

**MEMBANGUN SISTEM INFORMASI PERIZINAN ANGKUTAN UMUM  
DENGAN METODE *EXTREME PROGRAMMING*  
(STUDI KASUS DINAS PERHUBUNGAN  
KABUPATEN BANDUNG BARAT**

***BUILD PUBLIC TRANSPORTATION INFORMATION SYSTEM  
WITH EXTREME PROGRAMMING METHOD  
(CASE STUDY WEST BANDUNG DISTRICT  
DEPARTMENT OF TRANSPORTATION)***

**Olaf Arman Sebastian, Nia Ambarsari, S.Si, M.T., Ridha Hanafi, S.T, M.T.**

Prodi S1 Sistem Informasi, Fakultas Rekayasa Industri, Universitas Telkom  
Institut Teknologi Telkom, Jl. Telekomunikasi – Terusan Buah Batu Bandung 40257

<sup>1</sup> [olafarman@gmail.com](mailto:olafarman@gmail.com), <sup>2</sup> [ambarsarinia@gmail.com](mailto:ambarsarinia@gmail.com), <sup>3</sup> [ridhanafi@gmail.com](mailto:ridhanafi@gmail.com)

---

**Abstrak**

Dinas Perhubungan merupakan dinas pemerintahan yang bekerja dalam mengatur bidang perhubungan pada suatu daerah, salah satunya pada kabupaten Bandung barat. Pengaturan yang dilakukan salah satunya dalam pemberian izin atas pengajuan perizinan angkutan umum di Kabupaten Bandung Barat. Sampai saat ini proses pengajuan perizinan yang berjalan masih terbilang rumit, dikarenakan sistem perizinan yang bekerja kurang membantu pemohon dalam mengajukan perizinan. Seperti halnya, pengambilan dan pengembalian form pengajuan, pengumpulan berkas, hal – hal seperti ini bisa datang setiap harinya dan menyebabkan antrian yang panjang karena sistem yang berjalan kurang memuaskan. Untuk itu diperlukan sistem yang bisa mengolah informasi – informasi perizinan dan mengolah informasi tersebut dengan akurat.

Metode *agile development extreme programming* yang memiliki kelebihan dalam menjalin komunikasi yang baik dengan klien serta meningkatkan komunikasi dan relas terhadap client. Selain itu, *extreme programming* cocok digunakan saat pembuatan sistem membutuhkan waktu yang cepat, sementara kebutuhan sistem selalu berubah – ubah.

Oleh karenanya, aplikasi sistem informasi perizinan yang dinamakan “SIMAPA” yang didasarkan pada metode diatas, dibuat untuk menjawab masalah perizinan pada Dinas Perhubungan Kabupaten Bandung Barat, dalam mengelola informasi – informasi pengajuan perizinan hingga pengesahan perizinan, dan juga pencetakan perihal perizinan tersebut.

**Kata kunci :** Sistem Informasi, Perizinan, Dinas Perhubungan

---

**Abstract**

*Department of Transportation is a government agency that works to regulate transportation sector in an area, that is West Bandung. The one that be regulated is when granting permission for submission of public transport licensing at West Bandung. Up until now, the proses of license submission that works still complex. That is because the licensing system that works still not enough to cover the client request for licenses. For example when they need to submit or return the submission form, collecting the required papers, the things like this always happened and when its too crowded, it's making a long queue that is not fine. That's why a system that can work to gathered those submission information and can process information accurately are required.*

*Agile development extreme programming method have advantage in making good communication between client, and also increasing communication and relationship level to the client. And also, extreme programming match when we using it to make the system that need to be done in a short time, instead the requirement always changing.*

*That's why, license information system application that's called “SIMAPA” that based on the method above, and made to answer the license problem at Department of Transport at West Bandung, to manage information's of license submissions until license validation, and also license card and letter printing.*

**Key :** Information System, License, Department of Transportation

---

## 1. Pendahuluan

Pertumbuhan penduduk menawarkan banyak kesempatan, terutama pada transportasi. Transportasi umum menjadi sektor yang memiliki peran strategis, dan hal ini yang diperlukan di wilayah Bandung Barat. Bandung Barat adalah salah satu kabupaten di Provinsi Jawa Barat yang mempunyai potensi dalam usaha transportasi umum. Dan dalam catatan, kebutuhan masyarakat akan angkutan umum masih tetap tinggi. Sampai tahun ini, akses jalan yang dilalui oleh angkutan umum yang disediakan sudah cukup baik [1]. Namun tanpa adanya proses pengelolaan dan pelaksanaan sistem yang baik, akan dihadapkan kepada kendala pengusaha yang melakukan angkutan umum atau pun bekerja menggunakan angkutan umum di Bandung Barat.

