

ANALISIS KEEFEKTIFAN CHATGPT DALAM PERANCANGAN APLIKASI

Idham^{1*}, Arif Rahman², M.Rizkillah³

^{1,2,3}Sistem dan Teknologi Informasi, Universitas Muhammadiyah Mataram
email: idham@ummat.ac.id^{1*}

Abstrak: Berdasarkan hasil penelitian yang telah direview, ChatGPT mampu menghasilkan aplikasi sesuai dengan kebutuhan bisnis yang telah ditetapkan. Proses dimulai dari perancangan Use Case Diagram, pembuatan desain database dengan Teknik PDM, hingga penghasilan kode program sesuai dengan permintaan pengguna. Namun, kekurangan dari penelitian-penelitian tersebut terletak pada pembenaran hasil yang diperoleh, karena mayoritas penelitian yang berkaitan dengan ChatGPT dalam bidang IT lebih difokuskan pada analisis keefektifan pembuatan kode program. Sedikit penelitian yang mendetail pada tahapan SDLC menggunakan ChatGPT. Proses penelitian dimulai dengan pembuatan Use Case Diagram, desain database menggunakan Teknik PDM, pembuatan kode program, dan pembuatan skenario pengujian dengan menggunakan ChatGPT. Hasil penelitian menunjukkan bahwa ChatGPT efektif dalam pembuatan Use Case Diagram, desain database dengan Teknik PDM, dan pembuatan skenario pengujian. Namun, efektivitasnya tergantung pada spesifikasi kebutuhan yang detail dan penjelasan fitur yang lengkap dari aplikasi yang ingin dirancang untuk pembuatan skenario pengujian. Dalam pembuatan program, ChatGPT efektif dalam pembuatan program sederhana seperti login, dengan syarat instruksi yang diberikan harus mencakup database yang digunakan dan variabel kode program yang diperlukan. Namun, untuk pembuatan program yang kompleks, ChatGPT belum sepenuhnya efektif. Hal ini karena ChatGPT dapat menciptakan kode program tanpa masalah, programmer harus memberikan instruksi secara sistematis, mulai dari data yang dibutuhkan hingga koreksi yang diperlukan, dan memberikan gambaran kode program agar ChatGPT memiliki pemahaman yang lebih baik dalam pembuatan kode program.

Kata Kunci : ChatGpt, SDLC, UML, Use Case Diagram, PDM, kode program, skenario pengujian

Abstract: Based on the research findings that have been reviewed, ChatGPT is capable of generating applications according to the established business needs. The process start with designing Use Case Diagrams, creating database designs using PDM Technique, and generating program code as per user requests. However, the shortcomings of these studies lie in justifying the obtained results, as the majority of research related to ChatGPT in the IT field is more focused on analyzing the effectiveness of program code generation. There is limited research detailing the stages of SDLC using ChatGPT. The research process begins with creating Use Case Diagrams, designing databases using PDM Technique, coding, and creating testing scenarios using ChatGPT. The research results indicate that ChatGPT is effective in generating Use Case Diagrams, designing databases with PDM Technique, and creating testing scenarios. However, its effectiveness depends on detailed specification of requirements and comprehensive feature explanations of the application intended for testing scenario creation. In program development, ChatGPT is effective in creating simple programs such as login, provided that the given instructions include the database used and the necessary program variables. However, for complex program development, ChatGPT is not yet fully effective. This is because while ChatGPT can generate program code seamlessly, programmers need to provide systematic instructions, starting from the required data to necessary corrections, and provide an overview of the program code so that ChatGPT has a better understanding in program code generation.

Keywords : ChatGPT, SDLC, UML, Use Case Diagram, PDM, Program Code, Testing Scenarios

PENDAHULUAN

ChatGPT merupakan model kecerdasan buatan yang dikembangkan oleh OpenAI[1]. Dalam waktu satu minggu sejak peluncurannya pada 20 November 2022, ChatGPT telah memiliki satu juta pengguna [2]. Potensi ChatGPT dalam merevolusi berbagai bidang, termasuk Pendidikan dan pengolahan esai, telah menjadi perhatian. Di bidang Teknologi informasi, ChatGPT bermanfaat dalam pembuatan kode perangkat lunak serta memberikan bimbingan dengan menjelaskan kode yang telah diberikan. Selain itu, dalam konteks IT, ChatGPT juga digunakan untuk penyediaan basis data dan analisisnya[3].

