanya, bidang AI lemah memberdayakan perangkat lunak khusus atau bahasa pemrograman yang dibuat secara spesifik dengan fungsi yang lebih sempit.

Cara kerjanya masih murni berdasarkan teknik pencocokan pola tanpa kemampuan penalaran teknik yang sama yang digunakan ELIZA pada tahun 1966. Seiring berjalannya waktu, *chatbot* juga baru dikombinasikan dengan karakteristik fungsional karena mudah digunakan.

1.2 Sejarah *Rapid Application Development* (RAD)

Awal mula istilah *Rapid Application Development* atau yang biasa disebut RAD yaitu awalnya hanya digunakan untuk menggambarkan proses pengembangan pada perangkat lunak pertama kali. RAD dibuat untuk mengatasi metodologi sebelumnya yang dikembangkan pada tahun 1970.

Pada tahun 1980, James Martiin mengembangkan metodologi RAD di IBM dan diresmikan pada tahun 1991 melalui bukunya yang berjudul *Rapid Application Development*. RAD adalah salah satu model SDLC yang sangat populer juga dikenal sebagai model RAD yang lebih mengutamakan kecepatan dan fleksibilitas daripada perencanaan awal yang menyeluruh atau metode yang berfokus pada pengembangan aplikasi secara cepat, melalui pengulangan dan *feedback* berulang-ulang.

Adapun beberapa kelebihan dari *Rapid Applicarion Development* (RAD) jika digunakan yaitu sebagai berikut:

- a. Waktu pengerjaannya dapat diperkirakan juga perubahan kebutuhan dapat diakomodisi.
- b. Waktu iterasi dapat dipersingkat dengan menggunakan RAD *Tools*.
- c. Lebih efektif dari pendekatan

waterfall dalam menghasilkan sistem yang memenuhi kebutuhan langsung dari pelanggan.

- d. Siklus waktu dalam pengembangan langsung dari pelanggan.
- e. Pengembangnya banyak karena terdiri dari beberapa tim.
- f. Mudah dalam menentukan dasar sistem.

1.3 Sejarah Fakultas Teknologi Informasi

Fakultas Teknologi Informasi Universitas Mercu Buana merupakan salah satu dari beberapa fakultas yang ada di Universitas mercu buana yogyakarta, berdiri sejak tahun 2008. Sekarang ini Fakultas Teknologi Informasi Universitas mercu Buana Yogyakarta di pimpin oleh Bapak Anief Fauzan R., S.Kom., M.Eng. Pada kepemimpinan beliau Fakultas ini mulai berkembang dan menunjukkan berbagai

perubahan ke depan sehingga mulai dikenal oleh masyarakat luas. Fakultas ini beralamat di Kampus II Universitas Mercu Buana Yogyakarta, tepatnya di Jl. Jembatan Merah 84-C, Gejayan.

Perkembangan teknologi menuntut perguruan tinggi di zaman sekarang untuk selalu meningkatkan sumber daya manusianya. Tidak terkecuali Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta ini. Fakultas Teknologi Universitas Mercu Buana Yogyakarta akan selalu meningkatkan proses belajar mengajar agar tercapai lulusan yang beretika dan memiliki pengetahuan serta kompetensi teknologi informasi di bidang *artificial intelligence* dan multimedia.

Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta memiliki 2 Program Studi yaitu Informatika dan Sistem Informasi. Pada Program Studi Informatika bertujuan menyiapkan *programmer* muda yang kompeten

dan professional meliputi beberapa bidang, yaitu:

- a) *Analysist Programmer*
- b) *Database Administrator*
- c) *Web site Designer*
- d) *Network Engineer*
- e) *Intelligence System Developer*

Pada Program Studi Sistem Informasi dibekali ilmu pengetahuan dan keterampilan agar tercapai lulusan sebagai :

- a) *Business Analyst*
- b) *Project Manager Enterprise Resource Planning (ERP)*
- c) *Manager Data Warehouse*

Dengan sarana prasarana praktik yang memadai serta bimbingan dosen yang professional mahasiswa Fakultas Teknologi Universitas Mercu Buana Yogyakarta yakin mampu menjawab tantangan dunia kerja dan industri.

Untuk menciptakan kegiatan belajar mengajar yang nyaman Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta menyediakan fasilitas yang memadai, diantaranya Gedung Kampus II yang representatif dengan lokasi yang strategis serta lingkungan yang kondusif untuk kegiatan belajar mengajar, ruang kelas yang nyaman, laboratorium komputer, fasilitas olahraga, hotspot area atau internet, mushola dan pepustakaan Program Studi.

Selain aktifitas belajar mengajar Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta juga mendukung mahasiswanya untuk berorganisasi dalam Fakultas. Diantaranya Badan Eksekutif Mahasiswa (BEM) Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta, Himpunan Mahasiswa Program Studi Informatika (HMPSTI) dan

Himpunan Mahasiswa Program Studi Sistem Informasi (HMPSSI). Hal ini tentu saja agar mahasiswa memiliki kemampuan untuk bersosialisasi dengan mahasiswa lainnya serta pihak terkait agar tercipta pribadi yang memiliki mental yang baik dan Tangguh.

BAB II

DASAR TEORI

2.1. Konsep Dasar Rapid Application Development

Rapid Application Development (RAD) adalah suatu sistem yang menggunakan berbagai macam gabungan teknik yang terstruktur. Strategi siklus hidup yang ditujukan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang dicapai melalui siklus tradisional. Model *Rapid Application Development* merupakan sebuah adaptasi berkecepatan tinggi dari model waterfall yang dimana perkembangannya pesat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen dan jika tiap-tiap kebutuhan dan batasan ruang lingkup proyek telah diketahui

dengan baik maka proses RAD memungkinkan tim pengembang untuk menciptakan sebuah sistem yang berfungsi penuh dalam jangka waktu yang sangat singkat.

2.2. Pengertian *Rapid Application Development*

RAD adalah kepanjangan dari *Rapid Application Development*. RAD merupakan metode pengembangan sistem yang memiliki keunggulan karena tahapan yang singkat dan cepat seperti tahapan *requirements planning* untuk mengidentifikasi tujuan dari aplikasi atau sistem, tahapan RAD *design workshop* (pemodelan) untuk membangun tampilan visual desain dan alur kerja *user* dan tahapan implementasi untuk pembangunan sistem dan pengujian. (Daffa, 2020)

RAD adalah sebuah strategi pengembangan sistem yang menekankan kecepatan dalam pengembangan melalui keterlibatan pengguna dalam pembangunan

secara cepat, *iterative, incremental* dari suatu serangkaian *prototype* dari suatu sistem yang dapat berkembang menjadi suatu sistem akhir atau versi tertentu (Whitten & Bentley,2007:98).

Adapun ide dasar dari *Rapid Application Development* adalah sebagai berikut :

- a. Untuk mengurangi waktu yang dibutuhkan pengguna untuk melihat proses kerja sistem yang dikembangkan.
- b. Untuk mengorganisasikan pengembangan sistem agar lebih fokus dan lebih melibatkan *system owner, users, analyst, designer* dan *builder*.
- c. Untuk mempercepat kegiatan analisa dan perancangan kebutuhan sistem melalui pendekatan pembangunan

iteratif.