Pertumbuhan penduduk menawarkan banyak kesempatan, terutama pada transportasi. Transportasi umum menjadi sektor yang memiliki peran strategis, dan hal ini yang diperlukan di wilayah Bandung Barat. Masalah ini terjadi karena belum dilakukan sistem pemberdayaan angkutan umum, yakni mengelola dan memperbaharui surat – surat perizinan tiap angkutan umum oleh Dinas Perhubungan Bandung Barat.

Oleh karena nya diperlukan suatu sistem yang bisa mengolah informasi keberlakuan atas suatu izin pada pengelolaan perizinan – perizinan angkutan di Dinas Perhubungan Bandung barat. Proses perizinan yang dilakukan berawal dari proses pendaftaran hingga persetujuan izin yang mengeluarkan surat perizinan beserta kartu pengawasan. Kemudian informasi yang diproses dikemas dan kemudian dapat diintegrasikan ke *web service* simapa, sehingga informasi-informasi yang diberikan bisa dihantarkan ke pemohon lewat jaringan online.

Untuk mengatasi permasalahan diatas, dalam mengembangkan sistem informasi yang akan dibangun diperlukan model pengembangan sistem yang dapat mengatasi perubahan *requirement* dan informasi yang sangat cepat. Permasalahan yang sering muncul dalam tiap pengembangan sistem informasi adalah harusnya adanya penyesuaian antara sistem dengan perubahan informasi dalam waktu yang secara tiba – tiba.

*Agile Development* merupakan proses pendekatan *iterative* yang mengutamakan kepuasan *customer* sebagai prioritas tertinggi dan mempunyai keterlibatan langsung saat pengevaluasian *software*. *Extreme Programming* adalah metode pengembangan software yang membagi seluruh *life cycle* pengembangan software menjadi beberapa tahap pengembangan [2] *Extreme Programming* merupakan salah satu metodologi yang dapat mengimplementasikan metode *Agile Development* yang mempunyai fitur penyederhanaan proses disaat diperlukan perubahan mendadak. Untuk itu dengan mengkombinasikan metode sederhana dan proses pengembangan yang lebih responsif terhadap *customer*, metode *Extreme Programming* cocok untuk digunakan dalam pengembangan Sistem Informasi Perizinan.

## 2. Landasan Teori

### 2.1 Definisi Sistem Informasi

Sistem informasi adalah suatu kombinasi teratur apapun dari *people* (orang), *hardware* (perangkat keras), *software* (piranti lunak), *computer networks and data communications* (jaringan komunikasi), dan *database* (basis data) yang mengumpulkan, mengubah dan menyebarkan informasi di dalam suatu bentuk organisasi [3].

### 2.2 Definisi Angkutan Umum

Angkutan pada dasarnya adalah sarana untuk memindahkan orang dan atau barang dari satu tempat ke tempat lain. Angkutan umum penumpang adalah angkutan penumpang yang menggunakan kendaraan umum yang dilakukan dengan sistem sewa atau bayar. Termasuk dalam pengertian angkutan umum penumpang adalah angkutan kota (bus, minibus, dsb), kereta api, angkutan air, dan angkutan udara [4].

### 2.3 Framework PHP

PHP (akronim dari Hypertext Preprocessor) adalah bahasa pemrograman intrepeter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti oleh komputer secara langsung pada saat baris kode dijalankan sehingga web menjadi dinamis [5]. PHP yang merupakan bahasa pemrograman berbasis web yang memiliki kemampuan untuk memproses data dinamis.

### 2.4 MySQL

MySQL adalah suatu sistem manajemen database SQL yang bersifat *Open Source* dan paling populer saat ini. Suatu database adalah suatu koleksi data terstruktur. Sistem database MySQL mendukung beberapa fitur seperti *multithreaded*, *multi-user*, dan *SQL database management sistem* (DBMS). Database ini dibuat untuk keperluan sistem database yang cepat, handal dan mudah digunakan.

### 2.5 Codeigniter

Codeigniter adalah sebuah *web application framework* yang bersifat *open source* digunakan untuk membangun aplikasi php dinamis. Tujuan utama pengembangan Codeigniter adalah untuk membantu

pengembang aplikasi untuk mengerjakan aplikasi lebih cepat daripada menulis semua kode program dari awal. Codeigniter menyediakan berbagai macam *library* yang dapat mempermudah pengembangan aplikasi.