Dengan ChatGPT yang mampu merevolusi dan mempermudah pekerjaan terutama di bidang IT, banyak penelitian yang telah dilakukan untuk mengevaluasi seberapa efektif penggunaan ChatGPT dalam konteks tersebut. Contohnya, penelitian yang

bertujuan untuk mengurai peran serta kemampuan ChatGPT dalam pemrograman computer, seperti kemampuannya dalam memprediksi dan mengusulkan potongan kode berdasarkan input yang diberikan, memperbaiki kode yang bermasalah, memberikan saran untuk optimalisasi kode, dan sebagainya[4]. Dalam penelitian lainnya, fokusnya adalah pada interaksi antara ChatGPT dan pengembang aplikasi dalam kerangka GIT. Berdasarkan hasil penelitian, sebanyak 52,3% pengembang menggunakan ChatGPT untuk menghasilkan kode program, sementara sisanya menggunakannya untuk memperbaiki kode program yang telah ada[5]. Selain digunakan untuk mengubah atau membuat kode program, ChatGPT juga dimanfaatkan untuk membuat Use Case Diagram. Dalam konteks ini, peneliti menciptakan sebuah program bernama UCDBG (Use-Case Description Generator Based on Transformer Models) yang dikembangkan

menggunakan Python dengan integrasi GPT-3. Dilakukan tiga kali percobaan untuk membuat Use Case dengan tiga kasus yang berbeda, dan hasil yang diperoleh dari UCDGT terbukti sangat akurat[6]. Ketika menggunakan hanya ChatGPT tanpa menyertakan pembuatan program tambahan untuk meningkatkan akurasi, terdapat beberapa kekurangan berdasarkan eksperimen yang dilakukan terhadap 15 objek. Salah satunya adalah ChatGPT hanya mampu menangani model-model kecil[7]. Proses perancangan aplikasi yang komprehensif, berdasarkan SDLC, telah dilakukan dengan menerapkan ChatGPT pada setiap tahap, mulai dari pemahaman kebutuhan aplikasi, perancangan, proses pembuatan kode, hingga tahap pengujian. Berdasarkan eksperimen yang telah dilakukan, ChatGPT terbukti mampu mempercepat pengembangan perangkat lunak [8].

Berdasarkan penelitian-penelitian tersebut, ChatGPT terbukti mampu memenuhi kebutuhan aplikasi berdasarkan business requirement yang ada. ChatGPT dapat memulai proses desain dari perancangan Use Case Diagram hingga pembuatan desain database, dan mampu menghasilkan kode program dengan permintaan yang dimasukkan. Namun, kekurangan dari penelitian-penelitian tersebut terletak pada pembenaran hasil yang diperoleh, karena mayoritas penelitian yang dilakukan terkait ChatGPT dalam bidang IT lebih fokus pada analisis keefektifan pembuatan kode program. Sedangkan masih sedikit penelitian yang melakukan analisis yang mendetail pada tahapan-tahapan SDLC menggunakan ChatGPT.

Sehingga penelitian yang dilakukan difokuskan pada tahapan pembuatan aplikasi dengan menggunakan objek perancangan aplikasi sistem informasi perkuliahan (SIP), menggunakan model SDLC dimana setiap tahap telah memiliki referensi model yang benar dari tim pengembang. Dengan memiliki model yang tepat pada setiap tahapan SDLC, peneliti dapat menilai efektivitas ChatGPT dalam merancang aplikasi.

TINJAUAN PUSTAKA ChatGPT

ChatGPT merupakan chatbot percakapan yang dikembangkan oleh OpenAI yang menggunakan pemrosesan bahasa alami dalam menghasilkan tanggapan yang sesuai dengan tanggapan yang dibuat oleh manusia[9].

SDLC

SDLC merupakan proses berkelanjutan mulai dengan planning, analisis, desain dan implementasi [10].

Waterfall

Waterfall merupakan alur hidup klasik dengan pendekatan alur hidup perangkat lunak secara berurutan dimulai dari analisis, desain pengkodean, pengujian dan tahap pendukung [11].

UML dan Diagram Use Case

UML atau Unified Modeling Language merupakan salah satu tools yang handal untuk pengembangan sistem yang berorientasi objek. Sedangkan Diagram Use Case merupakan teknik merekam kebutuhan fungsional sebuah sistem[12].

Physical Data Model

PDM atau Physical Data Model merupakan model yang menggunakan beberapa tabel dalam menggambarkan data serta relasi antara data [13].