- d. Untuk lebih mengefektifkan pengguna dalam keterlibatan kegiatan menganalisa, mendesain maupun membangun program.

2.3. Tahapan *Rapid Application Development* (RAD)

Adapun tahapan-tahapan dalam RAD yang terstruktur dan saling bergantung pada setiap tahap yaitu sebagai berikut :

- 1) *Requirements Planning* (Perencanaan Persyaratan)

Pada tahap ini dilakukan pengumpulan informasi terkait beberapa hal berikut:

- a. Pengumpulan informasi tentang bagaimana cara menjalankan aplikasi *chatbot*.

- b. Pengumpulan informasi tentang bagaimana cara membangun sebuah *chatbot*

Pada tahap ini, *user* dan analis melakukan pertemuan untuk melakukan identifikasi tujuan dari aplikasi atau sistem dan melakukan kebutuhan informasi untuk mencapai tujuan. Pada tahap ini hal terpenting adalah adanya keterlibatan dari kedua belah pihak, bukan hanya sekedar persetujuan akan proposal yang sudah dibuat.

2) RAD *Design Workshop* (Pemodelan)

Pada tahap ini, melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan analis. Untuk tahap ini maka keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa

langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain.

3) Implementasi

Setelah desain dibuat dan disetujui baik oleh *user* dan analis, maka pada tahap ini programmer mengembangkan desain menjadi suatu program. Setelah program selesai baik itu sebagian maupun secara keseluruhan, maka dilakukan proses pengujian terhadap program tersebut apakah terdapat kesalahan atau tidak sebelum diaplikasikan pada suatu organisasi.

2.4. Unsur-unsur *Rapid Application Development (RAD)*

Rapid Application Development (RAD) memiliki beberapa unsur-unsur yang membuat sebuah metodologi yang unik yaitu *prototyping*,

time boxing, Iterative development, team member dan RAD Tools.

a. Prototyping

Aspek kunci dari *Rapid Application development* (RAD) adalah membangun *prototype* dengan tujuan untuk membangkitkan kembali desain kebutuhan pengguna, yang bertujuan untuk membangun sebuah fitur ringan yang hasilnya akhirnya dalam jumlah yang pendek dengan waktu yang cukup memungkinkan. Fungsi dari prototype sebagai bukti konsep untuk klien, tetapi fungsinya lebih penting sebagai titik bicara dan alat untuk kebutuhan pemurnian. Perkembangan *prototype* cepat dicapai dengan menggunakan *Computer Aided Engineering CASE Tools software* yang berfokus untuk menangkap persyaratan, mengkonversi ke model data, mengubah

data ke *database*, dan menghasilkan kode semua dalam satu alat.

b. *Time boxing*

Time boxing adalah suatu proses menunda fitur untuk versi aplikasi di masa mendatang yang untuk melengkapi versi saat ini sebagai bentuk ketepatan waktu. Ketepatan waktu merupakan aspek yang sangat penting bagi RAD, dikarenakan tanpa ketepatan waktu, ruang lingkup dapat mengancam untuk memperpanjang iterasi pada pembangunan sehingga akan membatasi umpan balik dari klien nantinya, meminimalkan manfaat dari pembangunan berulang, dan berpotensi mengembalikan proses kembalinya pendekatan metodologi air terjun.

c. *Iterative development*

Iterative development menciptakan versi yang lebih fungsional dari sebuah sistem dalam siklus pembangunan jangka pendek, yang setiap versi ditinjau dengan klien

untuk menghasilkan persyaratan untuk membuat ke versi berikutnya. Proses ini dilakukan secara berulang sampai semua fungsionalitas berkembang. Panjang ideal pada iterasi ini adalah antara satu hari dan tiga minggu. Setiap siklus pengembangan memberikan pengguna kesempatan untuk memberikan *feedback*, memperbaiki persyaratan, mamupun melihat kemajuan dalam pertemuan.

d. *Team member*

Metode *Rapid Application Application* (RAD) ini merekomendasikan tim kecil yang terdiri dari tim-tim yang sudah berpengalaman, serba bisa, dan motivasi yang mampu melakukan peran ganda. Sebagai klien memainkan peran penting dalam proses pembangunan, sumber daya klien khusus harus tersedia selama awal *Joint Application development* (RAD) sesi serta *Focus Group sessions* yang dilakukan pada akhir siklus

pengembangan. Pengembangan tim juga dikenal sebagai *Skilled Workes With Advanced Tools* (SWAT) yang idealnya harus memiliki pengalaman di *Rapid Application Development* (RAD) dan harus memiliki pengalaman dengan *Computer Aided Software Engineering*. Pendekatan manajemen aktif yang terlibat sangat penting untuk mengurangi resiko siklus pemngembangan diperpanjang, kesalahpahaman klien, dan melebihi tenggat waktu. Di atas manajemen semua harus kuat dan konsisten dalam keinginan mereka untuk menggunakan metodologi pada RAD. Selain menegakkan waktu yang cukup ketat, manajemen harus fokus pada pemilihan anggota tim, otivasi tim maupun pada kliring hambatan birokrasi atau politik.

e. *RAD Tools*

Salah satu tujuan utama dari metodologi RAD yaitu yang dikembangkan oleh

James Martiin pada tahun 1980an untuk memanfaatkan teknologi terbaru yang tersedia untuk mempercepat pembangunan. Teknologi tahun 1980-an sudah kuno, tetapi RAD fokus terhadap alat terbaru yang sama pentingnya seperti ketika metodologi awalnya diciptakan.

2.5. Waktu yang tepat untuk menggunakan *Rapid Application Development (RAD)*

Berdasarkan karakteristik pada *Rapid Application Development* ada beberapa kondisi yang membuat RAD akan berjalan lebih maksimal, yaitu seperti :

- 1) Memiliki Modal besar

Rapid Application Development (RAD) merupakan model yang terbilang efisien, oleh karena itu kebanyakan orang berfikir bahwa RAD ini merupakan model paling hemat budget, namun siapa sangka meskipun RAD terbilang efisien dan cepat,

kita tetap memerlukan modal yang besar agar RAD ini bisa dilakukan dengan efektif dengan membayar tim teknis ahli yang bayarannya cukup tinggi.

2) Tim Solid

Tahapan pada *Rapid Application Development* (RAD) sangat menekankan begitu pentingnya *feedback* dan testing ulang *prototypes* maka dari itu kita membutuhkan tim yang benar benar solid ketika kita memilih model RAD untuk digunakan. Melainkan tim yang kita butuhkan adalah tim internal *developer* dan juga *user* yang akan melakukan *testing* , anggota tim lain, maupun terhadap klien. Semuanya perlu memiliki cukup komitmen untuk terus mengikuti proses pengembangan pada aplikasi yang dengan begitu mereka bisa memberikan *feedback* yang berkualitas dan patuh dengan *timeline* yang sudah disepakati bersama.

3) Mempunyai sedikit waktu

Jika tidak mempunyai banyak waktu namun ingin menyelesaikan pekerjaan dengan cepat, pilihan menggunakan metode *Rapid Application Development* (RAD) adalah pilihan terbaik. Dikarenakan metode ini tidak menghabiskan banyak waktu di tahap perencanaan. Sebaliknya, proses pengembangan aplikasi bisa berjalan fleksibel dan sesuai kebutuhan. Kemudian proses *testing* dan *feedback* yang intens membuat aplikasi yang dihasilkan lebih stabil.

