

# PENERAPAN METODE *RATIONAL UNIFIED PROCESS* (RUP) DALAM PENGEMBANGAN SISTEM INFORMASI *MEDICAL CHECK UP* PADA CITRA MEDICAL CENTRE

*(The Application of Rational Unified Process (RUP) in Development of a Medical  
CheckUp Information System at Citra Medical Centre)*

Ririn Perwitasari\*, Royana Afwani, Sri Endang Anjarwani

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Mataram

Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

Email: ririnperwita2104@gmail.com, [royana, endang]@unram.ac.id

## Abstract

The Rational Unified Process (RUP) method is an iterative and incremental software development method with four stages. The first is inception stage to defining the system requirements and use case diagram, then designed previously issued by the user for conformity. Second stage is elaboration stage to improvement requirement and use case diagram, and designing another diagram design till interface design. Test elaboration by validating features with user. Construction stage direct implementation of the design that has been made in before stage and carried out testing using black box testing. The last is transition stage will be tested at the end of the system that has been built. Testing is done using questionnaire conducted by 2 people from the clinic and 30 people from the community with the results obtained amounted to 86.12% (very feasible) so the RUP method is a good method to be used in system development.

**Keywords:** Rational Unified Process, RUP, Information System, Medical CheckUp.

\*Penulis Korespondensi

## 1. PENDAHULUAN

Metode RUP merupakan metode pembangunan perangkat lunak yang *iterative* dan *incremental* serta berfokus pada arsitektur. Metode RUP dapat menangani risiko yang berhubungan dengan pengembangan kebutuhan sistem berdasarkan perubahan yang diinginkan oleh klien. Untuk mengurangi risiko tersebut dilakukan dengan pengujian pada setiap akhir tahapan RUP, sehingga akan mudah melakukan perubahan sebelum mencapai tahap akhir. Metode RUP juga mengutamakan kepuasan pengguna sehingga lebih sering melakukan interaksi dengan pengguna [1].

Metode RUP digunakan dalam pengembangan sistem informasi *medical checkup* pada Citra Medical Centre mulai dari proses pendaftaran pasien, pengolahan hasil pemeriksaan dan penyerahan hasil pemeriksaan ke pasien. *Medical checkup* merupakan pemeriksaan kesehatan secara menyeluruh mulai dari pemeriksaan fisik serta pemeriksaan laboratorium [2]. Salah satu tempat pemeriksaan kesehatan yang telah ditetapkan oleh Dirjen Bina Upaya Kesehatan Badan Nasional Penempatan dan Perlindungan Tenaga Kerja

Indonesia (BNP2TKI) adalah Citra Medical Centre yang terletak di kawasan Lombok Timur [3]. Dalam pelaksanaannya, Citra Medical Centre telah menggunakan sebuah aplikasi berbasis *desktop* yang digunakan sebagai pengelolaan data pasien. Namun adanya aplikasi *desktop* tersebut belum memenuhi kebutuhan pegawai karena pengaksesannya terbatas hanya pada jam kantor. Oleh karena itu, dilakukan pengembangan aplikasi berbasis *web* dengan menggunakan metode RUP untuk melakukan pengelolaan data pasien dari mana saja dan kapan saja.

Metode RUP digunakan dalam pengembangan sistem informasi *medical checkup* untuk mengantisipasi pendefinisian kebutuhan sistem yang tidak terlalu detail pada tahap awal. Dengan menggunakan metode RUP, hal tersebut dapat diatasi dengan adanya pengujian yang dilakukan oleh pengguna pada setiap akhir tahapan RUP untuk mengetahui kebutuhan sistem dan kesesuaian proses pengembangan sistem pada setiap tahapan. Tujuan melibatkan pengguna dalam setiap proses dalam RUP adalah untuk menghasilkan sistem yang sesuai dengan harapan pengguna.

## 2. TINJAUAN PUSTAKA DAN DASAR TEORI

### 2.1. Tinjauan Pustaka

Penelitian tentang sistem informasi pendistribusian suku cadang industri pada CV.Oilfindo menggunakan metode RUP untuk membantu karyawan dalam memproses pesanan dan pembuatan laporan sehingga pekerjaan karyawan menjadi efektif dan efisien[4]. Perbedaannya dengan penelitian yang akan dilakukan adalah fokus penelitiannya pada sistem yang dibangun, sedangkan penelitian yang akan dilakukan berfokus pada metodenya yaitu menggunakan RUP dalam pengembangan sistemnya.

Penelitian tentang analisis dan pengembangan sistem ujian akhir semester berbasis *web* di SMK Negeri 1 Malang bertujuan untuk penyelenggaraan ujian akhir semester yang digunakan oleh guru dan siswa [5]. Perbedaannya dengan penelitian ini adalah jenis penelitiannya, di mana peneliti sebelumnya terletak pada analisis hasil pengembangan sistem, sedangkan pada penelitian yang penulis lakukan yaitu penerapan metode dalam pengembangan sistem. Persamaan antara kedua penelitian adalah penggunaan metode RUP dalam pengembangan sistem yang memungkinkan perubahan selama proses pengembangan.

Penelitian tentang penerapan metode SDLC RUP (*Rational Unified Process*) dalam pembuatan sistem informasi E-kos berbasis *web* pada kota malang dibangun untuk memudahkan mahasiswa dalam mencari tempat kos dan juga memudahkan pemilik kos untuk mengiklankan kosnya[6]. Perbedaannya dengan penelitian yang akan dilakukan adalah jenis penelitian yang dilakukan. Penelitian sebelumnya membangun sistem informasi untuk memudahkan pencarian kos, sedangkan penelitian yang akan dilakukan adalah pengembangan sistem informasi *medical checkup* pada klinik. Persamaan penelitian yang akan dilakukan dengan penelitian sebelumnya yaitu menggunakan metode *Rational Unified Process* (RUP) dalam pengembangan sistem, di mana keempat tahap RUP akan membantu peneliti untuk merancang sistem informasi yang akan dibangun.

Penelitian yang berjudul *a comparison between two software engineering process, RUP and waterfall models* pengembangan sistem informasi data *medicalcheckup* bertujuan untuk membandingkan antara dua model pengembangan perangkat lunak yaitu RUP dan *waterfall* untuk menunjukkan fitur dan kekurangan masing-masing model [7]. Kelebihan dari RUP adalah fleksibel jika *user* ingin melakukan perubahan pada setiap tahapan pengembangan, sedangkan untuk *waterfall* tidak dapat terlalu banyak

melakukan revisi atau perubahan. Kelemahan RUP adalah prosesnya yang bisa jadi terlalu banyak karena dimungkinkan adanya perubahan, sedangkan *waterfall* menggunakan tahapan pasti sehingga mudah digunakan.