Codeigniter

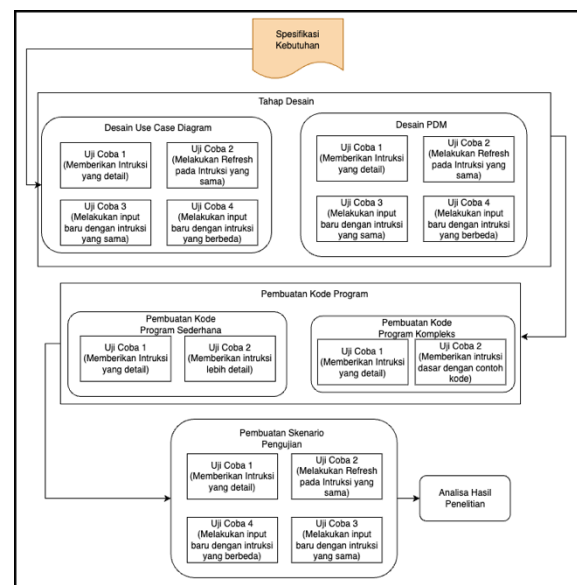
Codeigniter merupakan framework yang mengimplementasikan aturan MVC berbasis PHP. Kelebihan dari Codeigniter salah satunya adalah mempercepat pengerjaan aplikasi tanpa harus menulis code program dari awal [14].

Black Box Testing dan Teknik Equivalence Partitions

Black Box Testing memastikan setiap proses telah berfungsi sesuai dengan kebutuhan yang diharapkan. Equivalence Partitions adalah metode pengujian yang menerapkan masukan pada setiap menu yang ada pada sistem informasi[15].

METODE

Metode penelitian ini menerapkan SDLC (Software Development Life Cycle), yang mencakup semua aktivitas untuk pengembangan perangkat lunak melalui tahapan-tahapan yang telah ditentukan[16]. Pada penelitian ini, SDLC yang digunakan adalah Waterfall, dimana setiap pekerjaan dalam setiap fase harus diselesaikan sebelum beralih ke fase berikutnya[17]. Waterfall yang digunakan oleh peneliti akan disesuaikan dengan SDLC yang digunakan untuk AI generatif[8].



Gambar 1. Metode Penelitian

Tahap pertama adalah spesifikasi kebutuhan, yang berguna untuk merumuskan spesifikasi yang diperlukan dalam merancang suatu sistem[18]. Peneliti telah memiliki spesifikasi kebutuhan untuk sistem informasi perkuliahan yang akan digunakan pada tahap selanjutnya.

Tahap desain terdiri dari dua sub-tahap, yaitu desain basis data dan desain sistem[19]. Pada tahap desain sistem, digunakan UML untuk menjelaskan akur dan fungsi sistem[20]. Penggunaan UML hanya terbatas pada pembuatan diagram Use Case. Diagram Use Case digunakan untuk memahami fungsi-fungsi dalam sistem[21]. Desain Database dilakukan dengan menggunakan model Pyhsical Data Model yang menggambarkan struktur tabel, termasuk kolom, tipe data dan hubungan antar tabel[22].

Bahasa pemrograman yang digunakan pada tahap penulisan program adalah PHP dengan framework Codeigniter 3. Codeigniter menerapkan konsep MVC (Model View Controller) dalam struktur kodenya [23].

Tahap berikutnya adalah tahap pengujian, yang bertujuan untuk mendeteksi kembungkinan kesalahan pada sistem[24]. Metode pengujian yang digunakan adalah Black Box Testing dengan Teknik Equivalence Partitions, dimana sistem diuji dengan memasukkan data ke ap;ikasi sesuai dengan kasus uji tertentu[15].

Tahap selanjutnya adalah analisis hasil dari setiap tahap yang telah dilakukan, dimana efektivitas penggunaan ChatGPT dievaluasi pada setiap tahapan. Analisis efektivitas perancangan aplikasi dimulai dari tahap deasin, pembuatan kode program, hingga pembuatan skenarion pengujian, Pada tahap desain, termasuk pembuatan Use Case Diagram, Desain PDM, dan pembuatan skenario pengujian, peneliti Menyusun 4 skenario uji coba. Pegujian dengan 4 skenario dilakukan untuk membandingkan hasil yang dihasilkan oleh ChatGPT dengan hasil yang dibuat oleh pengembang, guna mengetahui kesesuaian antara keduanya.