**2.6 Metode Agile Development Extreme Programming**

*Extreme Programming* (XP) adalah sebuah cara yang ringan, efisien, beresiko rendah, fleksibel, mudah diprediksi, ilmiah dan menyenangkan dalam mengembangkan *sebuah* software [6].

Pada gambar 1 berikut ini adalah table perbandingan metode *Agile Development Extreme Programming* dengan metode *Agile Development* lainnya [7].

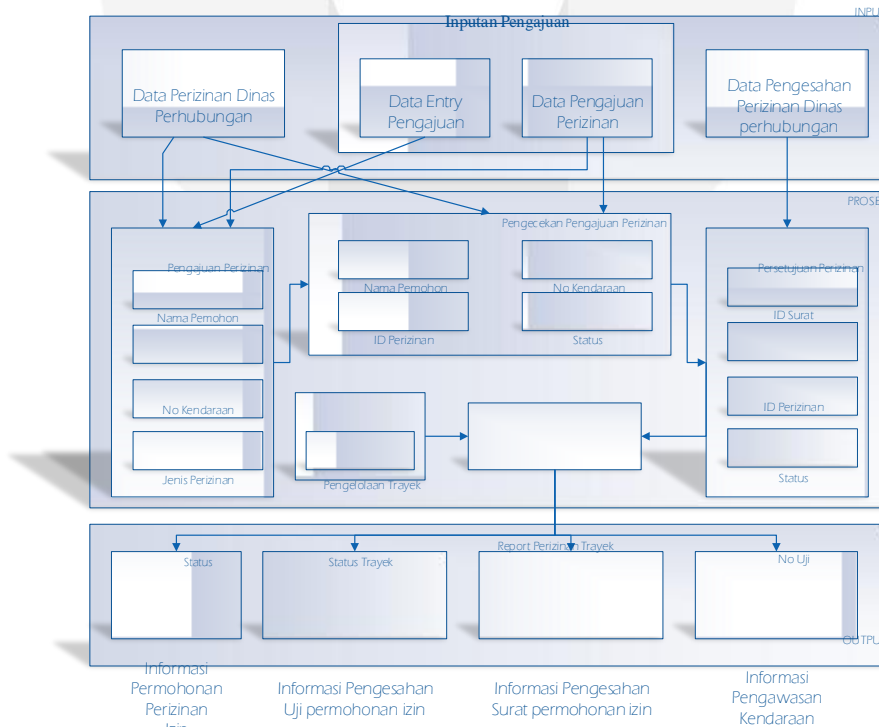
| Characteristic                    | XP  | Scrum  | DSDM   | FDD   | ASD   | Crystal  |
|-----------------------------------|---|--|--|---|---|--|
| Development approach              | Iterative increments  | Iterative increments   | Iterative  | Iterative   | Iterative   | Incremental  |
| Recommended iteration time period | One to six weeks  | Two to four weeks  | 80% solution in 20% total time                               | Two days to two weeks   | Four to eight weeks   | Depending on method from the family                            |
| Project team                      | Smaller teams<br>Less than twenty members   | All sizes<br>(Scrum of Scrums concept)                           | All sizes<br>Independent team                                | Many members<br>More than one team                            | Smaller teams<br>Five to nine team members                              | All sizes<br>Depending on method from the family               |
| Team communication                | Informal<br>Daily stand-up meetings   | Informal<br>Daily stand-up meetings                              | Documentation based  | Documentation based   | Informal<br>Face-to-face  | Informal<br>Face-to-face                                       |
| Project size                      | Smaller projects  | All types of projects  | All types of projects  | More complex projects   | Smaller projects  | All types of projects<br>Depending on method from the family   |
| Customer involvement              | Customer involved   | Customer through the role of Product Owner                       | Customer through frequent releases                           | Customer through reports                                      | Customer through releases   | Customer through incremental releases                          |
| Project documentation             | Only basic documentation  | Only basic documentation   | Documentation exist  | Documentation is important                                    | Only basic documentation  | Only basic documentation                                       |
| Specialties                       | TDD, User stories, Refactoring  | Sprint, Product and Sprint backlog, Planning Poker, Scrum master | Prototyping  | UML diagrams  | Learning cycle  | Adaptable methods family, All types of projects and team sizes |
| Advantages                        | Open workspace, customer as a part of the team, well defined best practices, feedback | High level of communication and collaboration                    | Requirements priority approach, efficient project management | Reports and documentation enables multitasking                | Development of high risk components first, importance of learning cycle | Methodologies that adjusts to project type and size            |
| Disadvantages                     | Weak documentation, lack of discipline, customer presence is mandatory                | Weak documentation, poor control over project                    | Complex documentation  | Individual code ownership, not applicable to smaller projects | Poor method documentation   | Efficient coordination of bigger teams                         |