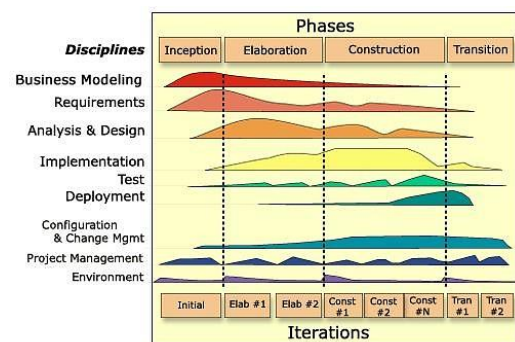
Pada penelitian *a comparison between three SDLC models waterfall model, spiral model, and incremental / iterative model* bertujuan untuk membandingkan ketiga model SDLC tersebut[8]. Perbedaan ketiga model dari segi *flexibility*, model *waterfall* merupakan model yang paling tidak fleksibel, model spiral agak fleksibel, dan model RUP yang paling fleksibel. Dari segi jaminan keberhasilan, model *waterfall* kurang dijamin keberhasilannya, sedangkan model spiral dan RUP jaminan keberhasilannya tinggi. Dari segi pengujian, model *waterfall* lambat dalam melakukan pengujian, sedangkan untuk model spiral dan RUP cepat dalam melakukan pengujian bahkan dilakukan di akhir setiap iterasi untuk model RUP[9].

### 2.2. Dasar Teori

Untuk mendasari dari penelitian ini digunakan beberapa dasar teori adalah sebagai berikut.

#### 2.2.1. Metode Rational Unified Process (RUP)

Seperti telah diuraikan sebelumnya bahwa metode perangkat lunak yang dikembangkan dilakukan secara interative dan incremental yang menekankan pada arsitektur. Untuk menggambarkan fase yang digunakan dalam metode ini seperti terlihat pada Gambar 1.



Gambar 1. Arsitektur metode RUP

RUP memiliki empat tahap yaitu *inception*, *elaboration*, *contruction*, dan *transition*[4]. Berikut penjelasan mengenai empat tahap pengembangan RUP.

#### a. *Inception*

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan (*business modeling*), mendefinisikan kebutuhan akan sistem yang

- akan dibuat (*requirement*) serta analisis dan desain.
- b. *Elaboration*  
Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem (*prototype*).
- c. *Construction*  
Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program.
- d. *Transition*  
Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*.

**2.2.2. ISO 25010**

ISO 25010 adalah standar kualitas perangkat lunak yang baru, sebagai pengganti ISO 9126. Pada penelitian ini diambil 2 karakteristik yaitu *functional suitability*, dan *usability* adalah sebagai berikut [5]:

- a. *Functional Suitability*  
Pengujian *functional suitability* menggunakan metode *blackbox testing*. Hasil pengujiannya dihitung dengan rumus:

$$X = P/I$$

Keterangan:

P = Jumlah fitur yang dirancang

I = Jumlah fitur yang berhasil diimplementasikan

Dalam pengujian, produk dikatakan baik dalam *functional suitability* jika nilai X mendekati 1.

- b. *Functional Usability*  
Pada pengujian *usability*, digunakan kuesioner dengan rumus sebagai berikut:  
Berikut rumus perhitungan skor pengujian *usability*:

$$Skortotal = (JSS \times 5) + (JS \times 4) + (JN \times 3) + (JTS \times 2) + (JSTS \times 1)$$

Keterangan:

JSS = jumlah responden menjawab Sangat Setuju

JS = jumlah responden menjawab Setuju

JN = jumlah responden menjawab Netral

JTS = jumlah responden menjawab Tidak Setuju

JSTS = jumlah responden menjawab Sangat Tidak Setuju

Kemudian mencari persentase skor untuk mendapatkan kriteria interpretasi hasil pengujian *usability* dengan rumus:

$$Pskor = Skor\ total \times r \times 5 \times 100\%$$

Keterangan:

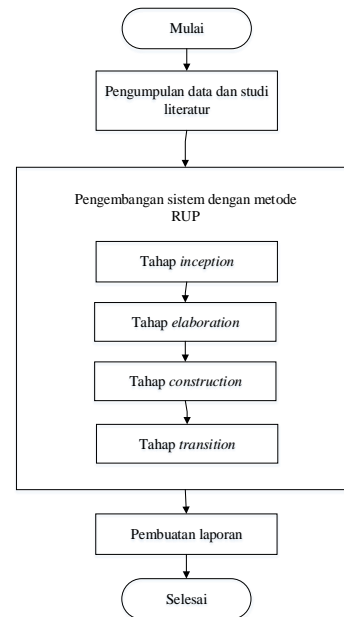
Skor total = skor total hasil jawaban responden

i = jumlah pertanyaan

r = jumlah responden

**3. METODE PENELITIAN**

Dalam pengembangan sistem menggunakan beberapa tahapan yang mengikuti metode RUP seperti yang terlihat pada Gambar 2.



Gambar 2. Tahapan pengembangan sistem pada penelitian.

Penjelasan dari tahapan pengembangan pada Gambar 2 adalah sebagai berikut.

**3.1. Pengumpulan Data dan Studi Literatur**

Pengumpulan data dilaksanakan dengan dua cara yaitu observasi dan wawancara.

**3.1.1. Observasi**

Pada tahap ini, penulis mengamati secara langsung bagaimana proses pendaftaran *medicalcheckup*, pengelolaan data pasien, dan pembuatan hasil pemeriksaan oleh pegawai Citra Medical Centre.

**3.1.2. Wawancara**

Berdasarkan wawancara diketahui masalah yang ada yaitu penggunaan *server* lokal sehingga pengelolaan data pasien hanya bisa dikerjakan pada jam kantor. Sehingga penulis menawarkan untuk melaksanakan penelitian pengembangan sebuah

sistem informasi yang dapat membantu terkait pendaftaran *medical checkup* secara *online* dan pengelolaan data yang dapat dikerjakan di mana saja dan kapan saja.

**3.1.3. Studi literatur**

Penulis melakukan studi literatur terhadap artikel, jurnal, dan skripsi yang sesuai dengan permasalahan yang ada sebagai bahan rujukan dan memperkuat argumentasi dari informasi diperoleh.

**3.2. Pengembangan Sistem dengan Metode RUP**

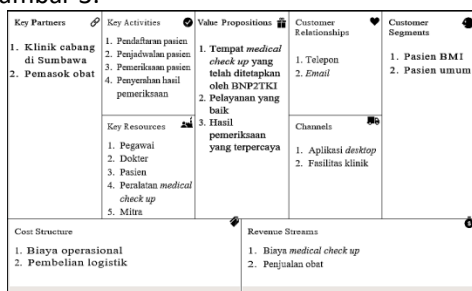
Pada pengembangan sistem dengan metode RUP, tahapan yang digunakan adalah tahap *inception*, *elaboration*, *construction*, dan *transition*.