Pada Uji Coba 1, peneliti memberikan intruksi detail, yaitu spesifikasi kebutuhan sebagai data yang harus digunakan oleh ChatGPT untuk membuat Use Case Diagram dan Desain PDM. Pada Uji Coba 2, peneliti melakukan refresh pada ChatGPT untuk memperbarui hasil yang ditampilkan. Disini, akan diperiksa apakah terdapat perbedaan hasil dengan uji coba sebelumnya. Pada Uji Coba 3, peneliti menghapus direktori ChatGPT sebelumnya dan kemudian menduplikasi instruksi dari Uji Coba sebelumnya untuk melihat apakah jawaban yang dihasilkan berbeda dari uji coba sebelumnya. Sedangkan pada Uji Coba 4, peneliti membuat direktori baru dan mengubah intruksi sebelumnya untuk melihat apakah hasil yang dihasilkan oleh ChatGPT berbeda atau tidak.

Tahap pembuatan kode program dibagi menjadi dua bagian. Pertama, pembuatan kode program sederhana dengan fitur login. Kemudian, pembuatan kode program kompleks dengan fitur yang digunakan adalah laporan absensi. Tujuanya adalah untuk mengamati apakah ChatGPT mampu menghasilkan kode program sesuai dengan intruksi yang diberikan. Terdapat dua uji coba dalam tahap pembuatan kode program ini. Pada uji coba 1, peneliti

memberikan instruksi lengkap seperti menambahkan nama tabel dan kolom yang ada pada database yang digunakan, serta nama halaman redirect yang ditentukan. Pada uji coba 2, peneliti memberikan instruksi yang sama dengan uji coba 1, namun dengan penambahan kode program untuk membantu ChatGPT menghasilkan kode program yang lebih presisi. Hasil penelitian digunakan untuk menganalisis hasil yang diperoleh pada proses ChatGPT dalam tahap desain, pembuatan kode program, dan pembuatan skenario pengujian.

Untuk seluruh proses pembuatan dengan ChatGPT, peneliti memberlakukan batasan koreksi terhadap hasil yaitu maksimal 7 kali hingga hasil desain, skenario pengujian, dan kode program sudah sesuai dan tidak mengalami masalah.

HASIL DAN PEMBAHASAN

A. Spesifikasi Kebutuhan

Peneliti sudah memiliki spesfikasi kebutuhan sistem informasi perkuliahan sebagai berikut :

Tabel 1. Spesifikasi Kebutuhan

No	Spesifikasi Kebutuhan
1	Admin dapat menambah dosen dengan data meliputi jenis_user yaitu admin dan dosen, nama, email, nidn dan juga password
2	Admin dan Dosen bisa melakukan login
3	Admin dan Dosen bisa menambah data mahasiswa meliputi nama dan nim
4	Admin dan Dosen bisa menambah data kelas yaitu nama kelas
5	Admin dan Dosen bisa menambah data mata kuliah meliputi nama, kode mata kuliah dan juga semesternya
6	Admin dan Dosen bisa membuat kelas mata kuliah dengan cara memilih mata kuliah yang sudah dibuat di point 5, memilih kelas yang sudah dibuat di point 4, memilih dosen yang sudah dibuat di point 1, menambah total pertemuan kelas tersebut, memilih perwakilan kelas dari data mahasiswa yang sudah dibuat di point 3 dan menambah tanggal dimulainya kelas
7	Admin dan Dosen bisa membuat kelas mahasiswa dimana admin dan dosen memilih kelas mata kuliah yang sudah dibuat di point 6 dan memilih mahasiswa yang sudah dibuat di point 3
8	Admin dan Dosen bisa membuat jenis nilai dengan cara memilih mata kuliah yang dibuat di point 5, membuat nama dari jenis nilai tersebut seperti nilai UTS, UAS, tugas dan lain-lain, menambah jenis perhitungan dan terakhir manambah bobot penilaian pada masing-masing jenis nilai dalam bentuk percent
9	Admin dan Dosen dapat memasukan nilai mahasiswa dengan cara memilih kelas mata kuliah yang dibuat pada point 6 dulu, setelah itu memilih jenis nilai yang ingin di isi nilainya, kemudian memilih mahasiswa yang dibuat di point 3, lalu memasukkan Nama nilanya dan input jumlah nilainya
10	Mahasiswa dapat melakukan absensi dengan kondisi sistem sudah berhasil login baik menggunakan akun admin ataupun akun dosen kemudian memilih kelas mata kuliah yang dibuat pada point 6 dulu, setelah itu mahasiswa memilih data mahasiswanya yang dibuat pada point 3, menambah tanggal absensi yang dimasukan secara otomatis ataupun tanggal absensi yang dapat dipilih, kemudian menambah status masuk yang diinputkan sesuai dengan jam masuk untuk menentukan apakah mahasiswa ini telat atau tidak
11	Admin dan Dosen dapat mendownload jurnal perkuliahan yang berisi Pertemuan, hari/tanggal, judul bab, rincian

materi, metode pengajaran online/offline, jumlah hadir mahasiswa dan paraf dosen
--

