

**APLIKASI ARSITEKCO BERBASIS *WEBSITE* UNTUK
MENGHUBUNGKAN MASYARAKAT DENGAN TENAGA
PROFESIONAL DI BIDANG ARSITEKTUR KONSULTAN DAN
KONTRAKTOR**

PROPOSAL TUGAS AKHIR



Oleh:

EKO PERMANA

NIM. 2003068

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI INDRAMAYU
JUNI 2022**

DAFTAR ISI

| | |
|--|----|
| DAFTAR ISI | ii |
| 1. Latar Belakang Masalah | 1 |
| 2. Rumusan Masalah | 3 |
| 3. Batasan Masalah | 3 |
| 4. Tujuan | 3 |
| 5. Manfaat | 4 |
| 6. Landasan Teori | 4 |
| 6.1 Scrum | 4 |
| 6.2 Unified Modeling Language (UML) | 4 |
| 6.3 Entity Relationship Diagram (ERD) | 8 |
| 7. Metode Pelaksanaan | 9 |
| 7.1 Teknik Pengumpulan Data | 9 |
| 7.2 Metode Pengembangan Perangkat Lunak | 9 |
| 8. Rencana Kegiatan | 10 |
| DAFTAR PUSTAKA | 11 |

1. Latar Belakang Masalah

Rumah yang indah dan nyaman untuk ditinggali adalah suatu pencapaian yang ingin didapatkan oleh setiap manusia. Manusia pada umumnya sering menempatkan pembangunan rumah sebagai tujuan dalam hidupnya. Membangun rumah impian terkadang memiliki banyak kendala salah satunya yaitu kesulitan dalam menentukan desain rumah sampai tahap konstruksi. Kesalahan dalam pembuatan rumah dapat berdampak pada penyesalan tersendiri. Oleh karena itu, pembuatan rumah harus dilakukan secara teliti dan matang agar penyesalan tersebut tidak ada. Tentunya hal ini dibutuhkan pengetahuan mengenai desain rumah yang dirancang oleh tenaga arsitek konsultan profesional dan tahap konstruksi yang dikerjakan oleh tenaga kontraktor dengan tim yang berpengalaman.

Tahapan dalam pembuatan rumah memiliki langkah yang terbilang rumit, seperti perencanaan, studi kelayakan, pemaparan, perancangan desain, pelelangan, pelaksanaan, pemeliharaan, dan persiapan pemakaian. Memahami tahapan tersebut akan membantu semua pihak yang terlibat sebelum memulai proyek pembangunan. Tanpa rancangan lengkap dan matang dapat menghambat atau menyebabkan kecelakaan. Tentu hal ini perlu dipahami untuk setiap individu yang ingin membangun sebuah rumah agar mendapatkan kenyamanan jangka panjang.

Pengetahuan masyarakat terhadap portofolio karya atau desain arsitektur terkadang masih minim. Sedangkan tenaga arsitek konsultan kesulitan dalam mencari pengguna jasa atau masyarakat yang ingin menggunakan jasanya. Sehingga dibutuhkan wadah yang terpusat untuk menghubungkan tenaga arsitek konsultan profesional dengan masyarakat atau calon pengguna jasa begitupun sebaliknya. Adanya wadah tersebut dapat menyelesaikan masalah yang dialami oleh keduanya.

Dalam dunia proyek konstruksi bangunan dibutuhkan desain dan rencana anggaran biaya yang sangat detail. Detail dalam konstruksi perlu ditampilkan sebagai persyaratan yang wajib dilakukan. Kesalahan dalam mempersiapkan detail tersebut sering sekali ditemukan sehingga pihak kontraktor bangunan mengalami kebingungan dalam mengerjakan sebuah proyek. Diperlukan informasi yang lengkap untuk memberikan gambaran terhadap calon pengguna jasa agar mempersiapkan segala hal yang dibutuhkan. Oleh karena itu, perlu adanya tempat atau wadah yang menyajikan informasi tahapan sebelum memulai konstruksi proyek bangunan.