**3.2.1. Tahap Inception**

Pada tahap *inception* terdapat beberapa proses RUP yang dilakukan diantaranya membuat *business modelling*, *requirements*, *analysis and design*, dan *test*.

a. *Business modelling*

*Business modelling* yang digunakan adalah *Business modelling canvas* (BMC) seperti Gambar 3.



Gambar 3. Perancangan *business modelling canvas* penelitian.

Dari BMC pada Gambar 3 yang dibutuhkan pada penelitian adalah *Key Activities*, *Customer Relationship*, *Customer Segments* dan *Channels*.

b. *Requirements*

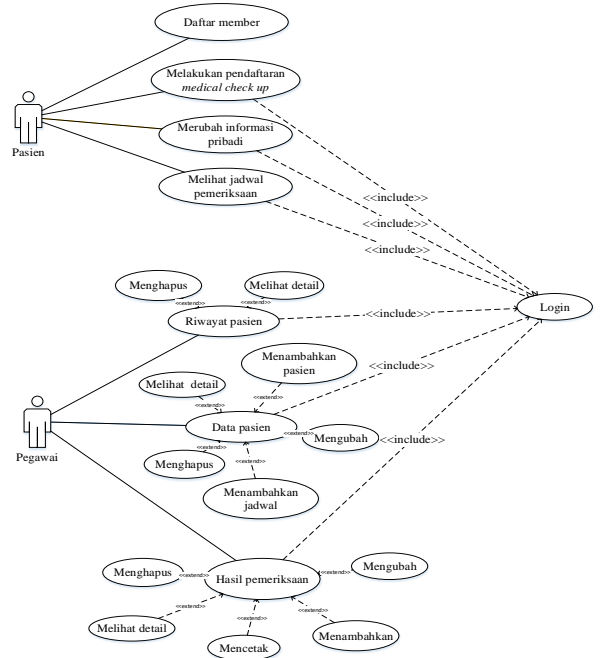
Kebutuhan fungsional sistem seperti yang terlihat pada Tabel I [10].

TABEL I. KEBUTUHAN FUNSIONAL SISTEM

No	Kebutuhan fungsional	Aktor
1	Melakukan pendaftaran untuk menjadi member	Pasien
2	Melakukan pendaftaran <i>medical check up</i>	Pasien
3	Melihat jadwal pemeriksaan pribadi	Pasien
4	Mengubah data pribadi	Pasien
5	Mengelola data pasien (melihat, menambahkan, mengubah, dan menghapus)	Pegawai

6	Mengelolahasil pemeriksaan (melihat, menambah, mengubah, dan menghapus)	Pegawai
7	Mengelolariwayat pasien (melihat dan menghapus)	Pegawai

c. *Analysis and design*



Gambar 4. *Usecase* diagram sistem.

Analisis pada tahap *inception* dijelaskan dengan menggunakan *usecase diagram* seperti pada Gambar 4. Terdapat dua aktor yang berperan yaitu pasien dan pegawai. Masing-masing aktor tersebut memiliki hak akses yang berbeda. Pasien memiliki 4 hak akses sedangkan pegawai memiliki 3 hak akses.

d. *Test*

Pada tahap ini, pengujian bertujuan untuk memvalidasi kebutuhan sistem dan *usecase* yang telah dirancang. Pengujian melibatkan 2 orang penguji yang berperan sebagai pengguna sistem yaitu pegawai dan pimpinan klinik. Pengujian dilakukan dengan mencocokkan *usecase* berdasarkan kebutuhan fungsional sistem untuk pasien dan pegawai. Proses validasi dimulai dengan peneliti memberikan tabel isian yang kemudian diisi dan di validasi oleh masing-masing penguji pada klinik. Hasil validasi dapat dilihat pada Tabel II.

TABEL II. VALIDASI KEBUTUHAN SISTEM DAN USE CASE

N	Kebutuhan fungsional sistem	Use case
---	-----------------------------	----------

o		Pasien	Pegawai
1	Daftar menjadi member	✓	
2	Daftar Medical check up	✓	
3	Melihat jadwal pemeriksaan pribadi	✓	
4	Mengubah data pribadi	✓	
5	Mengelola data pasien (melihat, menambahkan pasien, mengubah, menambahkan jadwal, dan menghapus)		✓
6	Mengelola riwayat pasien (melihat dan menghapus)		✓
7	Mengolah hasil pemeriksaan (melihat, menambahkan, mengubah, mencetak, dan menghapus)		✓

Dari validasi yang dilakukan diperoleh hasil bahwa kebutuhan dan *usecase* sistem yang dirancang sesuai dengan kebutuhan pengguna. Dengan demikian proses dapat dilanjutkan menuju tahap selanjutnya yaitu tahap *elaboration*.

**3.2.2. Tahap Elaboration**

Tahap *elaboration* merupakan tahap setelah *inception*. Pada tahap ini terdapat proses RUP yaitu *requirements, analysis and design, implementation, dan test*.

**a. Requirements**

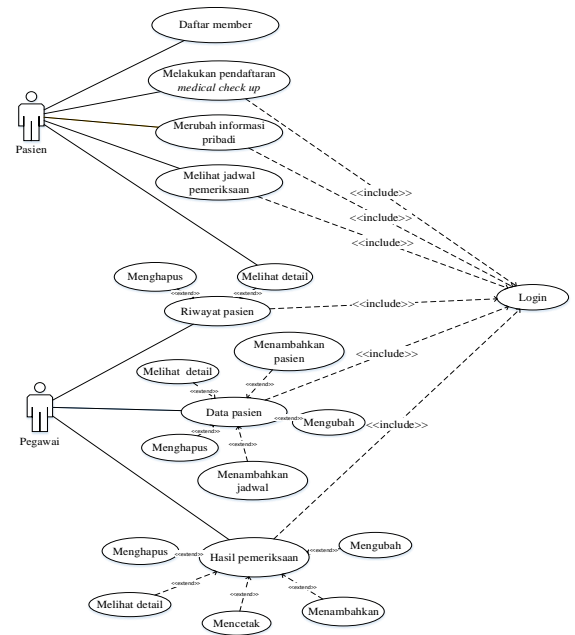
Terdapat penambahan kebutuhan fungsional sistem dari tahap sebelumnya pada Tabel yaitu pasien dapat melihat riwayat pemeriksaan pribadinya.

**b. Analysis and design**

Pada *analysis and design* tahap *elaboration* terdapat *usecase diagram, activity diagram, class diagram, sequencediagram* dan ERD.