B. Tahap Desain

Tahap desain dimulai dengan pembuatan salah satu UML yaitu Use Case Diagram pada ChatGPT, berikut hasil dari pembuatan usecase diagram tersebut :

Tabel 2. Desain Use Case Diagram dengan ChatGPT

Uji Coba	Intruksi	Jumlah Koreksi
Uji Coba 1 (memberikan data yang detail)	bisa membantu saya dalam merancang Use Case Diagram dengan spesifikasi kebutuhan sebagai berikut : (peneliti memberikan spesifikasi kebutuhan)	0
Uji Coba 2 (melakukan refresh pada Intruksi untuk melihat apakah hasil memiliki perubahan)	bisa membantu saya dalam merancang Use Case Diagram dengan spesifikasi kebutuhan sebagai berikut : (peneliti memberikan spesifikasi kebutuhan)	0
Uji Coba 3 (melakukan input baru dengan intruksi yang sama)	bisa membantu saya dalam merancang Use Case Diagram dengan spesifikasi kebutuhan sebagai berikut : (peneliti memberikan spesifikasi kebutuhan)	0
Uji Coba 4 (melakukan input baru dengan intruksi yang berbeda)	Buatkan saya Use Case Diagram menggunakan spesifikasi kebutuhan berikut ini : (peneliti memberikan spesifikasi kebutuhan)	0

Tahap selanjutnya adalah membuat Desain Database dengan model Physical Data Model, berikut proses pembuatan PDM tersebut pada ChatGPT :

Tabel 3. Desain PDM dengan ChatGPT

Uji Coba	Intruksi	Jumlah Koreksi
Uji Coba 1 (memberikan data yang detail)	buatkan saya desain database dengan model PDM berdasarkan spesifikasi kebutuhan berikut ini : (peneliti memasukkan spesifikasi kebutuhan)	1
Uji Coba 2 (melakukan refresh pada Intruksi untuk melihat apakah hasil memiliki perubahan)	buatkan saya desain database dengan model PDM berdasarkan spesifikasi kebutuhan berikut ini : (peneliti memasukkan spesifikasi kebutuhan)	1
Uji Coba 3 (melakukan input baru dengan intruksi yang sama)	buatkan saya desain database dengan model PDM berdasarkan spesifikasi kebutuhan berikut ini : (peneliti memasukkan spesifikasi kebutuhan)	1
Uji Coba 4 (melakukan input baru dengan intruksi yang berbeda)	buatkan saya desain database dengan model PDM berdasarkan spesifikasi kebutuhan berikut ini : (melakukan penambahan spesifikasi yaitu admin dapat menambah jenis user seperti super admin)	0

dan dosen merubah point 11 dengan menambahkan kalimat jurnal perkuliahan tidak ada peroses penginputan data)
--

C. Pembuatan Kode Program

Langkah selanjutnya adalah tahap pembuatan kode program. Berikut proses pembuatan kode program pada fitur yang sudah ditentukan :

Tabel 4. Pembuatan Kode Program Sederhana dengan ChatGPT

Uji Coba	Intruksi	Jumlah Koreksi
Uji Coba 1 (memberikan intruksi detail)	buatkan saya kode program login kodeigniter 3 dengan nama table databasenya adalah user dengan kolom id,id_jenis_user,nama,email,nidn,password,sts_hapus. Jika login berhasil maka akan diredirect ke halaman admin/dashboard jika gagal akan kehalaman login dan data session yang disimpan adalah id_admin dan id_jenis_user	3
Uji Coba 2 (memberikan intruksi lebih detail)	buatkan saya kode program login kodeigniter 3 dengan nama table databasenya adalah user dengan kolom id,id_jenis_user,nama,email,nidn,password,sts_hapus. Jika login berhasil maka akan diredirect ke halaman admin/dashboard jika gagal akan kehalaman login dan data session yang disimpan adalah id_admin dan id_jenis_user. Gunakan password verify karena jenis password menggunakan hash. Kemudian tambahkan set flashdata jika tidak berhasil login, 118scenario118i set flashdata sebagai berikut : 'message', '<div class="alert alert-danger" role="alert">Wrong password!</div>'	0