Gambar 1 Tabel Perbandingan Metode *Agile Development Extreme Programming* dengan Metode *Agile Development* lainnya

**3. Metodologi Penelitian**

**3.1 Model Konseptual**

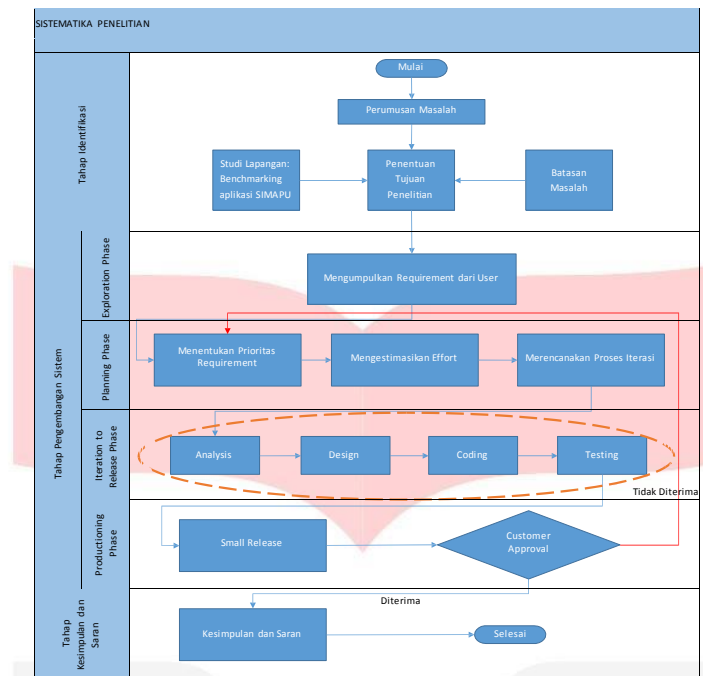
Model Konseptual adalah sebuah persentasi dalam bentuk tertulis maupun visual yang menjelaskan hal – hal utama yang akan dipelajari, faktor – faktor, konsep, atau variabel kunci dan ikatan yang dimiliki mereka secara bentuk naratif atau grafis [8].



Gambar 2 Model Konseptual

**3.2 Sistematika Pemecahan Masalah**

Sistematika penelitian merupakan bagan yang menjelaskan tahap – tahap yang harus dilakukan untuk menyelesaikan penelitian. Sistematika dari penelitian ini adalah sebagai berikut:

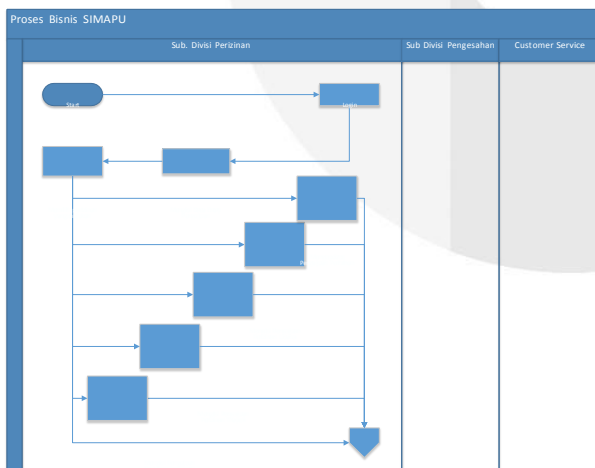


Gambar 3 Sistematisasi Pemecahan Masalah

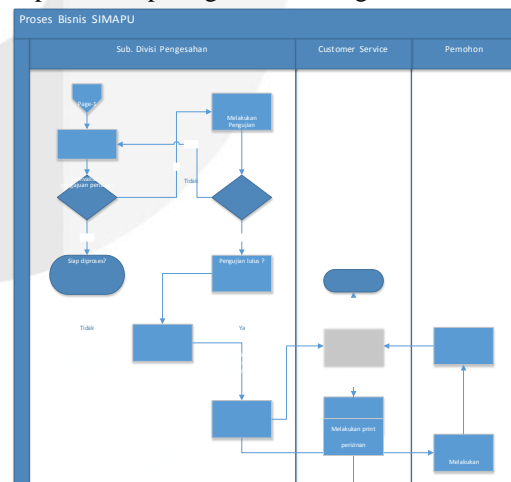
#### 4. Analisis dan Perancangan

##### 4.1 Exploration Phase

Tahap exploration merupakan tahap awal dalam pembangunan aplikasi. Dalam tahap ini, peneliti mengumpulkan seluruh *requirement* yang dibutuhkan untuk membangun aplikasi. Proses bisnis yang diusulkan adalah dengan menggunakan aplikasi penunjang proses bisnis di Dinas Perizinan Bandung Barat. Pada proses ini staff operasional menggunakan aplikasi yang dapat melakukan pencatatan permohonan perizinan angkutan hingga tercetaknya permohonan perizinan yang diajukan. Dimana data – data tersebut dapat tersimpan rapih dalam sebuah database. Bagian *Web Service* juga bisa meneruskan pengajuan perizinan via internet. Diagram gambaran proses bisnis usulan dapat dilihat pada gambar 4 dan gambar 5.



Gambar 4 Proses Bisnis Perizinan-1 SIMAPA



Gambar 5 Proses Bisnis Perizinan-2 SIMAPA

Tabel 1 Identifikasi Kebutuhan Sistem

| No. | REQ ID | Nama Kebutuhan  | Deskripsi  | Jenis User           |
|-----|--------|---|--|----------------------|
| 1   | REQ-01 | Menambah <i>data</i> permohonan perizinan angkutan umum | Proses menambah <i>data</i> permohonan perizinan angkutan umum | Sub Divisi Perizinan |
| 2   | REQ-02 | Mengedit <i>data</i> permohonan perizinan angkutan umum | Proses mengedit <i>data</i> permohonan perizinan angkutan umum | Sub Divisi Perizinan |

|   |        |  |   |                       |
|---|--------|--|---|-----------------------|
| 3 | REQ-03 | Melakukan pengesahan uji atas permohonan perizinan angkutan umum             | Proses persetujuan data berdasarkan uji atas permohonan perizinan angkutan umum | Sub Divisi Pengesahan |
| 4 | REQ-04 | Melakukan pengesahan surat atas permohonan perizinan angkutan umum           | Proses persetujuan data berdasarkan uji atas permohonan perizinan angkutan umum | Sub Divisi Pengesahan |
| 5 | REQ-05 | Membuat kartu pengawasan atas surat permohonan perizinan yang sudah disahkan | Proses pembuatan kartu pengawasan berdasarkan surat permohonan perizinan        | Sub Divisi Pengesahan |
| 6 | REQ-06 | Melakukan print data permohonan perizinan angkutan umum                      | Proses pencetakan data permohonan perizinan yang sudah selesai di proses        | Customer Service      |

Tabel 2 Aktor Sistem

| No. | Nama Aktor            | Deskripsi   |
|-----|-----------------------|---|
| 1   | Sub Divisi Perizinan  | Aktor yang dapat mengolah informasi dan data yang masuk dan sudah ada pada aplikasi                   |
| 2   | Sub Divisi Pengesahan | Aktor yang berwenang mengesahkan pengajuan perizinan dan yang menyetujui pengajuan perizinan tersebut |
| 3   | Customer Service      | Aktor yang menangani pemohon dan melakukan print data hasil akhir permohonan perizinan angkutan umum  |

#### 4.2 Planning Phase

Pada tahap *planning*, peneliti melakukan pengaturan prioritas terhadap *requirement user*, perhitungan estimasi *effort* pada tiap *requirement*, pengaturan jadwal pembuatan *software* berdasarkan perhitungan *effort* sebelumnya dan perencanaan proses iterasi. Pada tahap ini *requirement* akan ditulis ke dalam *user story*. *User Story* adalah model visualisasi untuk menangkap *actor system* serta aksi yang dapat dilakukan oleh aktor. Selanjutnya setiap *user story* akan diberi bobot dan nilai prioritas berdasarkan tiga tingkatan, yaitu sangat dibutuhkan (bernilai 1), menambah nilai bisnis (bernilai 2), dan bagus bila dimiliki (bernilai 3).