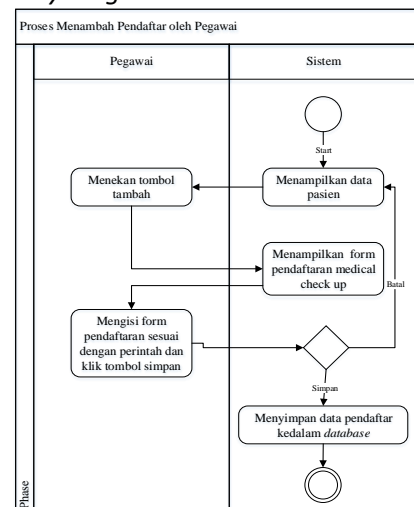
**1. Usecase diagram**

*Usecase diagram* yang telah dirancang pada tahap sebelumnya yaitu tahap *inception* seperti pada Gambar 4, dilakukan perubahan pada tahap *elaboration* karena adanya penambahan kebutuhan sistem yaitu sebuah menu untuk memungkinkan pasien melihat riwayat pemeriksaan pribadinya sehingga pasien dapat melihat detail hasil pemeriksaan yang telah dilakukan. *Usecase* setelah dilakukan perubahan dapat dilihat pada Gambar 5.



Gambar 5. *Usecase diagram* sistem setelah dilakukan perubahan.

**2. Activity diagram**

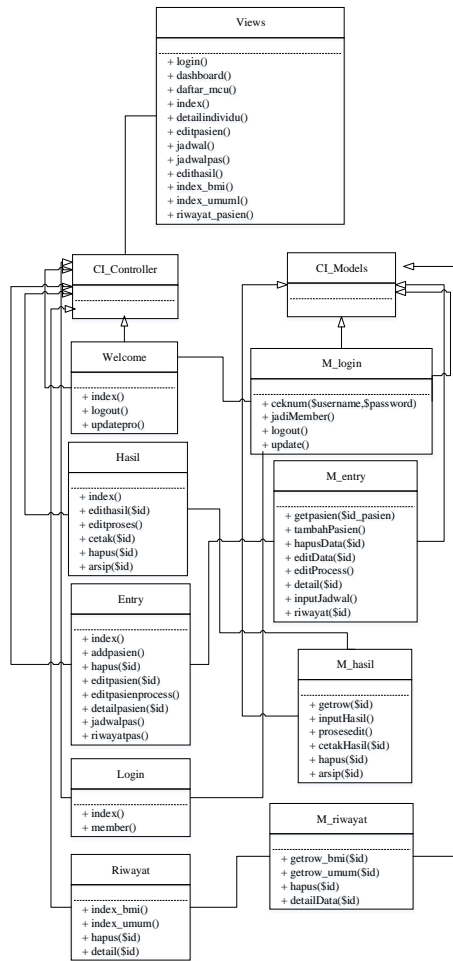


Gambar 6. *Activity diagram* pendaftaran pasien oleh pegawai

Gambar 6 merupakan *activity diagram* pegawai untuk menambahkan pasien dengan mengisi *form medical checkup*.

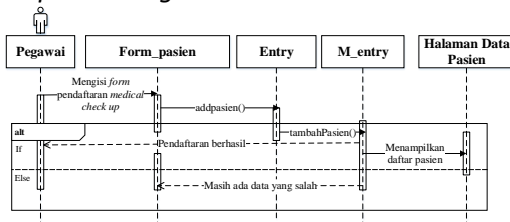
**3. Class diagram**

Keterhubungan antar kelas yang terdapat pada sistem dapat dilihat pada Gambar 7. Sistem yang akan dibangun menggunakan pola arsitektur *Models Views Controllers (MVC)*, sehingga dilakukan perancangan kelas – kelas yang ada pada sistem yang terdiri dari kelas *Controllers, Views* dan *Models*.



Gambar 7. Class diagram sistem

4. Sequence diagram

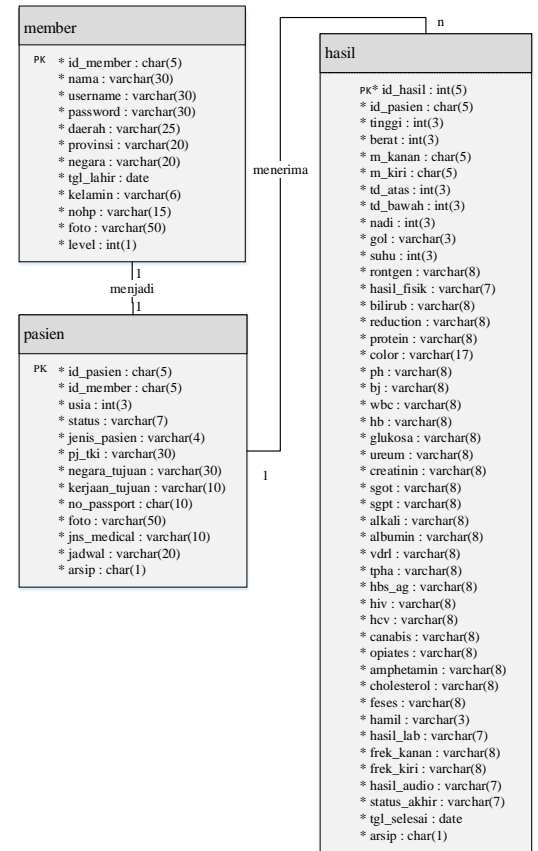


Gambar 8. Sequence diagram untuk proses penambahan pasien oleh pegawai. Pada Gambar 8, proses menambahkan pasien dilakukan dengan mengisi form pasien yang disediakan, selanjutnya akan di akses fungsi addpasien() pada controllerEntry dan jika sesuai maka data tersebut akan tersimpan dalam database dengan fungsi tambahPasien() pada model M\_entry.

5. Entity Relationship Diagram (ERD)

ERD sistem yang akan dibangun dapat dilihat pada Gambar 9, dimana terdapat

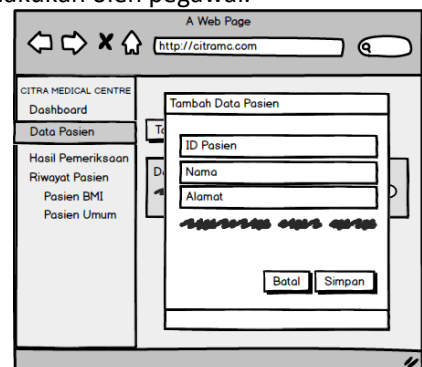
tabel member, tabel pasien, dan tabel hasil.



Gambar 9. ERD sistem

6. Implementation

Implementasi yang dilakukan pada tahap elaboration adalah perancangan interface sistem seperti yang terlihat pada Gambar 10 yang merupakan rancangan untuk proses penambahan pasien baru yang dilakukan oleh pegawai.



Gambar 10. Desain interface halaman tambah pasien oleh pegawai.