Tabel 5. Pembuatan Kode Program Kompleks dengan ChatGPT

Uji Coba	Intruksi	Jumlah Koreksi
Uji Coba 1 (memberikan intruksi detail)	Uji Coba 1 (memberikan intruksi yang lengkap)	7
Uji Coba 2 (memberikan intruksi dasar dengan contoh kode program)	saya bertanya, saya ingin membuat laporan absensi dengan tampilan dibawah ini dan dibuat menggunakan kodeigniter 3, dan dibawa sudah saya sediakan database yang bisa digunakan untuk membuat laporan absensi ini, apakah bisa dibantu	0

dengan data-data ini ??
(peneliti memberikan tabel
html dari contoh laporan
absensi yang ingin dibuat)
berikut database mysql
yang dibutuhkan untuk
membuat laporan absensi
diatas : (peneliti
memberikan database yang
dibutuhkan)

D. Pembuatan Skenario Pengujian

Tahap selanjutnya adalah tahap pengujian menggunakan Teknik Equivalence Partitions, berikut hasil pembuatan scenario pengujian megggunakan ChatGpt :

Tabel 6. Hasil Pembuatan Skenario Pengujian

Uji Coba	Intruksi	Jumlah Koreksi
Uji Coba 1 (memberikan data yang detail)	buatkan saya skenario tes sistem informasi perkuliahan per fitur dengan Teknik Equivalence Partitions dengan data yang harus digunakan untuk membuat skenario tes sebagai berikut : (peneliti memberikan data detail fitur dan gambaran fitur terkait sistem)	0
Uji Coba 2 (melakukan refresh pada Intruksi untuk melihat apakah hasil memiliki perubahan)	buatkan saya skenario tes sistem informasi perkuliahan per fitur dengan Teknik Equivalence Partitions dengan data yang harus digunakan untuk membuat skenario tes sebagai berikut : (peneliti memberikan data detail fitur dan gambaran fitur terkait sistem)	0
Uji Coba 3 (melakukan input baru dengan intruksi yang sama)	buatkan saya skenario tes sistem informasi perkuliahan per fitur dengan Teknik Equivalence Partitions dengan data yang harus digunakan untuk membuat skenario tes sebagai berikut : (peneliti memberikan data detail fitur dan gambaran fitur terkait sistem)	0
Uji Coba 4 (melakukan input baru dengan intruksi yang berbeda)	buatkan saya skenario tes sistem informasi perkuliahan per fitur dengan Teknik Equivalence Partitions menggunakan data detail fitur di sistemnya berikut ini : (peneliti memberikan data detail fitur dan gambaran fitur terkait sistem)	0

Berdasarkan hasil penelitian. Pembuatan Use Case Diagram menggunakan ChatGPT melalui 4 uji coba berhasil menghasilkan diagram yang serupa dengan dibuat oleh pengembang tanp perlu adanya koreksi. Hal ini disebabkan oleh spesifikasi kebutuhan yang diberikan peneliti pada setiap fitu,

sehingga Use Case Diagram yang dihasilkan sama dengan yang dibuat pengembang.

Kemudian, dalam pembuatan PDM menggunakan ChatGPT pada Uji Coba1, Uji Coba 2, dan Uji Coba 3, terdapat satu kali koreksi karena hasil yang diperoleh masih kurang dalam hal jumlah tabel, serta penambahan satu tabel yang tidak ada pada PDM yang dibuat oleh pengembang. Pada Uji Coba 4, peneliti memperbarui spesifikasi dengan lebih mentail, khususnya terkait penejelasan fungsionalitas sistem. Hal ini mengakibatkan hasil yang diperoleh sesuai dengan PDM yang dirancang oleh pengembang, dan tidak memerlukan koreksi lebih lanjut.

Pada tahap pembuatan kode program sederhana dengan ChatGPT, pada Uji Coba 1, peneliti memberikan intruksi dengan menjabarkan nama tabel beserta kolomnya, serta mengindikasikan arah pengguna setelah berhasil atau gagal login. Peneliti juga mencantumkan data apa saja yang harus ditambahkan. Hasilnya, peneliti melakukan koreksi sebanyak 3 kali untuk memperbaiki kode tersebut. Setelah koreksi ketiga, kode tersebut berhasil berjalan dengan baik. Pada Uji Coba 2, tidak ada koreksi dikarenakan hasil yang dikeluarkan sudah sesuai.