Tabel 3 User Story

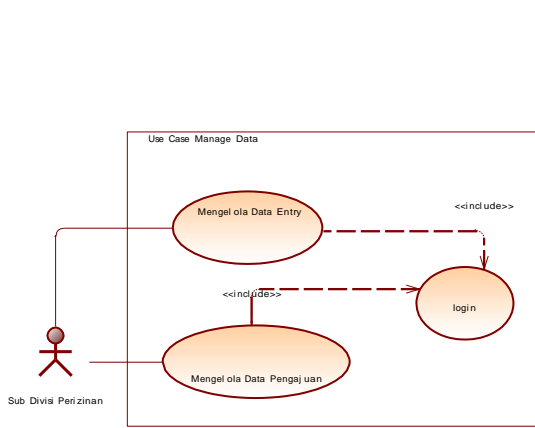
| REQ ID | Aktor                 | Aksi   | Objek   | Prioritas | Estimasi Waktu |
|--------|-----------------------|--|---|-----------|----------------|
| REQ-01 | Sub Divisi Perizinan  | Menambah dan mengedit data permohonan perizinan angkutan umum                      | Proses menambah dan mengedit data permohonan perizinan angkutan umum                          | 1         | 1 hari         |
| REQ-02 | Sub Divisi Pengesahan | Melakukan dan mengesahkan uji atas permohonan perizinan angkutan umum              | Proses pembuatan dan pengesahan data uji atas permohonan perizinan angkutan umum              | 1         | 1 hari         |
| REQ-03 | Sub Divisi Pengesahan | Melakukan dan mengesahkan surat atas permohonan perizinan angkutan umum            | Proses pembuatan dan pengesahan data surat atas permohonan perizinan angkutan umum            | 1         | 1 hari         |
| REQ-04 | Sub Divisi Pengesahan | Melakukan dan mengesahkan kartu pengawasan atas permohonan perizinan angkutan umum | Proses pembuatan dan pengesahan data kartu pengawasan atas permohonan perizinan angkutan umum | 1         | 1 hari         |
| REQ-0  | Customer Service      | Melakukan print data permohonan perizinan angkutan umum                            | Proses pencetakan data permohonan perizinan yang sudah selesai di proses                      | 1         | 7 hari         |

#### 4.3 Iteration to Release Phase

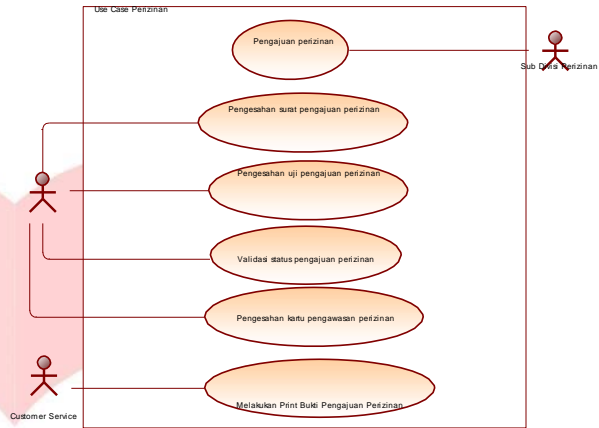
Pada tahap *iteration to release*, akan dijelaskan proses iterasi selama pembangunan aplikasi. Selain itu, peneliti akan melakukan proses *analysis*, *design*, *coding*, dan *testing*. Proses *analysis* dan proses *design* akan menghasilkan beberapa UML diagram seperti *use case diagram*, *activity diagram*, *sequence diagram*, *class*



diagram dan *entity relationship diagram*. Pada proses *coding*, *output* yang dihasilkan berupa aplikasi yang siap dirilis. Pada proses *testing*, aplikasi tersebut akan dicoba dengan menggunakan *black box testing*. *Use Case Diagram* dirancang untuk menjelaskan interaksi apa saja yang dapat dilakukan user terhadap aplikasi. *Use Case Diagram* yang dibuat membahas per sub sistem yang ada pada aplikasi.