7. Test

Pada tahap elaboration, pengujian dibutuhkan untuk menguji fitur yang telah dirancang apakah sudah sesuai dengan

yang diinginkan pengguna atau belum. Pengujian dilakukan oleh 2 orang penguji yaitu pegawai dan pimpinan klinik. Proses validasi fitur dimulai dengan peneliti menampilkan rancangan *interface* sistem, setelah itu responden memberikan pernyataan setuju atau tidak setuju pada tabel isian yang telah disediakan. Hasil validasi dapat dilihat pada Tabel III.

TABEL III. VALIDASIFITURSISTEM

No	Pengguna	Fitur	Validasi	
			Setuju	Tidak
1	Pasien	Daftar member	✓	
		Melakukan pendaftaran Medical check up	✓	
		Mengubah informasi pribadi	✓	
		Melihat jadwal pemeriksaan	✓	
		Melihat riwayat pemeriksaan	✓	
2	Admin	Melihat detail riwayat pasien	✓	
		Menghapus riwayat pasien	✓	
		Melihat detail data pasien	✓	
		Menambahkan pasien baru	✓	
		Mengubah data pasien	✓	
		Menambahkan jadwal pemeriksaan	✓	
		Menghapus data pasien	✓	
		Melihat detail hasil pemeriksaan	✓	
		Menambahkan hasil	✓	

		pemeriksaan		
		Mencetak hasil pemeriksaan	✓	
		Mengarsipkan hasil pemeriksaan	✓	
		Menghapus hasil pemeriksaan	✓	

Dari pengujian yang dilakukan oleh 2 orang penguji, dapat disimpulkan bahwa rancangan *interface* sistem telah sesuai dengan keinginan pengguna. Maka proses dapat dilanjutkan menuju tahap *construction* atau pengembangan sistem.

#### 4. HASIL DAN PEMBAHASAN

Penerapan metode RUP pada pengembangan sistem dari keempat tahapan RUP adalah sebagai berikut.

##### 4.1. Tahap Inception

Pada tahap *inception* dilakukan perancangan *business modeling canvas*, pendefinisian kebutuhan sistem, perancangan *usecase diagram*, dan pengujian rancangan.

##### 4.2. Tahap Elaboration

Pada tahap *elaboration* dilakukan penambahan kebutuhan sistem dan *usecase diagram*, perancangan *activity diagram*, *class diagram*, *sequence diagram*, dan ERD, serta pengujian validasi fitur yang telah dirancang.

##### 4.3. Tahap Construction

Pada tahap *construction* terdapat dua proses RUP yang dilakukan yaitu *implementation* dan *test*, karena proses *implementation* sudah selesai pada tahap *elaboration*.

##### 4.3.1. Implementation

Implementasi pada tahap ini adalah pembuatan *database*, pengembangan sistem, dan *interface* sistem.

###### a. Database

Pada pengembangan *database* sistem informasi *medical checkup* Klinik Citra, terdapat 3 tabel yang digunakan yaitu tabel member, tabel pasien dan tabel hasil.

1. Tabel member

#	Name	Type	Collation	Attributes	Null	Default
1	id_member	char(5)	latin1_swedish_ci	No	None	
2	nama	varchar(30)	latin1_swedish_ci	No	None	
3	username	varchar(30)	latin1_swedish_ci	No	None	
4	password	varchar(30)	latin1_swedish_ci	No	None	
5	daerah	varchar(25)	latin1_swedish_ci	No	None	
6	provinsi	varchar(20)	latin1_swedish_ci	No	None	
7	negara	varchar(20)	latin1_swedish_ci	No	None	
8	tgl_lahir	date		No	None	
9	kelamin	varchar(6)	latin1_swedish_ci	No	None	
10	nohp	varchar(15)	latin1_swedish_ci	No	None	
11	foto	varchar(50)	latin1_swedish_ci	No	None	
12	level	int(1)		No	None	

Gambar 11. Tabel member

Pada tabel member terdapat 12 atribut seperti yang terlihat pada Gambar 11. Tabel member digunakan untuk menampung data member yang akan digunakan untuk melakukan pendaftaran *medicalcheckup*.

2. Tabel pasien

Pada tabel pasien terdapat 13 atribut seperti yang terlihat pada Gambar 12. Tabel pasien ini digunakan untuk menampung data pasien yang telah melakukan pendaftaran *medicalcheckup*.

#	Name	Type	Collation	Attributes	Null	Default
1	id_pasien	char(5)	latin1_swedish_ci	No	None	
2	id_member	char(5)	latin1_swedish_ci	No	None	
3	usia	int(3)		No	None	
4	status	varchar(7)	latin1_swedish_ci	No	None	
5	jenis_pasien	varchar(4)	latin1_swedish_ci	No	None	
6	pi_tki	varchar(30)	latin1_swedish_ci	No	None	
7	negara_tujuan	varchar(30)	latin1_swedish_ci	No	None	
8	kerjaan_tujuan	varchar(10)	latin1_swedish_ci	No	None	
9	no_passport	char(10)	latin1_swedish_ci	No	None	
10	foto	varchar(50)	latin1_swedish_ci	No	None	
11	jns_medical	varchar(10)	latin1_swedish_ci	No	None	
12	jadwal	date		No	None	
13	arsip	char(1)	latin1_swedish_ci	No	None	

Gambar 12. Tabel pasien.

3. Tabel hasil

Pada tabel hasil terdapat 46 atribut seperti yang terlihat pada Gambar 13. Tabel hasil ini digunakan untuk menampung data hasil pemeriksaan pasien.