Adapun masyarakat yang telah mempersiapkan semua persyaratan masih diharuskan mencari tenaga kontraktor profesional agar proyek bangunan miliknya dapat dikerjakan dengan baik. Hal itu dibutuhkan tempat atau media terpusat yang menyajikan berbagai profil kontraktor dan tim yang mendukungnya. Profil kontraktor dapat menarik minat calon pengguna jasa sehingga proses pencarian dapat berlangsung dengan mudah. Setelah tahapan pencarian dilakukan pengguna jasa dapat menyepakati ketentuan awal yang ditentukan oleh kontraktor dan ketentuan akhir sesuai dengan isi dari desain dan rencana anggaran biaya milik pengguna jasa. Dengan demikian, kesepakatan yang diperoleh menjadi awal dari sebuah konstruksi proyek bangunan.

Tempat atau wadah yang terpusat menjadi patokan utama dalam mempertemukan semua pihak yang saling berkaitan. Wadah tersebut memberikan banyak peluang terutama peluang kerja setiap sektor dalam bidang arsitek konsultan dan proyek konstruksi bangunan. Peluang kerja yang menjadi luas sehingga dapat menjadi solusi untuk mengurangi angka pengangguran saat ini. Peluang masyarakat dapat menemukan jasa tenaga arsitek konsultan profesional dan menyelesaikan tahapan konstruksi proyek bangunan dengan sukses.

Berdasarkan yang telah dipaparkan, wadah atau aplikasi untuk membantu masyarakat dalam mencari tenaga arsitek profesional masih sangat minim. Adapun aplikasi sejenis yang mengangkat tema yang sama sejauh ini masih belum diketahui masyarakat luas. Aplikasi sejenis juga akan lebih membantu dalam tahap pengembangan dan langsung diketahui masyarakat secara luas. Dengan demikian, penulis ingin membuat aplikasi agar bisa mempermudah masyarakat dalam mencari kebutuhan seputar desain arsitektur dan tahapan konstruksinya.

Dapat disimpulkan bahwa rancangan aplikasi untuk menghubungkan masyarakat dengan tenaga arsitek konsultan profesional dan tenaga kontraktor bangunan memiliki peluang yang sangat besar. Tentu hal ini mendorong penulis untuk membuat “Aplikasi Arsitekco berbasis *website* untuk menghubungkan masyarakat dengan tenaga profesional di bidang arsitektur dan kontraktor”. Sistem aplikasi yang dibuat akan melihat dari sudut pandang pengguna aplikasi. Melalui aplikasi penulis berharap dapat membantu masyarakat dalam mencari informasi terkait desain arsitektur dan tahap pembangunannya serta memudahkan arsitek konsultan profesional dan kontraktor bangunan dalam memasarkan jasanya.

2. Rumusan Masalah

Berdasarkan latar belakang yang dijabarkan, maka diperoleh suatu rumusan permasalahan:

1. Bagaimana menyajikan karya desain arsitektur para tenaga arsitek konsultan secara terpusat?
2. Bagaimana cara membantu masyarakat dalam menemukan desain arsitektur karya yang disajikan para tenaga arsitek konsultan?
3. Bagaimana mempertemukan para tenaga arsitek konsultan dengan masyarakat?
4. Bagaimana cara membantu masyarakat dalam menyampaikan ide kustomisasi desain arsitektur kepada para tenaga arsitek konsultan?
5. Bagaimana cara penyajian informasi tahapan konstruksi proyek bangunan kepada masyarakat?
6. Bagaimana menyajikan profil tim para kontraktor proyek bangunan secara terpusat?
7. Bagaimana mempertemukan para tenaga kontraktor proyek bangunan dengan masyarakat?

3. Batasan Masalah

Berdasarkan latar belakang yang dijelaskan sebelumnya agar dapat terarah dan permasalahan yang dihadapi tidak terlalu luas serta dengan judul proposal yang diajukan maka ditetapkan batasan-batasan terhadap masalah yang diteliti. Adapun beberapa hal yang membatasi dalam proposal ini, antara lain:

1. studi kasus tenaga arsitektur konsultan PT. Lima Benua Consultant,
2. studi kasus tenaga kontraktor proyek bangunan CV. Anugrah Driatama.

4. Tujuan

Tujuan dari pembuatan aplikasi yang dibuat meliputi:

1. mewadahi karya desain arsitektur para tenaga arsitek konsultan secara terpusat,
2. mempertemukan para tenaga arsitek konsultan dengan masyarakat,
3. mewadahi masyarakat dalam mencari karya desain arsitektur,
4. mewadahi profil tim para kontraktor proyek bangunan secara terpusat,
5. mempertemukan para tenaga kontraktor proyek bangunan dengan masyarakat,
6. memberikan informasi mengenai tahapan konstruksi proyek bangunan.