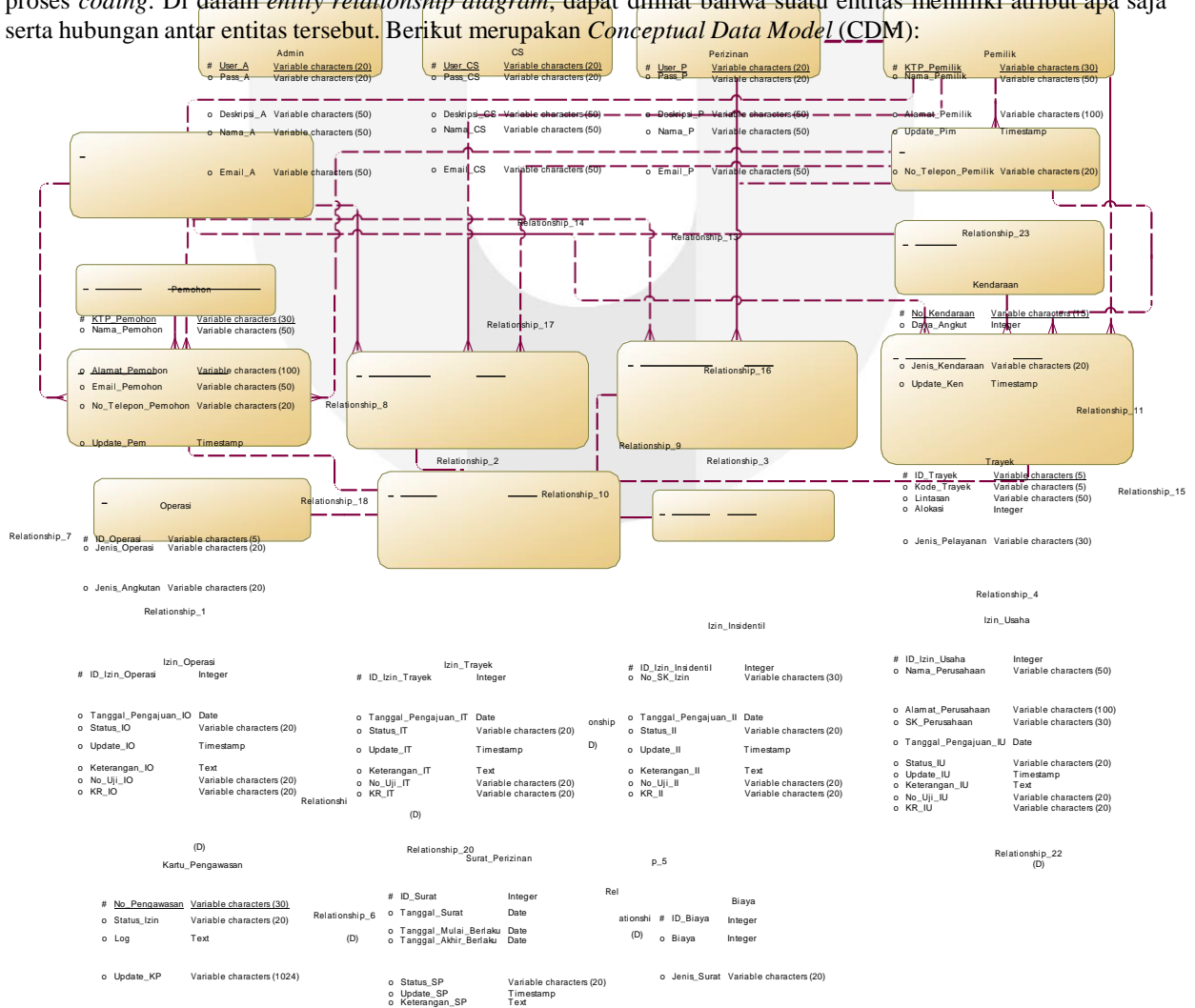


Gambar 6 Use Case Diagram Manage Data



Gambar 7 Use Case Diagram Perizinan

Setelah melakukan perancangan *usecase diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*, selanjutnya dilakukan perancangan database. Perancangan database ini digunakan sebagai acuan untuk melakukan proses *coding*. Di dalam *entity relationship diagram*, dapat dilihat bahwa suatu entitas memiliki atribut apa saja serta hubungan antar entitas tersebut. Berikut merupakan *Conceptual Data Model (CDM)*:

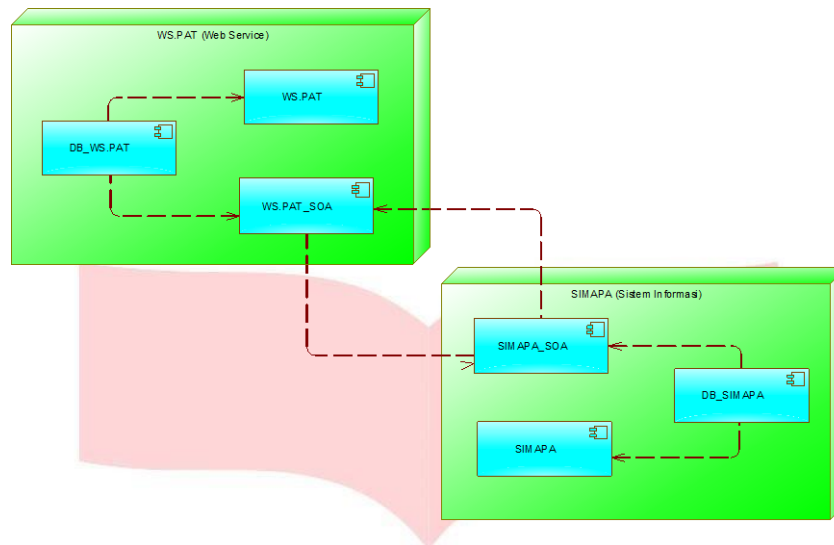


Gambar 8 Conceptual Data Model

### 5. Implementasi dan Testing

Pada bagian implementasi, dijelaskan beberapa kelas yang menjadi komponen implementasi. Kelas – kelas tersebut terbagi berdasarkan konsep MVC (*Model, View, Controller*). Adapun implementasi yang

dilakukan pada aplikasi SIMAPA ini, diintegrasikan dengan aplikasi web online yakni WS.PAT. Integrasi yang dilakukan menggunakan metode SOA(*Service Oriented Architecture*), dimana harus ada yang *request* dan harus ada pula yang menangani *request* tersebut. Berikut merupakan gambar *model deployment* dari implementasi integrasi yang dilakukan :