BAB III

PENGENALAN SISTEM

3.1 HTML5

HTML5 merupakan versi terbaru dari HTML. *Hypertext Markup Language* (HTML) adalah Bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman *website*. Halaman ini kemudian diakses untuk menampilkan berbagai informasi di dalam sebuah browser. HTML adalah Bahasa pemrograman yang paling umum digunakan yang dirancang untuk memproses, menentukan, dan menyajikan teks dengan menyematkan tag dan anotasi teks dalam file untuk membuat manipulasi teks yang mudah bagi komputer.

HTML5 adalah revisi dari *Hypertext Markup Language* (HTML) Bahasa pemrograman standar untuk menggambarkan konten dan tampilan halaman *web*. HTML5 dikembangkan

untuk menyelesaikan masalah ompatibilitas yang memengaruhi standar saat ini, HTML4 salah satu perbedaan terbesar antara HTML5 dan versi standar sebelumnya adalah bahwa versi HTML yang lebih lama memerlukan plugins dan API eksklusif. HTML5 menyediakan satu antarmuka umum untuk membuat elemen yang dimuat dengan lebih mudah. Contohnya seperti tidak perlu menginstal *plugin Flash* di HTML5 karena akan berjalan dengan sendirinya.

Salah satu tujuan desain untuk HTML5 adalah untuk mendukung multimedia di perangkat seluler. Fitur sintaksus baru diperkenalkan untuk mendukung inni seperti tag video, audio, dan kanvas. HTML5 diadopsi oleh kelompok kerja baru *World Wide Consortium (W3C)* pada tahun 2007. Grup ini menerbitkan konsep publik pertama pada HTML5 pada januari 2008 sampai skrg HTML5 berada dalam status “*Call for Review*” dan W3C waktu itu menargetkan bahwa versi ini akan mencapai finalnya pada akhir 2014.

HTML5 memiliki sejumlah fitur tambahan bagi pendahulunya seperti dukungan penyimpanan media *offline*, elemen konten yang lebih spesifik seperti *footer*, *header*, navigasi dll. Doctype inline yang lebih sederhana, dukungan *embedding audio*, dan video.

3.1.1 Sejarah HTML5

HTML versi ini atau lebih dikenal dengan HTML5 lahir pada tahun 2008 yang menjadi standar baru untuk HTML, XHTML, & DOM HTML. HTML5 ini merupakan proyek kerjasama antara W3C dengan WHATWG (*Web Hypertext Application Technology working group*). HTML5 menambahkan beberapa perintah baru yang membantu para programmer *web* untuk mendesain *website* lebih menarik dan interaktif.

Sejarah HTML diciptakan oleh *Timothy John Barners Lee* pada tahun 1980

pada saat ini dia bekerja di Lembaga CERN (*Organisation Europeene pour la Recherche Nucleaire*), CERN adalah sebuah sebuah organisasi riset fisika yang terbesar di dunia. Pada saat itu *Tim Barners Lee* sedang menjaankan sebuah peoyek yang berbasiskan konsep *hypertext* untuk memfasilitasi pembagian dan pembaharuan informasi diantara para peneliti, tujuan awalnya adalah agar setiap peneliti dapat melakukan share atau membagikan ilmu terhdap apa yang telah ditemukan selama proses penelitian, berkat bantuan dari Robert cailliau akhirnya *prototype* bernama Enquire tercipta dari sinilah sejarah HTML dan perkembangan HTML5 dimulai.

Setelah *Lee* tidak bekerja lagi di CERN, *Lee* memiliki pemikiran bagaimana jika proses *share* yang dia lakukan di CERN dapat dirasakan oleh peneliti ditempat atau belahan lain, lalu dia bekerja dengan *John*

Poole's Image Computer systems Ltd pada tahun 1984 sebagai seorang rekan peneliti. Tujuannya bergabung dengan John adalah untuk mengembangkan ide yang dia gunakan pada *enquire* untuk menciptakan *World Wide Web*, yang di mana dia mendesain dan membangun browser yang pertama bernama *World Wide Web* dan dikembangkan dalam *NEXTSTEP* dan server *web* pertama yang bernama *httpd*. Setelah sukses dengan penelitiannya pada tahun 1989, *Lee* mendirikan sebuah lembaga yang bernama *World Wide Consortium (W3C)* forum untuk terus mengembangkan *web*, dari sinilah sejarah HTML dan perkembangan HTML5 terlahir dari Generasi HTML dimulai.

3.1.2 Keunggulan HTML5

Berikut keunggulan dari HTML5 yaitu sebagai berikut :

1. Dukungan untuk pembuatan *website responsive*.
2. Menjalankan *JavaScript* di *web browser*.
3. Penyimpanan informasi secara lokal.
4. Dukungan untuk grafik *vector*.
5. *Support* untuk konten video dan audio.
6. *Syntax* yang lebih sederhana.
7. Kemudahan untuk membuat aplikasi *web*.
8. Penanganan *error* yang lebih baik.
9. Fokus otomatis pada kolom *form*.
10. Kompatibel dengan lebih banyak *browser*.
11. Lebih *markUp* untuk menggantikan *scripting*.
12. Mengurangi kebutuhan untuk plugin eksternal.

3.1.3 Kekurangan HTML5

Selain keunggulan tentunya semua bahasa pemrograman mempunyai kekurangan dan berikut kekurangannya yaitu sebagai berikut :

1. Fitur keamanan yang ditawarkan HTML5 masih terbatas.
2. Kekurangan utama yang dimiliki oleh HTML5 ini adalah versi ini hanya *support* untuk *browser modern*.
3. Karena bahasa HTML5 ini masih dalam perkembangan, jadi beberapa elemen yang ada bisa saja berubah..

3.2 PHP

Hypertext Preprocessor (PHP) yaitu Bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan situs *web* dan bisa digunakan bersamaan dengan HTML. PHP adalah Bahasa pemrograman yang

berjalan pada *server side scripting* dan bersifat *Open Source*. Untuk penggunaan dari Bahasa ini sering digunakan untuk pembuatan aplikasi berbasis *website* yang berjalan secara dinamis, sehingga dapat terintegrasi dengan basis data (*database*).

PHP banyak digunakan untuk pembuatan *website* untuk kebutuhan *e-commerce*, sistem informasi, maupun *landing page*. Karena PHP tergolong dalam bahasa pemrograman berbasis *server-side*, maka *script* yang digunakan akan diproses oleh server. Jenis server yang sering dipakai adalah *Apache*, *Nginx*, dan *LiteSpeed*.

3.2.1 Sejarah PHP

Sejarah pertama kali PHP diciptakan oleh *Rasmus Lerdorf* yang merupakan seorang pemrograman bahasa C yang bertujuan untuk mencatat jumlah pengunjung yang masuk pada halaman beranda. Pada tahun 1995, PHP/FI (*Form Interpreter*) dirilis

dengan kemampuan untuk membangun tampilan aplikasi *web*, melakukan pemrosesan *form*, serta mendukung data pada *MySQL*.