#	Name	Type	Collation	Attributes	Null	Default
1	id_hasil	int(5)		No	None	
2	id_pasien	char(5)	latin1_swedish_ci	No	None	
3	tinggi	int(3)		No	None	
4	berat	int(3)		No	None	
5	m_kanan	char(5)	latin1_swedish_ci	No	None	
6	m_kiri	char(5)	latin1_swedish_ci	No	None	
7	td_atas	int(3)		No	None	
8	td_bawah	int(3)		No	None	
9	nadi	int(3)		No	None	
10	gol	varchar(3)	latin1_swedish_ci	No	None	
11	suhu	int(3)		No	None	
12	rontgen	varchar(8)	latin1_swedish_ci	No	None	
13	hasil_fisik	varchar(7)	latin1_swedish_ci	No	None	
14	bilirub	varchar(8)	latin1_swedish_ci	No	None	
15	reduction	varchar(8)	latin1_swedish_ci	No	None	
16	protein	varchar(8)	latin1_swedish_ci	No	None	
17	color	varchar(17)	latin1_swedish_ci	No	None	
18	ph	varchar(8)	latin1_swedish_ci	No	None	
19	bj	varchar(8)	latin1_swedish_ci	No	None	
20	wbc	varchar(8)	latin1_swedish_ci	No	None	
21	hb	varchar(8)	latin1_swedish_ci	No	None	
22	glukosa	varchar(8)	latin1_swedish_ci	No	None	
23	ureum	varchar(8)	latin1_swedish_ci	No	None	
24	creatinin	varchar(8)	latin1_swedish_ci	No	None	
25	sgot	varchar(8)	latin1_swedish_ci	No	None	
26	sgpt	varchar(8)	latin1_swedish_ci	No	None	
27	alkali	varchar(8)	latin1_swedish_ci	No	None	
28	albumin	varchar(8)	latin1_swedish_ci	No	None	
29	vdri	varchar(8)	latin1_swedish_ci	No	None	
30	tpha	varchar(8)	latin1_swedish_ci	No	None	
31	hbs_ag	varchar(8)	latin1_swedish_ci	No	None	
32	hiv	varchar(8)	latin1_swedish_ci	No	None	
33	hcv	varchar(8)	latin1_swedish_ci	No	None	
34	canabis	varchar(8)	latin1_swedish_ci	No	None	
35	opiates	varchar(8)	latin1_swedish_ci	No	None	
36	amphetamine	varchar(8)	latin1_swedish_ci	No	None	
37	cholesterol	varchar(8)	latin1_swedish_ci	No	None	
38	feses	varchar(8)	latin1_swedish_ci	No	None	
39	hamil	varchar(3)	latin1_swedish_ci	No	None	
40	hasil_lab	varchar(7)	latin1_swedish_ci	No	None	
41	frek_kanan	varchar(8)	latin1_swedish_ci	No	None	
42	frek_kiri	varchar(8)	latin1_swedish_ci	No	None	
43	hasil_audio	varchar(7)	latin1_swedish_ci	No	None	
44	status_akhir	varchar(7)	latin1_swedish_ci	No	None	
45	tgl_selesai	date		No	None	
46	arsip	char(1)	latin1_swedish_ci	Yes	NULL	

Gambar 13. Tabel hasil

b. Project

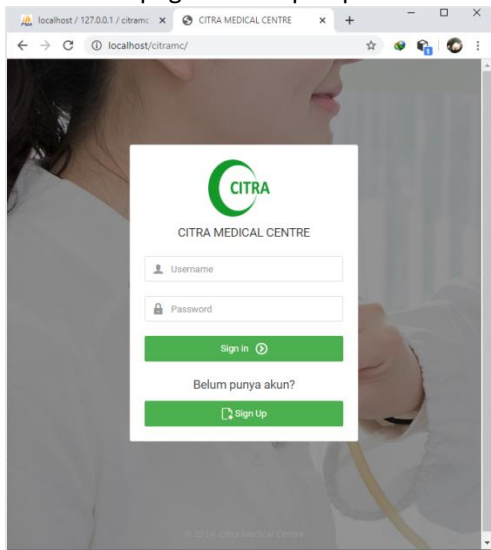
Pengembangan sistem dengan *frameworkcodeigniter* menggunakan konsep MVC yaitu *Models*, *Views* dan *Controllers*. *Models* berisi fungsi yang dapat digunakan untuk mengelola *database*, *Views* untuk mengatur tampilan, dan *Controllers* berisi fungsi untuk memproses data dan merupakan bagian yang berfungsi sebagai penghubung antara *Models* dan *Views*.

c. Interface

Halaman *log in* pada Gambar 14 digunakan untuk dapat masuk ke sistem dengan memasukkan *username* dan *password* jika sudah memiliki akun, Sedangkan jika belum maka harus melakukan pendaftaran untuk



menjadi member dengan memilih tombol *Signup*. Halaman *log in* ini bisa digunakan bersama oleh pegawai maupun pasien.



Gambar 14. Halaman awal sistem

d. *Test*

Pada tahap *construction* dilakukan pengujian untuk fitur sistem yang telah dibangun dan kesesuaian sistem dengan kebutuhan pengguna. Pengujian dilakukan dengan menggunakan *blackbox testing* untuk pengujian *functional suitability*.

1. *Functional Completeness*

Pada pengujian *functional completeness* terdapat 20 fitur yang di uji untuk admin dan juga pasien yaitu seperti yang terlihat Tabel IV.

TABEL IV. PENGUJIAN FUNGSI COMPLETENESS

No	Fungsi	Hasil yang diharapkan	Hasil pengujian	
			Sesuai	Tidak
<b>Admin</b>				
1	Log in sebagai admin	Fungsi log in sebagai admin sudah berjalan dengan benar	✓	
2	Menambahkan data pasien	Fungsi tambah pasien sudah berjalan dengan benar	✓	
3	Melihat detail data pasien	Fungsi lihat detail data pasien sudah berjalan dengan benar	✓	
4	Mengubah data pasien	Fungsi ubah data pasien sudah	✓	

		berjalan dengan benar		
5	Menambahkan jadwal pemeriksaan	Fungsi tambah jadwal pasien sudah berjalan dengan benar	✓	
6	Menghapus data pasien	Fungsi hapus data pasien sudah berjalan dengan benar	✓	
7	Melihat detail hasil pemeriksaan pasien	Fungsi lihat detail hasil pemeriksaan pasien sudah berjalan dengan benar	✓	
8	Menambahkan hasil pemeriksaan pasien	Fungsi tambah hasil pemeriksaan pasien sudah berjalan dengan benar	✓	
9	Mencetak hasil pemeriksaan pasien	Fungsi cetak hasil pemeriksaan pasien sudah berjalan dengan benar	✓	
10	Mengarsipkan hasil pemeriksaan pasien	Fungsi arsip hasil pemeriksaan pasien sudah berjalan dengan benar	✓	
11	Menghapus hasil pemeriksaan pasien	Fungsi hapus hasil pemeriksaan pasien sudah berjalan dengan benar	✓	
12	Melihat detail riwayat pasien	Fungsi lihat detail riwayat pasien sudah berjalan dengan benar	✓	
13	Menghapus riwayat pasien	Fungsi hapus riwayat pasien sudah berjalan dengan benar	✓	
14	Log out	Fungsi log out sudah berjalan dengan benar	✓	
<b>Pasien</b>				
15	Melakukan	Fungsi daftar member	✓	

	pendaftaran sebagai member	sudah berjalan dengan benar		
16	Log in sebagai pasien	Fungsi log in sebagai pasien sudah berjalan dengan benar	✓	
17	Melakukan pendaftaran <i>medical checkup</i>	Fungsi daftar <i>medical check up</i> sudah berjalan dengan benar	✓	
18	Melihat jadwal pemeriksaan pribadi	Fungsi lihat jadwal pemeriksaan pribadi sudah berjalan dengan benar	✓	
19	Melihat riwayat pemeriksaan pribadi	Fungsi lihat riwayat pemeriksaan pribadi sudah berjalan dengan benar	✓	
20	Mengubah data pribadi	Fungsi ubah data pribadi sudah berjalan dengan benar	✓	

Berdasarkan pengujian dengan Tabel IV yang dilakukan oleh 2 orang pengguna diperoleh hasil seperti pada Tabel V.