Tahap pembuatan kode program kompleks dengan ChatGPT mengalami bebera perbedaan antara Uji Coba 1 dan Uji Coba 2. Pada Uji Coba 1, terjadi kegagalan dalam memberikan kode program sesuai dengan instruksi, dimana jumlah koreksi yang diberikan oleh peneliti telah mencapai 7 kali. Ini disebabkan oleh kompelsitas instruksi yang diberikan peneliti, yang meliputi 3 jenis data, yaitu kode HTML untuk laporan, daftar tabel yang digunakan, dan instruksi yang rumit mengenai penyusunan laporan. Sebaliknya, paeda Uji Coba 2, peneliti hanya memberikan 2 jenis data, yaitu kode HTML untuk laporan dan daftar database. Setelah hasil diperoleh, dilakukan koreksi untuk menambahkan data atau nilai tertentu guna memperbaiki hasil. Peneliti juga memberikan instruksi berupa contoh kode program, dengan tujuan agar ChatGPT dapa berkerja secara sistematis dan focus dalam menyelesaikan masalah koreksi yang diberikan. Hasil dari Uji Coba 2 tidak memerlukan koreksi tambahan untuk memperbaiki error dalam program.

Selanjutnya,hasil pembuatan skenario pengujian menggunakan Teknik Equivalence Partitions dengan 4 jenis Uji Coba telah sesuai dengan pembuatan skenarion pengujian oleh pengembang. Fitur-fitur yang terdapat dalam sistem, sehingga tidak diperlukan koreksi tambahan dalam pembuatan skenarion pengujian.

KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, ChatGPT terbukti efektif dalam beberapa aspek pembangunan perangkat lunak yaitu pembuatan Use Case Diagram, desain database dengan Teknik PDM, dan pembuatan

skenario pengujian, asalkan aplikasi yang akan dirancang memiliki spesifikasi kebutuhan yang detail dan penjabaran fitur yang lengkap untuk kebutuhan pembuatan skenario pengujian.

Dalam pembuatan program, ChatGPT efektif dalam menghasilkan program-program sederhana seperti fitur login, dengan syarat intruksi yang diberikan harus mencakup database yang digunakan dan variable variable kode program yang dibutuhkan. Namun, untuk pembuatan program yang kompleks, ChatGPT belum sepenuhnya efektif. Hal ini disebabkan karena dalam menciptakan kode program yang sesuai dan bebas dari masalah, programmer harus Menyusun instruksi secara sistematis, dimulai dari data yang diperlukan dan memberikan gambaran kode program yang jelas saat melakukan koreksi, sehingga ChatGPT dapat memiliki pemahaman yang lebih baik dalam pembuatan kode program.

Saran untuk penelitian selanjutnya agar berfokus pada pengembangan aplikasi untuk pembuatan desain usecase, database, program dan skenario testing dengan memanfaatkan API ChatGPT yang sudah disediakan oleh OpenAI.