Karena semakin banyak dan antusias dari pengguna internet, akhirnya pada pertengahan tahun 1997, *Rasmus Lerdorf* menyerahkan pengembangannya kepada tim *developer* yang terdiri dari *Zeev Suraski* dan *Andi Gutmans*. Sehingga, terciptalah versi PHP 3.0 yang memiliki dukungan luas terhadap *database Oracle* dan *MySQL*.

Selanjutnya, seiring berkembangnya teknologi informasi dan penggunaan *website* yang telah mendukung HTML5, maka PHP mengalami beberapa peningkatan versi. Dimulai dari PHP versi 4.0 hingga 8.0 yang memiliki fitur dan fungsionalitas yang lebih optimal. Dengan fitur yang lebih advanced, maka Bahasa pemrograman ini mampu untuk

menangani *website* dengan skala dan trafik yang besar dan kompleks.

3.2.2 Keunggulan PHP

Berikut keunggulan dari PHP yaitu sebagai berikut :

1. Pengembangan Cepat, membuat aplikasi menggunakan pemrograman PHP jauh lebih cepat daripada mengembangkan aplikasi *web* menggunakan bahasa pemrograman yang lain. Karena ada banyak *Tools*, *boiler* yang tersedia secara terbuka (*Open Source*) untuk pemrograman PHP ini. Hal ini mempercepat proses dalam pembuatan sebuah *project*.
2. *Maintenance* mudah, aplikasi *web* yang dibuat dengan menggunakan PHP dapat dengan mudah *upgrade* versi PHP tanpa harus melakukan kompilasi ulang *Source code*.

Berbeda dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika melakukan upgrade versi dari bahasa pemrograman. PHP juga dapat berjalan pada berbagai macam *web server* seperti *Apache*, *Nginx*, dan *IIS*.

3. Perkembangan Pesat, karena PHP yang bersifat *Open Source*, ada banyak bermunculan *project-project* besar *Open Source* yang menggunakan PHP seperti *Prestashop*, *WordPress*, *Drupal*, dan lain-lain. Hal ini menjadi keunggulan yang sangat besar bagi yang orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas.

4. *Open Source*, PHP merupakan sebuah project pen source dengan license yang dikeluarkan oleh PHP *group* yaitu PHP *license* V3.01. Inti dari *license* ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan *royalty* apapun ke PHP *Group* namun tetap wajib mencantumkan *license* atau PHP yang dimiliki PHP *Group*. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buaatannya maka perjual berlian program yang menggunakan PHP diperbolehkan tanpa harus membayar *license* apapun.
5. Ringkas, Menggunakan PHP terasa sangat ringkas dan praktis berbeda dengan bahasa pemrograman lain yang membutuhkan proses kompilasi untuk dapat melihat *website* yang

telah diselesaikan pembuatan kodenya. Bahkan, bahasa pemrograman PHP dapat digunakan di dalam dokumen HTML,

6. Mudah dipelajari, PHP mudah diinstal dan dikonfigurasi membuatnya menjadi bahasa pemrograman tingkat *entry level* yang mudah dipelajari bagi seseorang yang baru memulai belajar pengembangan *web*.
7. Memiliki Komunitas yang besar, ada banyak *website* yang menggunakan PHP sebagai bahasa pemrograman untuk membuat aplikasi berbasis *web* atau *websitenya*. *Facebook*, *Yahoo*, *Wikipedia*, *wordpress* adalah contoh *website* terkenal yang menggunakan PHP. Forum untuk membahas dan juga saling bertukar pikiran dalam pemrograman PHP juga telah banyak muncul di berbagai situs.

3.2.3 Kekurangan PHP

Selain keunggulan tentunya semua bahasa pemrograman mempunyai kekurangan dan berikut kekurangannya yaitu sebagai berikut :

1. Kode PHP bisa dibaca semua orang dan kompilasi hanya bisa dilakukan dengan yang maham dari Zend (\$2000).
2. Tidak ideal jika untuk pengembangan skala besar.
3. PHP mempunyai kelemahan *security* tertentu yang mana jika *programmer* tidak jeli dalam melakukan pemrograman dan kurang memperhatikan isu dan konfigurasi PHP.
4. Tidak mempunyai sistem pemrograman berorientasi objek yang sesungguhnya.

5. Tidak dapat memisahkan antara tampilan dengan *logic* dengan baik meskipun penggunaan *template* bisa memperbaikinya.
6. Tidak adanya tipe data pada PHP. Hal inilah yang membuat munculnya *bug* *bug* yang tidak diinginkan berkaitan dengan tidak adanya standar tipe data pada PHP.
7. PHP tidak mengenal *package*.

3.3 CSS

CSS adalah kepanjangan dari *Cascading Style Sheet* yang berfungsi untuk mengatur tampilan elemen yang tertulis dalam bahasa *markup*. Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang CSS bisa diaplikasikan untuk segala dokumen XML, termasuk SVG dan XUL bahkan Android.

CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen yang meliputi *layout*, warna dan font. Pemisahan ini dapat meningkatkan daya akses konten pada *web*, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi dari sebuah karakteristik dari sebuah tampilan, memungkinkan untuk membagi halaman untuk sebuah formatting dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten. CSS juga memungkinkan sebuah halaman untuk ditampilkan dalam berbagai *style* dengan menggunakan metode pembawaan yang berbeda-beda seperti *on-screen*, *in-print*, *by voice*, dan lain-lain. Sementara itu, pemilik konten *web* bisa menentukan link yang menghubungkan konten dengan file CSS.

Tujuan utama CSS diciptakan untuk membedakan konten dari dokumen dan dari tampilan dokumen, dengan itu, pembuatan ataupun pemrograman ulang *web* akan lebih mudah dilakukan. Hal yang termasuk dalam

desain *web* diantaranya warna, ukuran dan formatiing. Dengan adanya CSS, konten dan desain *web* akan mudah dibedakan jadi memungkinkan untuk melakukan pengulangan pada tampilan-tampilan tertentu dalam suatu *web*, sehingga akan memudahkan dalam membuat halaman *web* yang banyak, yang pada akhirnya dapat memangkas waktu pembuatan *web*. Fungsi utama CSS adalah merancang, merubah, mendesain, membentuk halaman *website* da nisi dari halaman *website* seperti *tag-tag* HTML.

3.3.1 Sejarah CSS

CSS pertama kali bermula sejak munculnya SGML pada tahun 1970an. Sejak kemunculannya tersebut, CSS mengalami perkembangan yang sangat pesat. Format dasar CSS yang banyak kita gunakan sekarang ini merupakan ide dari seorang *programmer* bernama *Hakon Wium Lie* yang tertuang dalam proposalnya mengenai *Cascading HTML Style sheet* (CHSS) pada

bulan oktober 1994 dalam konferensi W3C di Chicago, Illinois. kemudian beliau bersama-sama dengan seorang temannya yang bernama Bert Bos mengembangkan suatu standard CSS.