TABEL V. HASIL PENGUJIAN FUNGSI COMPLETENESS

Nomor Fungsi	Sesuai	Tidak
1	2	0
2	2	0
3	2	0
4	2	0
5	2	0
6	2	0
7	2	0
8	2	0
9	2	0
10	2	0
11	2	0
12	2	0
13	2	0
14	2	0
15	2	0
16	2	0
17	2	0
18	2	0
19	2	0

20	2	0
<b>Total</b>	<b>40</b>	<b>0</b>

$P = \text{Jumlah pernyataan} \times \text{jumlah penguji} = 40$

$I = \text{Jumlah nilai} \times \text{jumlah penguji} = 40$

Sehingga  $X = \frac{40}{40} = 1$

Pada pengujian yang diperoleh hasil sama dengan 1, jadi dapat disimpulkan bahwa semua fitur sudah berfungsi dengan baik

2. *Functional Correctness*

Pada pengujian *functional correctness* terdapat 5 fitur yang di uji untuk admin dan juga pasien yaitu seperti yang terlihat Tabel VI.

TABEL VI. PENGUJIAN FUNGSI CORRECTNESS.

No	Fungsi	Hasil yang diharapkan	Hasil pengujian	
			Sesuai	Tidak
<b>Umum</b>				
1	Identifikasi <i>username dan password</i>	Proses log in ke sistem sesuai dengan jenis <i>users</i> sudah berjalan dengan benar	✓	
2	Perhitungan jumlah data pasien pada <i>dashboard</i>	Proses menampilkan grafik jumlah data pasien sesuai dengan jenis kelamin dan jenis pasien pada <i>dashboard</i> sistem sudah berjalan dengan benar	✓	
<b>Admin</b>				
3	Menampilkan data pasien berdasarkan pencarian	Proses pencarian pada data pasien berdasarkan kata kunci tertentu sudah berjalan dengan benar	✓	
4	Menampilkan data hasil pemeriksaan berdasarkan kata kunci	Proses pencarian pada hasil pemeriksaan pasien berdasarkan kata kunci tertentu	✓	

	pencarian	sudah berjalan dengan benar		
5	Menampilkan riwayat pasien berdasarkan pencarian	Proses pencarian pada riwayat pasien berdasarkan kata kunci tertentu sudah berjalan dengan benar	✓	

Berdasarkan pengujian dari Tabel VI diperoleh hasil seperti pada Tabel VII.

TABEL VII. HASIL PENGUJIAN FUNGSI CORRECTNESS.

Nomor Fungsi	Sesuai	Tidak
1	2	0
2	2	0
3	2	0
4	2	0
5	2	0
<b>Total</b>	<b>10</b>	<b>0</b>

P = Jumlah pernyataan x jumlah pengujian = 10

I = Jumlah nilai x jumlah pengujian = 10

Sehingga  $X = \frac{10}{10} = 1$

Pada pengujian yang dilakukan diperoleh hasil sama dengan 1, jadi dapat disimpulkan bahwa semua fitur sudah berjalan dengan baik

3. *Functional Appropriateness*

Pada pengujian *functional appropriateness* terdapat 1 fitur yang di uji untuk admin yaitu seperti yang terlihat pada Tabel VIII.

TABEL VIII. PENGUJIAN FUNGSI APPROPRIATENESS

No	Fungsi	Hasil yang diharapkan	Hasil pengujian	
			Sesuai	Tidak
<b>Admin</b>				
1	Mencetak hasil pemeriksaan pasien	Fungsi mencetak hasil pemeriksaan pasien yang bersangkutan menghasilkan <i>output</i> yang benar	✓	

Berdasarkan pengujian dengan Tabel VIII yang dilakukan oleh 2 orang pengguna diperoleh hasil seperti pada Tabel IX.

TABEL IX. HASIL PENGUJIAN FUNGSI APPROPRIATENESS

TENESS		
Nomor Fungsi	Sesuai	Tidak
1	2	0
<b>Total</b>	<b>2</b>	<b>0</b>

P = Jumlah pernyataan x jumlah pengujian = 2

I = Jumlah nilai x jumlah pengujian = 2

Sehingga  $X = \frac{2}{2} = 1$

Pada pengujian yang dilakukan terhadap ketiga *functional* sistem diperoleh hasil sama dengan 1, jadi dapat disimpulkan bahwa semua fitur sistem sudah berjalan dengan baik, sehingga proses dapat dilanjutkan menuju tahap *transition*.

4.4. Tahap *Transition*

Tahap *transition* adalah tahap selanjutnya dari tahap *construction*. Pada tahap ini akan dilakukan pengujian *usability* sistem untuk mengetahui kesesuaian sistem dengan kebutuhan pengguna. Proses pengujian dilakukan oleh dua pihak, yaitu dari pihak klinik sebagai admin, dan dari pihak masyarakat umum sebagai pasien.

4.4.1. Admin

Pengujian untuk pihak klinik dilakukan oleh 2 orang yaitu pegawai dan pimpinan klinik. Pengujian menggunakan kuesioner dengan 21 butir pernyataan yang terbagi menjadi 4 kriteria yaitu *usefulness*, *ease of use*, *ease of learning* dan *satisfaction*[5].

Dari pengujian yang dilakukan oleh 2 orang pengguna diperoleh hasil seperti yang terlihat pada Gambar 15.

Nomor Responden	Pernyataan																					Total Nilai	Nilai Maksimal
	Usefulness							Ease of use							Ease of learning				Satisfaction				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21		
1	4	4	5	4	4	4	5	5	4	4	5	4	4	4	4	4	4	4	5	4	5	90	105
2	4	4	4	4	5	4	5	4	4	4	4	4	4	4	4	5	5	4	5	4	4	89	105
Total Nilai																					179	210	

Gambar 15. Hasil pengujian *usability* sistem oleh pihak klinik.

Berdasarkan hasil pengujian pada Gambar 15, nilai yang diperoleh sebesar 179. Hasil pengujian *usability* dihitung menggunakan rumus sebagai berikut.

$$P \text{ Skor} = \frac{179}{210} \times 100\% = 85,23\%$$

Dari hasil perhitungan diperoleh persentase pengujian *usability* sebesar 85,23%, sehingga sistem informasi *medical checkup* dapat dinyatakan "Sangat Layak" dan memenuhi standar *usability*.