DAFTAR PUSTAKA

- [1] P. P. Ray, "ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope," *Internet Things Cyber-Phys. Syst.*, vol. 3, pp. 121–154, 2023, doi: 10.1016/j.iotcps.2023.04.003.
- [2] P. Thaware, S. Trivedi, and P. R. Lakra, "ChatGPT: is it really a threat to medical research paper writing?," *Ain-Shams J. Anesthesiol.*, vol. 15, no. 1, p. 67, Aug. 2023, doi: 10.1186/s42077-023-00365-z.
- [3] M. Halaweh, "ChatGPT in education: Strategies for responsible implementation," *Contemp. Educ. Technol.*, vol. 15, no. 2, p. ep421, Apr. 2023, doi: 10.30935/cedtech/13036.
- [4] "4.pdf."
- [5] K. Jin, C.-Y. Wang, H. V. Pham, and H. Hemmati, "Can ChatGPT Support Developers? An Empirical Evaluation of Large Language Models for Code Generation." Mar. 16, 2024. doi: 10.1145/3643991.3645074.
- [6] A. T. Imam and I. Altawailha, "The Use of the Pre-Trained BERT and GPT-3 Models to Automate the Composing of Use Case Descriptions." Aug. 14, 2023. doi: 10.22541/au.169205044.43001223/v1.
- [7] J. Cámara, J. Troya, L. Burgueño, and A. Vallecillo, "On the assessment of generative AI in modeling tasks: an experience report with ChatGPT and UML," *Softw. Syst. Model.*, vol. 22, no. 3, pp. 781–793, Jun. 2023, doi: 10.1007/s10270-023-01105-5.
- [8] A. Rajbhoj, A. Somase, P. Kulkarni, and V. Kulkarni, "Accelerating Software Development Using Generative AI: ChatGPT Case Study," in *Proceedings of the 17th Innovations in Software Engineering Conference*, Bangalore India: ACM, Feb. 2024, pp. 1–11. doi: 10.1145/3641399.3641403.
- [9] C. K. Lo, "What Is the Impact of ChatGPT on Education? A Rapid Review of the Literature," *Educ. Sci.*, vol. 13, no. 4, p. 410, Apr. 2023, doi: 10.3390/educsci13040410.
- [10] A. S. Maharani, N. R. Naafian, and A. Pantjarani, "Sistem Informasi Buku Tamu Berbasis Web Menggunakan PHP Framework Laravel dan MySQL Di Kecamatan Jatisrono Wonogiri".
- [11] A. R. F. Hadinata, N. Suarna, and F. M. Basysyar, "Sistem Informasi Penjualan Berbasis Web Dengan Metode Waterfall Untuk Meningkatkan Area Pemasaran".
- [12] S. Supriyanta, D. Supriadi, and B. Susanto, "Perancangan Sistem Informasi Penggajian Karyawan Dengan metode Waterfall," *Indones. J. Comput. Sci.*, vol. 1, no. 1, pp. 1–6, Apr. 2022, doi: 10.31294/ijcs.v1i1.1040.
- [13] F. Irwanda, S. A. Ferary, S. A. Kamila, and B. F. Kartono, "PERANCANGAN SISTEM INFORMASI PENJUALAN UMKM ANDIN DAN TUDUNG SAJI BERBASIS WEBSITE MENGGUNAKAN METODE WATERFALL," *J. Pengabd. Kpd. Masy.*.
- [14] Y. Christian and D. Alfath, "Perancangan Sistem Manajemen Kerja Harian Berbasis Website Menggunakan Framework Codeigniter di Universitas Internasional Batam," vol. 1, no. 1, 2021.
- [15] A. Fahrezi, F. N. Salam, G. M. Ibrahim, R. Rahman, and A. Saifudin, "Pengujian Black Box Testing pada Aplikasi Inventori Barang Berbasis Web di PT. AINO Indonesia," vol. 1, no. 1, 2022.
- [16] B. Acharya and P. K. Sahu, "SOFTWARE DEVELOPMENT LIFE CYCLE MODELS: A REVIEW PAPER".
- [17] G. Gurung, R. Shah, and D. P. Jaiswal, "Software Development Life Cycle Models-A Comparative Study," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, pp. 30–37, Jul. 2020, doi: 10.32628/CSEIT206410.
- [18] N. Hasanah and M. N. Indriawan, "Rancangan Aplikasi Batam Travel Menggunakan Metode Software Development Life Cycle (SDLC)," vol. 1, no. 1, 2021.
- [19] Y. S. Dwanoko, "IMPLEMENTASI SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) DALAM PENERAPAN PEMBANGUNAN APLIKASI PERANGKAT LUNAK," *J. Teknol. Inf.*, Oct. 2016, doi: 10.36382/jti-tki.v7i2.219.

- [20] R. Abdillah, A. Kuncoro, and I. Kurniawan, "ANALISIS APLIKASI PEMBELAJARAN MATEMATIKA BERBASIS ANDROID DAN DESAIN SISTEM MENGGUNAKAN UML 2.0," vol. 4, 2019.
- [21] Institut Teknologi Sepuluh Nopember *et al.*, "A Different Approach on Automated Use Case Diagram Semantic Assessment," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 1, pp. 496–505, Feb. 2021, doi: 10.22266/ijies2021.0228.46.
- [22] D. O. Sumadya, H. H. Ginardi, and R. J. Akbar, "Perancangan dan Implementasi Basis Data Aplikasi Web Fotokita," *J. Tek. ITS*, vol. 5, no. 2, pp. A552–A555, Oct. 2016, doi: 10.12962/j23373539.v5i2.18771.
- [23] Y. Pratama, "SISTEM PRAKTEK KERJA LAPANGAN (PKL) PADA SEKOLAH TINGGI XYZ MENGGUNAKAN CODEIGNITER," vol. 3, 2023.
- [24] Ichsan Raksa Gumilang, "PENERAPAN METODE SDLC (SYSTEM DEVELOPMENT LIFE CYCLE) PADA WEBSITE PENJUALAN PRODUK VAPOR: APPLICATION OF SDLC (SYSTEM DEVELOPMENT LIFE CYCLE) METHOD ON VAPOR PRODUCT SALES WEBSITE," *JURAL Ris. RUMPUN ILMU Tek.*, vol. 1, no. 1, pp. 47–56, Apr. 2022, doi: 10.55606/jurritek.v1i1.144.