Pada akhir tahun 1996, CSS telah resmi dipublikasikan dan menyusul kemudian CSS level 1 pada bulan desember. Pengerjaan proyek ini juga didukung oleh seorang *programmer* bernama *Thomas Reardon* dari perusahaan *software* ternama, *Microsoft*. CSS digunakan oleh penulis maupun pembaca halaman *web* untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen yang ditulis dengan HTML atau Bahasa markup lainnya dengan presentasi dokumen yang ditulis dengan CSS.

Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan control terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada struktur isi. CSS memungkinkan halaman yang sama untuk ditampilkan dengan cara yang berbeda untuk metode presentasi yang berbeda. Sekarang penggunaan CSS telah semakin meluas dan terus dikembangkan. Hal ini juga akan mempermudah seorang *web* designer dalam mengembangkan suatu halaman *web*.

3.3.2 Keunggulan CSS

Berikut keunggulan dari CSS yaitu sebagai berikut :

1. Dapat digunakan pada semua *browser*.
2. Beban *bandwidth* lebih kecil.
3. *Search engine friendly*.

4. Menghemat penulisan kode, dikarenakan satu *css* saja dapat dipakai beberapa kali.
5. Mempersingkat waktu kerja, baik saat membuat maupun saat modifikasi halaman *web/blog*.
6. Menghadirkan sesuatu yang tidak dapat dilakukan oleh HTML.
7. Fleksibel dalam penggunaannya dan interaktif serta nyaman dipandang.

3.3.3 Kekurangan CSS

Selain keunggulan tentunya semua bahasa pemrograman mempunyai kekurangan dan berikut kekurangannya yaitu sebagai berikut :

1. Menyita waktu karena tidak semua *browser* dapat mengartikan sintaks CSS yang sama.
2. Tampilan pada *browser* berbeda-beda.
3. Harus tahu cara menggunakannya.

4. Kadang terdapat *browser* yang tidak support CSS oleh *browser* versi lama
5. Adanya *bug/error* dalam CSS.

BAB IV

METODE RAPID APPLICATION DEVELOPMENT (RAD)

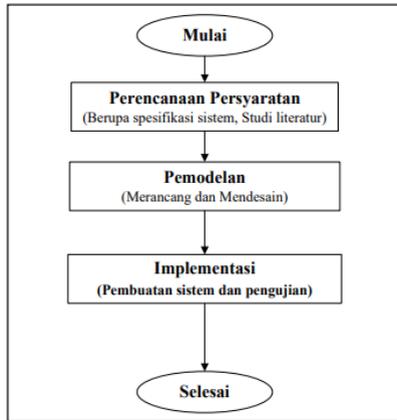
Pada bab ini membahas tentang implementasi dari perancangan yang dibuat sebelumnya serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi tersebut telah berjalan sesuai dengan yang diharapkan.

4.1 Pengertian *Rapid Application Development* (RAD)

Metode yang digunakan dalam penelitian atau tugas kuliah industri ini adalah model RAD (*Rapid application development*). Model RAD berfokus pada pengiriman model kerja yang berulang dan bertahap kepada *user*. RAD merupakan salah satu metode pengembangan suatu sistem informasi dengan waktu yang relatif singkat. Untuk pengembangan suatu sistem

informasi yang normal membutuhkan waktu minimal 180 hari, akan tetapi dengan menggunakan metode RAD suatu sistem dapat diselesaikan hanya dalam waktu 30-90 hari. Metode ini dapat menghasilkan suatu sistem dengan cepat karena sistem yang dikembangkan dapat memenuhi keinginan dari pengguna sehingga dapat mengurangi waktu untuk pengembangan ulang setelah tahap implementasi.

Berikut ini merupakan tahapan-tahapan yang akan dilakukan dalam penyelesaian masalah yang akan dibahas. Berikut adalah tahapan metode RAD.



Gambar 1 . Tahapan Penelitian Menggunakan Metode RAD

4.2 Perencanaan Persyaratan

Pada tahap ini dilakukan pengumpulan informasi terkait beberapa hal berikut:

- a. Pengumpulan informasi tentang bagaimana cara menjalankan aplikasi *Chatbot*.
- b. Pengumpulan informasi tentang bagaimana cara membangun sebuah *Chatbot*

Pada tahap ini, *user* dan analis melakukan pertemuan untuk melakukan identifikasi tujuan

dari aplikasi atau sistem dan melakukan kebutuhan informasi untuk mencapai tujuan. Pada tahap ini hal terpenting adalah adanya keterlibatan dari kedua belah pihak, bukan hanya sekedar persetujuan akan proposal yang sudah dibuat.

4.3 Pemodelan

Pada tahap ini, melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan analis. Untuk tahap ini maka keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain.

4.4 Implementasi

Implementasi merupakan proses pembangunan komponen-komponen pokok sebuah sistem berdasarkan desain yang sudah

dibuat. Tahap ini juga merupakan proses dan penerapan sistem secara utuh, baik dari sisi perangkat keras maupun perangkat lunaknya. Berikut ini pemaparan implementasi yang dilakukan.

4.4.1 Implementasi Interface

Implementasi sistem pakar pada aplikasi *chatbot* ini terletak pada proses pencarian jawaban atas input *query* sesuai dengan pertanyaan dari *user* yang telah disediakan oleh aplikasi. Proses pertama terjadi ketika *user* memasukkan *query* sesuai nomor pertanyaan ke dalam kotak pertanyaan. Setelah itu inputan yang masuk akan dicari berdasarkan *query number* yang dipilih yang ada di dalam *list* pertanyaan. Berikut adalah tabel hasil implementasi algoritma sistem pakar pada aplikasi *chatbot*.

No.	Query Number	Question	Answer
1.	1	Apakah bisa melakukan perubahan KRS Ketika KRS sudah terisi sebelumnya min?	Perubahan KRS boleh dilakukan maksimal seminggu sebelum UTS.
2.	2	Apabila tunggakan untuk pembayaran SPP saat pandemic apakah masih diberi denda min?	Denda SPP selama pandemi ditiadakan.
3.	3	Kapan ujian ulang dilaksanakan min?	Ujian ulang dilaksanakan setelah UTS.
4.	4	Apakah pembayaran KRS bisa menggunakan bank yang lain min?	Pembayaran hanya bisa di bank yang tertera BRI, BNI, BPD dan pembayaran di kampus.
5.	5	Ketika ada tunggakan lalu tidak dibayar apa masih bisa mengikuti ujian min?	Mengikuti ujian bisa namun nilai tidak bisa keluar karena dosen tidak bisa input nilai.
6.	6	Jadi apakah harus ujian ulang min?	Dilihat dari nilai keluar di SIA atau tidak.

7.	7	Untuk kerja praktek itu dilakukan pada semester berapa min?	Untuk kerja praktek lakukan pada semester genap.
8.	8	Untuk KKN dilakukan pada?	Setiap semester selalu ada KKN.
9.	9	Untuk yudisium itu diadakan bulan berapa saja min?	Untuk yudisium diadakan pada bulan Februari dan Agustus.
10.	10	Untuk syarat yudisium yang harus dipenuhi adalah?	<ol style="list-style-type: none"> 1. Surat Keterangan dari Bagian Kemahasiswaan dan Alumni UMBY. 2. Surat Keterangan dari UPT Komputer. 3. Surat Keterangan Bebas Pinjaman dari Perpustakaan Universitas Mercu Buana Yogyakarta. 4. Bukti Penyerahan Skripsi dari Perpustakaan Universitas Mercu Buana Yogyakarta. <ol style="list-style-type: none"> a. Softcopy secara lengkap dalam satu file berbentuk PDF. b. Softcopy Abstrak dalam bentuk MS Word berbahasa Indonesia, Abstrak disertai judul, nama penulis, Instansi (Prodi & Universitas) dan kata kunci (3-5 kata).