5. Manfaat

Manfaat yang dapat diperoleh dengan dibuatnya aplikasi ini meliputi:

1. meningkatkan kenyamanan masyarakat dalam mencari desain arsitektur dan tenaga arsitek konsultan,
2. meningkatkan kenyamanan masyarakat dalam memulai tahapan konstruksi proyek bangunan hingga selesai,
3. membangun pemahaman masyarakat mengenai tahapan konstruksi proyek bangunan,
4. meningkatkan peluang kesejahteraan para tenaga arsitek konsultan dan kontraktor proyek bangunan,
5. memberikan kesempatan peluang kerja bagi tenaga arsitek konsultan dan kontraktor proyek bangunan beserta tim konstruksinya,
6. mengurangi angka pengangguran di bidang arsitektur dan konstruksi proyek bangunan.

6. Landasan Teori

6.1 *Scrum*

Scrum adalah sebuah kerangka kerja yang dapat mengatasi suatu masalah kompleks yang selalu berubah, dan juga dinilai dapat memberikan kualitas produk yang baik sesuai dengan keinginan pengguna secara kreatif dan produktif (Schwaber & Sutherland, 2020).

6.2 *Unified Modeling Language (UML)*











Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif objek antara *user* dengan developer, antara developer dengan developer, antara developer analis dengan developer desain, dan antara developer desain dengan developer pemrograman (Suendri, 2018).

UML terdiri atas pengelompokan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Beberapa contoh diagram UML diantaranya yaitu *Use Case Diagram*, *Activity diagram* dan *Class diagram* (Subekti et al., 2014).

a. Diagram *Use Case* (*Use Case Diagram*)

Use case diagram menggambarkan fungsionalitas dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan sistem. Simbol-simbol *use case diagram* dapat dilihat pada Tabel 2.1.

Tabel 6.1 Simbol-simbol *Use Case Diagram*

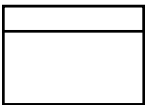
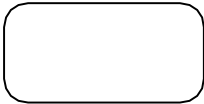
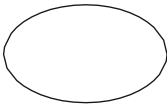




| Gambar | Nama | Keterangan |
|---|-----------------------|---|
|  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> |
|  | <i>Include</i> | Menspesifikasikan bahwa <i>usecase</i> sumber secara eksplisit |
|  | <i>Depedency</i> | Hubungan dimana perubahan yang terjadi pada satu elemen mandiri () akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri |
|  | <i>Generalization</i> | Hubungan dimana objek anak(<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>) |
|  | <i>Extend</i> | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan |
|  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya |
|  | <i>System</i> | Menspesifikasikan paket yang berisi tampilan sistem secara terbatas |
|  | <i>Use Case</i> | Deskripsi dari urutan aksi- aksi yang ditampilkan sistem yang memberi suatu hasil terstruktur bagi aktor |
|  | <i>Collaboration</i> | Interaksi aturan-aturan dan elemen lain yang bekerja untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi) |
|  | <i>Note</i> | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

Sumber: (Hendini, 2016)

b. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *Activity diagram* mesti digunakan sejajar (horizontal) dengan teknik pemodelan lainnya, seperti diagram *Use Case*. *Activity diagram* berfungsi juga untuk menganalisis diagram use case dengan cara mendeskripsikan aktor, *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu. *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis (Hendini, 2016). Simbol *Activity Diagram* dapat dilihat pada Tabel 2.2.

Tabel 6.2 Simbol-simbol *Activity Diagram*

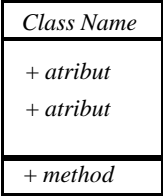




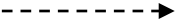
| Gambar | Nama | Keterangan |
|---|----------------------------|---|
|  | <i>Swimlane</i> | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |
|  | <i>Activity</i> | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
|  | <i>Action</i> | State dari sistem yang mencerminkan eksekusi dari suatu aksi |
|  | <i>Decision</i> | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
|  | <i>Initial Node</i> | Bagaimana objek dibentuk atau diawali. |
|  | <i>Activity Final Node</i> | Bagaimana objek dibentuk dan dihancurkan |
|  | <i>Fork Node</i> | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |

Sumber: (Hendini, 2016)

c. Class Diagram

Class diagram adalah gambar grafis mengenai struktur objek statis dari suatu sistem, menunjukkan *class-class* objek yang menyusun sebuah sistem dan juga hubungan antara *class* objek tersebut. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan (Hendini, 2016). Simbol-simbol *class diagram* dapat dilihat pada Tabel 2.7.