**4.4.2. Pasien**

Pengujian dari pihak pasien dilakukan oleh 30 orang yang diambil secara acak dari berbagai bidang pekerjaan. Pengujian menggunakan kuesioner dengan 8 butir pernyataan yang terbagi menjadi 4 kriteria yaitu *usefulness*, *ease of use*, *ease of learning* dan *satisfaction*[5].

Dari pengujian yang dilakukan oleh 30 orang pengguna diperoleh hasil seperti yang terlihat pada Tabel X.

TABEL X. HASIL PENGUJIAN OLEH PASIEN

No	Pernyataan								Total Nilai	Nilai Maksimal	
	Usefulness	Ease of use			Ease of learning		Satisfaction				
		1	2	3	4	5	6	7			8
1	4	5	4	4	4	5	5	4	35	40	
2	4	4	5	5	4	4	4	5	35	40	
3	5	4	4	4	4	5	5	4	35	40	
4	5	4	4	5	5	4	5	5	37	40	
5	4	4	4	3	4	4	5	4	32	40	
6	4	4	4	5	4	5	5	4	35	40	
7	4	4	4	5	4	5	5	5	36	40	
8	4	4	4	4	3	5	5	5	34	40	
9	4	4	5	4	4	4	5	5	35	40	
10	4	4	4	4	4	4	4	4	32	40	
11	5	4	4	4	4	5	5	5	36	40	
12	4	4	4	4	4	4	5	4	33	40	
13	5	4	4	5	4	5	5	4	36	40	
14	4	4	5	4	4	4	4	5	34	40	
15	4	4	5	5	4	4	5	5	36	40	
16	5	4	4	4	5	4	5	4	35	40	
17	4	4	4	5	5	4	5	4	35	40	
18	4	4	5	4	4	4	5	4	34	40	
19	4	4	4	5	5	4	5	4	35	40	
20	4	4	4	4	4	5	5	5	35	40	
21	4	4	4	4	4	5	5	4	34	40	
22	5	4	4	4	4	4	5	5	35	40	
23	4	4	4	5	4	4	5	4	34	40	
24	5	4	4	4	4	4	4	4	33	40	
25	5	4	4	4	4	5	5	5	36	40	
26	4	4	4	5	4	5	5	4	35	40	
27	4	4	4	4	5	5	5	4	35	40	
28	4	4	5	5	4	5	4	4	35	40	
29	5	4	4	4	5	4	5	4	35	40	
30	4	4	4	4	3	4	5	4	32	40	
Total Nilai									1039	1200	

Berdasarkan hasil pengujian pada Tabel X, dapat diperoleh total nilai sebesar 1039. Hasil pengujian *usability* dapat diperoleh dengan menggunakan rumus sebagai berikut.

$$P \text{ Skor} = \frac{1}{1} \times 100\% = 87\%$$

Persentase pengujian *usability* diperoleh sebesar 87%, sehingga dapat disimpulkan bahwa sistem informasi *medical checkup* memenuhi standar *usability* dan dinyatakan "Sangat Layak".

**5. KESIMPULAN**

Berdasarkan analisis dan pembahasan, dapat diambil kesimpulan sebagai berikut:

1. Pengujian RUP dilakukan pada setiap tahapan. Pada tahap *inception* dilakukan pengujian untuk validasi kebutuhan dan *usecase diagram* sistem yang dirancang. Tahap *elaboration* dilakukan pengujian untuk validasi fitur sistem. Untuk tahap *construction* dilakukan pengujian *suitability* dengan menggunakan *blackbox testing*, dan untuk tahap *transition* dilakukan pengujian *usability* sistem dengan menggunakan kuesioner.
2. Ketika terdapat kebutuhan sistem pada tahap *inception* yang masih perlu dilakukan perubahan atau penambahan, maka proses perubahannya bisa dilakukan pada tahap awal *elaboration*.
3. Pengujian terhadap sistem dilakukan dengan menggunakan *blackbox testing* dan kuesioner. Dari *blackbox testing* untuk pengujian fitur sistem diperoleh hasil sebesar 1 yang artinya bahwa semua fitur sistem sudah berjalan dengan benar. Pada pengujian *usability* diperoleh hasil sebesar 86,12% dan jika interpretasikan menjadi "Sangat Layak", sehingga metode RUP merupakan metode yang baik untuk pengembangan sistem.

**DAFTAR PUSTAKA**

- [1] P. Kroll and P. Kruchten, *The Rational Unified Process Made Easy*. 2003.
- [2] S. A. Muhamad, "Implementasi dan Pengembangan Aplikasi Kerjasama Kesehatan Medical Check Up pada Rumah Sakit Citra Medika Cibitung-Bekasi-Jawa Barat," Universitas Islam Negeri Syarif Hidayatullah Jakarta, 2010.
- [3] "Pantau PJTKI - Daftar Tempat Pemeriksaan Kesehatan (Medical Check Up) untuk BMI." [Online]. Available: [http://pantaupjtki.buruhmigran.or.id/index.php/read/daftar-tempat-pemeriksaan-kesehatan-\(medical-check-up\)-untuk-bmi](http://pantaupjtki.buruhmigran.or.id/index.php/read/daftar-tempat-pemeriksaan-kesehatan-(medical-check-up)-untuk-bmi).
- [4] Beny, "Sistem Informasi Pendistribusian Suku Cadang Industri pada CV. Oilfindo Menggunakan Metode Rational Unified Process," Sekolah Tinggi Manajemen Informatika dan Komputer Palcomtech, 2019.
- [5] A. A. D. I. Nugroho, "Analisis dan Pengembangan Sistem Ujian Akhir Semester Berbasis Komputer di SMK Negeri 1 Magelang," Universitas Negeri Yogyakarta, 2018.
- [6] S. Young, "Penerapan Metode SDLC RUP (Rational Unified Proses) dalam Pembuatan Sistem Informasi E-kos Berbasis Web pada Kota Malang," Universitas Muhammadiyah Malang, 2018.

- [7] M. Zaminkar and M. R. Reshadinezhad, "A Comparison Between Two Software Engineering Processes, RUP And Waterfall Models," *Int. J. Eng. Res. Technol.*, vol. 2, no. 7, pp. 1348–1352, 2013.
- [8] A. Alshamrani and A. Bahattab, "A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model," *IJCSI Int. J. Comput. Sci. Issues*, vol. 12, no. 1, pp. 106–111, 2015.
- [9] F. Mubarok, H. Harliana, and I. Hadijah, "Perbandingan Antara Metode RUP dan Prototype Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web," *Creat. Inf. Technol. J.*, vol. 2, no. 2, p. 114, 2015.
- [10] T. F. Alimuddin Yasin dan MZ. Yumarlin, "Analisis Kebutuhan Sistem Informasi di LPK RJ-COMP Yogyakarta," no. January 2015, pp. 111–116, 2017.