			<p>c. Kedua Softcopy tersebut dimasukkan dalam 1 (satu) CD.</p> <p>d. Calon peserta yudisium wajib mengUpload abstrak tersebut ke web.perputakaan UMBY.</p> <p>5. Lembar Foto copy Kartu Mahasiswa yang masih berlaku pada saat mendaftar yudisium.</p> <p>6. Surat Keterangan Bebas Tunggalan/Lunas dari Bagian Registrasi.</p> <p>7. Foto copy ijazah SMU yang telah dilegalisir.</p> <p>8. Foto copy Akte Kelahiran.</p> <p>9. Foto copy halaman Pengesahan Skripsi (lembar 1,2,3)</p> <p>10. Foto copy KRS Terakhir</p> <p>11. Memberikan 2 (dua) buku Informatika</p> <p>12. 1 (satu) CD skripsi untuk Prodi dalam bentuk PDF.</p> <p>13. 1 (satu) CD jurnal skripsi untuk Prodi.</p> <p>14. Lulus TOEFL dengan tanda bukti lulus minimal 450.</p>
11.	11	Untuk wisuda dalam 1 tahun diadakan kapan?	Untuk wisuda sekarang di adakan pada bulan Oktober.
12.	12	Untuk biaya wisuda?	Biaya wisuda Rp 1.300.000

13.	13	Persyaratan untuk pendadaran?	<ol style="list-style-type: none"> 1. Scan kartu mahasiswa yang masih berlaku sebanyak 1 lembar. 2. Scan slip pembayaran SPP, SKS, dan pendadaran. 3. Pas foto 3x4 sebanyak 2 lembar 4. Naskah skripsi rangkap 3 yang telah disetujui oleh dosen pembimbing skripsi 5. Scan lembar konsultasi yang telah disetujui oleh dosen pembimbing skripsi 6. Transkrip nilai sementara (tanpa nilai E dan kosong) 7. Scan lembar konversi (bagi mahasiswa pindahan/alih jalur/transfer) 8. Lembar Surat Pernyataan 2.
-----	----	-------------------------------	--

Tabel 1 . List Pertanyaan

Hasil inputan yang terdapat pada tabel di atas membuktikan bahwa aplikasi mampu mendeteksi angka yang ada pada *query number*.

4.5 Implementasi Aplikasi

4.5.1 Implementasi Interface

Tema yang sederhana dalam pembuatan “aplikasi teknologi *Chatbot* untuk pelayanan di FTI UMBY” ini adalah seperti *chatbot* pada umumnya dimana tampilannya berupa interaksi 2 arah yang dilakukan dalam suatu percakapan. Tema ini diambil dengan beberapa alasan, yaitu:

1. Aplikasi yang dibangun berhubungan dengan dunia pendidikan. Untuk menjaga nilai kependidikannya, maka dipilihlah gambaran sederhana seperti sedang berinteraksi layaknya orang berkomunikasi.
2. Dengan adanya gambaran tersebut dimaksudkan supaya terlihat berkomunikasi antara orang dengan admin tersebut, sehingga *user*

mengerti maksud dari gambaran tersebut jika *user* sedang berbicara dengan mesin.

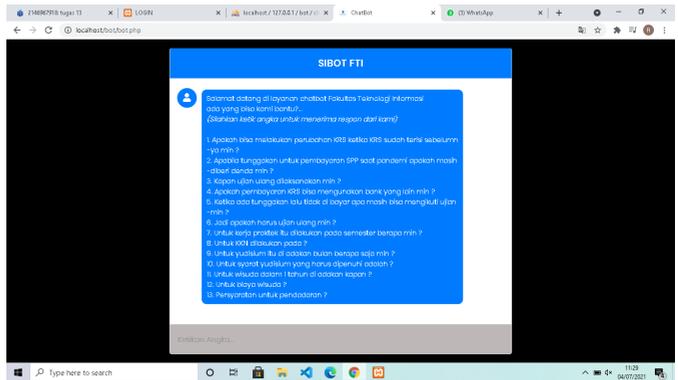
3. Aplikasi ini dibuat dalam suasana percakapan yang telah dibatasi seputar informasi tentang Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta.

Dari hasil perancangan, berikut ini implementasi dari rancangan-rancangan interface tersebut.

- a. Tampilan Awal

Tampilan awal ini muncul ketika program pertama kali dijalankan. Dalam implementasi ini akan ditunjukkan list pertanyaan yang dapat diajukan. Tampilan awal dalam “aplikasi teknologi *Chatbot* untuk pelayanan di FTI UMBY” ditunjukkan oleh Gambar 2 . Tampilan Awal

Aplikasi.

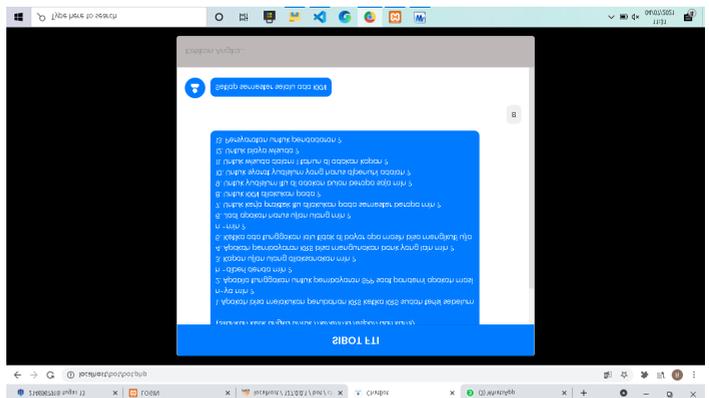


Gambar 2 . Tampilan Awal Aplikasi

b. Tampilan Implementasi *Chatbot*

Tampilan ini merupakan halaman dimana *user* dapat menginputkan *query number* yang ingin ditanyakan sesuai dengan informasi yang ingin didapatkan. Langkah pertama *user* setelah menjalankan aplikasi adalah memilih *query number* yang ingin ditanyakan setelah itu memasukkannya pada *text field* yang disediakan dan

dilanjutkan meng-klik “kirim” berfungsi sebagai enter yang mana *query number* yang telah diketikkan atau pertanyaan yang telah dipilih akan muncul hasil berupa balasan dan itu juga merupakan sebuah proses pencarian jawaban terhadap pertanyaan yang ditanyakan. (lihat Gambar 3. Tampilan Implementasi *Chatbot*).