Tabel 6.3 Simbol-simbol *Class Diagram*



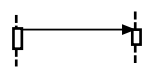

| Simbol | Nama | Keterangan |
|---|-----------------------|---|
|  | <i>Class</i> | Merupakan blok - blok pembangun pada pemrograman berorientasi objek. Terdiri dari nama <i>class</i> , atribut dan <i>method</i> pada <i>class</i> . |
|  | <i>Association</i> | Merupakan sebuah <i>relationship</i> paling umum antara dua <i>class</i> . <i>Association</i> melambangkan tipe-tipe <i>relationship</i> dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah <i>relationship</i> . |
|  | <i>Composition</i> | Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap <i>class</i> tempat dibergantung tersebut. |
|  | <i>Generalization</i> | Relasi antar kelas dengan makna turunan spesialisasi (umum khusus). |
|  | <i>Aggregation</i> | Mengindikasikan keseluruhan bagian <i>relationship</i> dan biasanya disebut sebagai relasi. |
|  | <i>Depedency</i> | Umumnya <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain. |

Sumber: (Hendini, 2016)

d. Sequence Diagram

Sequence diagram merupakan diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*. Simbol-simbol pada *sequence diagram* dapat dilihat pada Tabel 2.9.

Tabel 6.4 Simbol-simbol *Sequence Diagram*

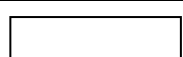

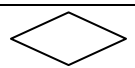
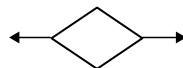
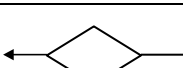
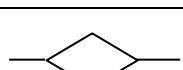
| Simbol | Nama | Keterangan |
|---|------------------------|--|
|  | <i>Actor</i> | Menggambarkan seseorang yang berinteraksi dengan aplikasi. |
|  | <i>Life Line</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |
|  | <i>Object Message</i> | Menggambarkan pesan antar objek yang menunjukkan urutan kejadian yang terjadi. |
|  | <i>Message to Self</i> | Menggambarkan pesan objek itu sendiri yang menunjukkan urutan kejadian. |

Sumber: (Hendini, 2016)

6.3 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan suatu model yang bertujuan menjelaskan hubungan antar data dalam basis data. ERD terdiri dari hubungan antar objek-objek dasar yang dilukiskan dengan menggunakan komponen-komponen data entitas, atribut dan relasi antar entitas (Rohayani et al., 2020). Simbol-simbol *Entity Relationship Diagram* (ERD) dapat dilihat pada Tabel 2.10.

Tabel 6.5 Simbol-Simbol ERD

| Simbol | Nama | Keterangan |
|---|--------------|--|
|  | Entitas | Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain. |
|  | Atribut | Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut. |
|  | Relasi | Menunjukkan hubungan sejumlah entitas yang berbeda. |
|  | Relasi 1 : 1 | Relasi yang menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua |
|  | Relasi 1 : N | Relasi yang menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak. |
|  | Relasi N : N | Hubungan ini menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhubungan dengan banyak entitas pada himpunan entitas yang kedua. |

Sumber: (Edi & Betshani, 2009)

7. Metode Pelaksanaan

7.1 Teknik Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan metode pengumpulan data penelitian kualitatif. Adapun tahapan-tahapan pengumpulan data sebagai berikut.

a. Observasi

Pada penelitian ini, observasi dilakukan di PT. Lima Benua Consultant untuk mengamati proses perencanaan desain proyek bangunan, mulai dari proses pengajuan desain, perancangan rencana anggaran biaya, sampai dengan proses pencarian tenaga kontraktor dan melakukan konstruksi proyek bangunan (Mahendra & Eby Yanto, 2018).

b. Studi Pustaka

Studi pustaka dilakukan untuk mengumpulkan data dan informasi dalam bentuk digital melalui CV. Anugrah Driatama (Mahendra & Eby Yanto, 2018).