Gambar 9 Deployment Diagram

Pengujian terhadap aplikasi menghasilkan feedback user yang dilakukan oleh 15 responden yang mencakup metode *extreme programming*, Adapun hasil testing dapat dilihat pada tabel 4.

TABEL 4 Rencana Pengujian

| Nama Dimensi        | Pertanyaan   | Jawaban (responden) |       |
|---------------------|--|---------------------|-------|
|                     |  | Ya                  | Tidak |
| <b>Learnability</b> | Apakah tampilan dan fungsi pada aplikasi SIMAPA mudah dimengerti dan berfungsi dengan baik?          | 70%                 | 30%   |
| <b>Efficiency</b>   | Apakah dengan alur proses aplikasi memberikan kemudahan lebih daripada pengurusan perizinan offline? | 80%                 | 20%   |
| <b>Memorability</b> | Apakah aplikasi SIMAPA mudah dan cocok sehingga bisa terus dilanjutkan penggunaannya oleh pemakai?   | 80%                 | 20%   |
| <b>Errors</b>       | Apakah aplikasi SIMAPA masih mempunyai banyak kesalahan atau ketidak sesuaian pada saat pemakaian?   | 30%                 | 70%   |
| <b>Satisfaction</b> | Apakah tampilan dan fungsi aplikasi SIMAPA sudah sesuai bagi pengguna                                | 70%                 | 30%   |

Hasil dari *usability testing*, dapat terlihat bahwa aplikasi SIMAPA sudah berjalan dengan baik, mudah dimengerti, memberikan desain yang mempermudah pengguna dalam menjalankan aktifitas, mudah diingat dalam penggunaan selanjutnya, error yang terjadi tidak mengganggu ketika menggunakan aplikasi, dan tampilan sudah sesuai dan memberikan kepuasan bagi pengguna

## 6. Kesimpulan dan Saran

### 6.1. Kesimpulan

Kesimpulan yang didapat dari pembangunan aplikasi sistem informasi perizinan ini adalah sebagai berikut:

1. Aplikasi SIMAPA dapat menyediakan sistem pengajuan perizinan hingga pengesahan perizinan yang berjalan di Dinas Perhubungan Kabupaten badnung Barat
2. Aplikasi SIMAPA dibangun menjadi sebuah aplikasi *intranet* sehingga aplikasi diakses berdasarkan jaringan yang terhubung dengan *port server*



## 6.2. Saran

Adapun saran – saran yang diberikan untuk perbaikan dan pengembangan selanjutnya aplikasi SIMAPA ini adalah sebagai berikut :

1. Aplikasi dibangun dengan desain warna yang lebih bervariasi
2. Aplikasi yang dibangun dapat menampilkan pengajuan perizinan hingga pengesahan perizinan

## Daftar Pustaka:

- [1] Setyawan, D. F. (2014). Rancang Bangun Sistem Informasi Surat UIN Sunan Kalijaga Menggunakan Pendekatan Agile Process dengan Model Extreme Programming. *RESUME SKRIPSI*.
- [2] Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile Processes and Methodologies - Aconceptual Study. *Agile Processes and methodologies: A Conceptual Study*, 892-895.
- [3] O'Brien, A. J. (2010). *Introduction to Information Systems*. McGraw-Hill Irwin.
- [4] Warpani, S. (1990). *Merencanakan Sistem perangkatan*. Bandung: ITB.
- [5] Sibero, A. F. (2011). *Kitab Suci Web Programing*. Jakarta: MediaKom.
- [6] Awad, A. M. (2005). *A Comparison between Agile and Traditional Software Development Methodologies*. Australia: The University of Western Australia
- [7] CATALYSTS. (2014). *Different Agile Methodologies*. Retrieved from CATALYSTS KNOWLEDGE BOOSTERS
- [8] Miles, M. B. (1994). *Qualitative Data Analysis: An Expanded Sourcebook*. Beverly Hills: Sage.