Gambar 3. Tampilan Implementasi *Chatbot*

BAB V

DEMO SIBOT FTI

5.1 *Source Code*

Bot.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="shortcut icon" href="logo_umbry.png">
  <title>ChatBot</title>
  <link rel="stylesheet" href="style.css">
  <script src="https://kit.fontawesome.com/a076d05399.js"></script>
  <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
</head>
<body bgcolor="black">
  <div class="wrapper">
    <div class="title">SIBOT FTI</div>
    <div class="form">
      <div class="bot-inbox inbox">
        <div class="icon">
```

```

        <i class="fas fa-
user"></i>
    </div>
    <div class="msg-header">
    <p>Salamat datang di layanan
chatbot Fakultas Teknologi Informasi
    <br>ada yang bisa kami bantu
?...<br><i>(Silahkan ketik angka untuk mener
ima respon dari kami)</i></br>
    <br>1. Apakah bisa melakukan
perubahan KRS ketika KRS sudah terisi sebel
umn-ya min ?<br>
    2. Apabila tunggakan untuk
pembayaran SPP saat pandemi apakah masih -
diberi denda min ?<br>
    3. Kapan ujian ulang dilak
sanakan min ?<br>
    4. Apakah pembayaran KRS b
isa menggunakan bank yang lain min ?<br>
    5. Ketika ada tunggakan la
lu tidak di bayar apa masih bisa mengikuti u
jian -min ?<br>
    6. Jadi apakah harus ujian
ulang min ?<br>
    7. Untuk kerja praktek itu
dilakukan pada semester berapa min ?<br>
    8. Untuk KKN dilakukan pad
a ?<br>
    9. Untuk yudisium itu di ad
akan bulan berapa saja min ?</br>

```

```

        10. Untuk syarat yudisium yang harus dipenuhi adalah ?</br>
        11. Untuk wisuda dalam 1 tahun di adakan kapan ? </br>
        12. Untuk biaya wisuda ? </br>
        13. Persyaratan untuk penda
        daran ? </br>
    </p>
    </div>
</div>
<div class="typing-field">
    <div class="input-data">
        <input id="data" type="text"
        placeholder="Ketikan Angka..." required>
        <button id="send-
btn">KIRIM</button>
    </div>
</div>
</div>
</div>
<script>
    $(document).ready(function(){
        $("#send-
btn").on("click", function(){
            $value = $("#data").val();
            $msg = '<div class="user-
inbox inbox"><div class="msg-
header"><p>'+ $value + '</p></div></div>';

```

```

        $(".form").append($msg);
        $("#data").val('');

        // start ajax code
        $.ajax({
            url: 'message.php',
            type: 'POST',
            data: 'text='+$value,
            success: function(result
    ){
                $replay = '<div clas
s="bot-
inbox inbox"><div class="icon"><i class="fas
fa-user"></i></div><div class="msg-
header"><p>'+ result +'</p></div></div>';
                $(".form").append($r
eplay);

                // when chat goes do
wn the scroll bar automatically comes to the
                bottom

                $(".form").scrollTop
                ($(".form")[0].scrollHeight);
            }
        });
    });
</script>

</body>
</html>

```

Message.php

```
<?php
// connecting to database
$conn = mysqli_connect("localhost", "root",
"", "bot") or die("Database Error");

// getting user message through ajax
$getMesg = mysqli_real_escape_string($conn,
$_POST['text']);

//checking user query to database query
$check_data = "SELECT replies FROM chatbot W
HERE queries LIKE '%$getMesg%'";
$run_query = mysqli_query($conn, $check_data
) or die("Error");