7.2 Metode Pengembangan Perangkat Lunak

Model pengembangan perangkat lunak yang digunakan dalam penelitian ini adalah *agile software development methods*, dengan menggunakan Model *Scrum* (Mahendra & Eby Yanto, 2018). Adapun tahapan-tahapan *scrum* yang dilakukan, sebagai berikut.

a. *Backlog*

Menyusun rincian prioritas pada fitur-fitur yang akan dibangun pada tahapan konstruksi proyek bangunan yang akan dibangun. Isi pada fitur-fitur ditambahkan setiap saat.

b. *Sprints*

Menyusun kegiatan yang akan dilakukan untuk memenuhi kebutuhan yang ditetapkan dalam *backlog* dengan durasi realisasi selama 30 hari kerja.

c. *Scrum Meeting*

Menyelenggarakan rapat dengan tim atau pihak yang telah ditunjuk untuk membahas kemajuan kegiatan pengembangan aplikasi Arsitekco berbasis *website* untuk menghubungkan masyarakat dengan tenaga profesional di bidang arsitektur konsultan dan kontraktor.

d. *Demos*

Menunjukkan fitur-fitur *software* yang telah dihasilkan untuk dievaluasi oleh pengguna sesuai dengan waktu yang telah ditentukan.

8. Rencana Kegiatan

Tabel 8.1 Rencana Kegiatan

| NO | KEGIATAN | April 2023 | | | | Mei 2023 | | | | Juni 2023 | | | | Juli 2023 | | | | Agustus 2023 | | | |
|----|--|------------|----|-----|----|----------|----|-----|----|-----------|----|-----|----|-----------|----|-----|----|--------------|----|-----|----|
| | | I | II | III | IV | I | II | III | IV | I | II | III | IV | I | II | III | IV | I | II | III | IV |
| 1 | Identifikasi Masalah | | | | | | | | | | | | | | | | | | | | |
| 2 | Analisis Kebutuhan Sistem | | | | | | | | | | | | | | | | | | | | |
| 3 | Studi Literatur | | | | | | | | | | | | | | | | | | | | |
| 4 | Membuat Rancangan Sistem | | | | | | | | | | | | | | | | | | | | |
| 5 | Implementasi Program | | | | | | | | | | | | | | | | | | | | |
| 6 | Uji Coba Program | | | | | | | | | | | | | | | | | | | | |
| 7 | Revisi Konsep, Desain, Rancangan, Kode Program | | | | | | | | | | | | | | | | | | | | |
| 8 | Penyusunan Laporan Penulisan Tugas Akhir | | | | | | | | | | | | | | | | | | | | |
| 9 | Pelaksanaan Sidang Tugas Akhir | | | | | | | | | | | | | | | | | | | | |
| 10 | Pelaksanaan Revisi Tugas Akhir | | | | | | | | | | | | | | | | | | | | |

DAFTAR PUSTAKA

- Edi, D., & Betshani, S. (2009). Analisis Data dengan Menggunakan ERD dan Model Konseptual Data Warehouse. *Jurnal Informatika*, 5(1), 71–85. <http://www.itmaranatha.org/jurnal.informatika>
- Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang. *Jurnal Khatulistiwa Informatika*, 2(9), 107–116. <https://doi.org/10.1017/CBO9781107415324.004>
- Mahendra, I., & Eby Yanto, D. T. (2018). Sistem Informasi Pengajuan Kredit Berbasis Web Menggunakan Agile Development Methods Pada Bank Bri Unit Kolonel Sugiono. *Jurnal Teknologi Dan Open Source*, 1(2), 13–24. <https://doi.org/10.36378/jtos.v1i2.20>
- Rohayani, Y., Penulis, I., & Bina Sarana Informatika, U. (2020). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Akuntansi Pengiriman Barang. *Jurnal Sistem Informasi Akuntansi*, 01(01), 1–09. <https://ejournal.bsi.ac.id/ejurnal/index.php/justian>
- Schwaber, K., & Sutherland, J. (2020). Scrum Guide V7. *Agile Metrics : Agile Health Metrics for Predictability*, November, 133–152.
- Subekti, M., Lukman, L., Indrawan, D., & Putra, G. (2014). Perancangan Case Tools untuk Diagram Use Case, Activity, dan Class untuk Permodelan Uml Berbasis Web Menggunakan HTML5 dan PHP. *ComTech: Computer, Mathematics and Engineering Applications*, 5(2), 625. <https://doi.org/10.21512/comtech.v5i2.2199>
- Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, 3(1), 1–9. <http://jurnal.uinsu.ac.id/index.php/algorithm/article/download/3148/1871>