// if user query matched to database query w
e'll show the reply otherwise it go to else
statement
if(mysqli_num_rows($run_query) > 0){
    //fetching replay from the database acco
rding to the user query
    $fetch_data = mysqli_fetch_assoc($run_qu
ery);
    //storing replay to a variable which we'l
l send to ajax
    $replay = $fetch_data['replies'];
    echo $replay;
}else{
    echo "Maaf, kami tidak maksud anda!";
}
```



5.2 Demo SIBOT FTI

5.2.1 Langkah-Langkah Uji Coba

Aplikasi ini menggunakan metode *chatting*. Dimana *user* memasukkan *query number* pada *text field* yang sudah disediakan kemudian meng-klik “*irim*” untuk mendapatkan jawaban dari pertanyaan yang diajukan.

Suatu langkah dianggap benar apabila:

1. *User* memasukkan *query number* pada *text field* yang tersedia pada aplikasi. (misal: 4).
2. Sistem membaca *query number* yang dimasukkan sehingga

mendapatkan pertanyaan yang sesuai.

3. Tiap pertanyaan tersebut kemudian dicocokkan dengan data yang sudah dibuat sehingga menghasilkan respon berupa jawaban dari pertanyaan tersebut.
4. Kemudian sistem akan menghentikan pekerjaannya dan memberikan informasi bahwa *query number* yang dimasukkan oleh *user* BENAR dengan menunjukkan respon jawaban yang sesuai.

Suatu langkah dianggap salah apabila:

1. *User* memasukkan kalimat pada *text field* yang tersedia (misal: berapa biaya pendaran?).
2. Sistem membaca kalimat tersebut

sehingga didapatkan pertanyaan.

3. Tiap pertanyaan tersebut kemudian dicocokkan dengan data yang sudah dibuat. Jika pertanyaan tersebut ditemukan dalam data pertanyaan, maka sistem akan memberikan inisialisasi rule dan mendefinisikan respon pertanyaan yang sekaligus memberikan fakta. Yang dimaksudkan respon adalah apakah pertanyaan tersebut benar. Hasil dari langkah ini adalah sebagai berikut:

Inputan	Respon	Jawaban
Berapa biaya pendadaran ?	-	Kami tidak maksud Anda!

Tabel 2 , Hasil Analisis Inputan

4. Jika fakta dari inputan tersebut tidak cocok dengan salah satu query number data pertanyaan, sistem menyatakan kalimat tersebut tidak sesuai dengan respon kata kunci yang ada. Tidak ada yang teridentifikasi, maka tidak ada respon yang benar.

5.2.2 Hasil Demo SIBOT FTI

Tahap ini dilakukan untuk mengetahui kemampuan aplikasi dalam menangani pertanyaan. Proses pengujian dilakukan sesuai dengan batasan masalah yang ada. Proses pengujian ini lebih terfokus pada respon hasil dari masukkan.

Pengujian sistem ini dilakukan dengan menjalankan aplikasi kemudian memasukkan *query number* dalam text field yang telah tersedia.

5.2.3 Pembahasan

Dari hasil uji coba di atas, membuktikan bahwa aplikasi tersebut mampu menyelesaikan permasalahan untuk mengenali angka masukan, apakah angka tersebut merupakan salah satu *query number* atau bukan. Meskipun hampir mendekati sempurna, namun hasil uji coba juga membuktikan bahwa aplikasi ini masih memerlukan penyempurnaan.

BAB VI

KESIMPULAN

Berdasarkan analisis, perancangan dan implementasi pada sistem yang telah dibuat dapat diambil kesimpulan sebagai berikut :

1. Kelompok kami telah berhasil membuat chatbot Pelayanan Di Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta yang dinamai “SIBOT FTI” (Sistem Informasi ChatBot Fakultas Teknologi Informasi) dengan menggunakan Metode RAD.
2. Sistem ini berfungsi untuk menjawab pertanyaan mengenai informasi seputar Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta.
3. Sistem dapat dilakukan dengan cara memilih query sesuai dengan nomor yang ada pada kiri pertanyaan. Dalam kasus ini menggunakan data pertanyaan dan jawaban sesuai dengan pertanyaan yang seringkali diajukan ke TU Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta.

4. Sistem yang telah dibuat cukup mudah untuk digunakan, dapat menampilkan informasi dan memberikan hasil perhitungan sesuai dengan kebutuhan permintaan.

DAFTAR PUSTAKA

- Academy, B. (n.d.). *apa itu rad ? mengenal salah satu metode pengembangan aplikasi*. Retrieved from <https://bitlabs.id/blog/rad-adalah/>
- Aliya, H. (2021, februari 03). *rapid application development, metode pengembangan software yang hemat waktu*. Retrieved from <https://glints.com/id/lowongan/rapid-application-development-adalah/#.YOdOMWQxdPw>
- Nugroho, A. (2019, desember 14). *apa itu html ? inilah pengertian dan sejarah perkembangannya*. Retrieved from <https://qwords.com/blog/pengertian-apa-itu-html/>
- Nugroho, A. (2021, january). *Mengenal kelebihan dan kekurangan php*. Retrieved from <https://qwords.com/blog/kelebihan-php-7/>
- nursyifa, T. (2019, juli 18). *pengeertian, sejarah, fungsi, kelebihan, kekurangan, dan struktur dasar css*. Retrieved from <https://www.plimbi.com/article/175065/-pengertian-sejarah-fungsi-kelebihan-kekurangan-dan-struktur-dasar-css>
- piyaneo. (2014, mei 10). *rapid applicaton development (RAD)*. Retrieved from

<https://piyaneo.wordpress.com/2014/05/10/rapid-application-development-rad/>

Suharjo, I. (2018, Desember 3). *Perbedaan kampus 1 2 3 umby*. Retrieved from <http://imam.mercubuana-yogya.ac.id/blog/2018/12/03/perbedaan-kampus-1-2-3-umby/>

Usamah, D. A. (2018, oktober). *Model rapid application development (RAD)*. Retrieved from <https://dinda3113.wordpress.com/2018/10/04/model-rapid-application-development-rad/>

wawanhidayatullah. (2019, juni 29). *kelebihan dan kekurangan css dalam web*. Retrieved from <https://www.google.com/amp/s/wawanhidayatullah691092551.wordpress.com/2019/06/29/kelebihan-dan-kekurangan-css-dalam-web/amp/>

BIODATA PENULIS

	<p>Putri Taqwa Prasetyaningrum, S.T.,M.T. MCE adalah alumni lulusan dari Teknik Informatika Universitas Atma Jaya Yogyakarta. Beliau juga bekerja sebagai pengajar program studi Sistem Informasi Universitas Mercu Buana Yogyakarta. Beliau banyak melakukan penelitian di bidang Enterprise Information System, Data mining dan Artificial Intelligence. Beliau juga menulis buku dengan judul Pintar Aplikasi Ai Promosi Media Sosial Dengan Php & Mysql. Buku kedua beliau berjudul Penerapan Metode Logika Fuzzy Tsukamoto Untuk Memprediksi Pemasaran Omah Jamu.</p> <p>E-mail : putri@mercubuana-yogya.ac.id</p>
	<p>Haya Azzura Rassy Lahir di Jakarta, 31 Maret 2000. Anak ke-3 dari 3 bersaudara dari pasangan Tri Handoko dan Tazpinusa Indrakesuma. Seorang mahasiswi S1 Informatika Universitas Mercu Buana Yogyakarta. Pernah aktif di Organisasi Himpunan Mahasiswa Teknik Informatika (HMPS-TI) pada tahun 2018-2020. Memiliki ketertarikan dalam seni lukis wajah serta hobi menulis, membaca dan olahraga. Dalam <i>project</i> ini ia bekerja dibagian observasi, pembuat laporan serta penyusun buku.</p>

	<p>Saat ini, ia berdomisili di Yogyakarta. Ia bisa dihubungi lewat akun sosial media Instagram @zurarassya atau</p> <p>E-mail : zurarassya@gmail.com</p> <p>Motto : "Jika tidak menyukai suatu hal, ubahlah. Jika tidak bisa, maka lihat dari sudut pandang lain tentangnya."</p>
	<p>Hasnidar Lahir di Kolaka Utara, 25 Maret 1999. Anak ke-2 dari 3 bersaudara dari pasangan suami istri, Makmur.M dan Fitriani. Mahasiswi S1 Sistem Informasi R1 Universitas Mercu Buana Yogyakarta. Pernah aktif di kelembagaan mahasiswa HMPS-SI pada tahun 2018-2020 dan sekarang sedang aktif di kelembagaan BEM-FTI (Badan Eksekutif Mahasiswa Fakultas Teknologi Informasi) periode 2020-2021. Memiliki ketertarikan terhadap kuliner dan traveling. Dalam <i>project</i> ini ia bekerja di bagian observasi, laporan serta penyusunan buku.</p> <p>Motto : “bekerjalah di masa mudamu, hingga di masa tuamu nanti kamu hanya tinggal bersantai-santai”</p> <p>E-mail : hasnidarmakmur@gmail.com</p>



Fuji Handayani Lahir di Sosa, 17 November 1997. Mempunyai hobi membaca.

E-mail : fujihandayani.13@gmail.com



Mochamad Rizki Darmawan Lahir di Jakarta, 29 Maret 2000. Mempunyai hobi berenang. Dalam *project* ini ia bekerja di bagian pembuatan program.

E-mail : rizkichamad10@gmail.com



M Hamzah Alfani Suri Lahir di Tebing Tinggi, 04 Oktober 1999. Mempunyai hobi bermain *game*. Dalam *project* ini ia bekerja di bagian pembuatan program.

E-mail : alvanhouten11@gmail.com



Ardiansyah Fahrul Ahmadi Lahir di Jayapura, 05 Desember 1996. Mempunyai hobi menikmati senja. .

E-mail : ardiansyahhfahrul12@gmail.com



Miftakhul Khasan Lahir di Musi Banyuasin, 08 April 2000. Mempunyai hobi bermain bola voli. Dalam *project* ini ia bekerja dibagian observasi.

E-mail: Miftakhulkhasan@gmail.com



Mohamad Tri Laksono Lahir di Kotamobagu, 25 April 1998. Mempunyai hobi berenang. Dalam *project* ini ia bekerja di bagian pembuatan program.

E-mail : xtritshukata@gmail.com

Sibot FTI

Aplikasi Chatbot Berbasis Artificial Intelligence

Dewasa ini, bermunculan chatbot-chatbot (sistem komputer yang berbasiskan Bahasa alami atau natural language). Semua chatbot yang bermunculan menggunakan kecerdasan buatan sebagai dasar aplikasi. Walaupun telah banyak bermunculan chatbot-chatbot, tetapi penggunaan dari chatbot itu sendiri belum dimaksimalkan.

Buku ini ditulis agar menjadi sebuah referensi bagi pembaca untuk dapat memahami perancangan aplikasi sistem informasi chatbot untuk pelayanan di Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta. Tujuan dari ditulisnya buku ini agar pembaca dapat memahami penggunaan dan cara kerja dari pengembangan aplikasi chatbot menggunakan bahasa pemrograman PHP dan basis data MySQL.

