JURUSAN SISTEM INFORMASI

FAKULTAS TEKNOLOGI INFORMASI

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

**IDENTITAS PENGUSUL**

**NAMA : AGUNG LAKSONO**

**NRP : 5207100703**

**DOSEN PEMBIMBING I : Ir. Ahmad Holil Noor Ali, M.Kom**

**DOSEN PEMBIMBING II : Feby Artwodini,S.Kom,MT**

**LAB : Perencanaan dan Pengembangan Sistem   
 Informasi**

1. **JUDUL TUGAS AKHIR**

“RANCANG BANGUN APLIKASI ADMINISTRASI KESANTRIAN MADRASAH ALIYAH PP AMANATUL UMMAH, SURABAYA SEBAGAI SOLUSI EFISIENSI WAKTU OPERASIONAL”

1. **ABSTRAK**

*Administrasi kesantrian merupakan bagian penting pada MA Pondok Pesantren Amanatul Ummah, Surabaya. Tanpa adanya administrasi kesantrian, pengelolaan informasi santri akan menghambat keberlangsungan aktivitas pembelajaran. Administrasi kesantrian memiliki 3 domain, yaitu mengelola santri, asrama dan ekstrakurikuler. Administrasi kesantrian sering mengalami permasalahan-permasalahan, sehingga mengganggu aktivitas-aktivitas dari pendidikan yang telah direncanakan.*

*Permasalahan yang ada pada administrasi kesantrian yaitu pengelolaan klasifikasi santri baru, pengelolaan asrama yang masih kacau, kegiatan antar ekstrakurikuler sering berbenturan dan informasi anggota ekstrakurikuler tidak konsisten.*

*Metode penelitian pada tugas akhir ini dimulai dengan pengumpulan data-data pendukung, identifikasi masalah, analisa kebutuhan menggunakan viewpoint-oriented requirement definition(VORD) yang menghasilkan dokumen spesifikasi kebutuhan perangkat lunak(SKPL). Membuat desain sistem dengan menggunakan standart UML berupa dokumen deskripsi perancangan perangkat lunak(DPPL), implementasi sistem sesuai kebutuhan dan desain sistem, verifikasi dan validasi sistem dengan matriks kerunutan serta perbandingan efisiensi waktu kondisi proses sebelum dan sesudah implementasi dan diakhiri dengan pembuatan buku tugas akhir.*

*Hasil yang diharapkan dari tugas akhir adalah sebuah desain dan Aplikasi Sistem administrasi Kesantrian yang Pondok Pesantren berdasarkan studi kasus PP Amanatul Ummah, Surabaya. Sistem Administrasi Kesantrian ini, akan menjadi solusi dalam permasalahan pada bagian kesantrian, yaitu pada konsistensi data kesiswaan, akurasi data terkini, membantu menyediakan data sesuai kebutuhan, efisiensi waktu dan penyimpanan data yang konsisten.***Kata Kunci:** *VORD, UML,Sistem Administrasi Kesantrian, Pondok Pesantren Amanatul Ummah*

1. **PENDAHULUAN**

Sistem administrasi kesantrian merupakan sistem yang dibangun untuk menyediakan layanan tertib administrasi yang menghubungkan antara pondok pesantren dan akademik sekolah pada bagian Kesantrian. Sistem ini dapat mendukung administrasi kesantrian yang berhubungan dengan pembelajaran di pondok pesantren dan di sekolah. Pada bab ini akan dijelaskan tentang Latar Belakang Permasalahan, Perumusan Masalah, Batasan Masalah, Tujuan Tugas Akhir, dan Relevansi atau Manfaat Kegiatan Tugas Akhir.

* 1. **Latar Belakang**

Administrasi Kesantrian merupakan bagian yang cukup penting dalam layanan pendidikan di pondok pesantren. pada bagian kesantrian memiliki beberapa area kerja, diantarannya yaitu: biodata santri, ekstrakurikuler santri, dan penginapan santri.

Permasalahan-permasalahan yang dialami dalam pengelolaan santri di Pondok Pesantren Amanatul Ummah yaitu:

* Informasi asrama tidak terkelola dengan baik, sehingga menyulitkan untuk mengetahui status masing-masing asrama, seperti kapasitas, jumlah santri yang tinggal, tenaga pembimbing di asrama dan *ta'zir* yang diberlakukan.
* Kesulitan untuk mengaturan alokasi asrama bagi santri baru agar dapat diklasifikasikan berdasarkan kemampuan bahasa serta dapat membaurkan santi yang berbeda domisili.
* Informasi ekstrakurikuler tidak konsisten dengan informasi yang ada pada bagian tata usaha, sehingga menyulitkan dalam bagian tata usaha memperoleh informasi santri yang memang mengikuti ekstrakurikuler dengan yang tidak mengikuti.
* Santri diperbolehkan untuk mengikuti lebih dari satu ekstrakurikuler, namun permasalahan yang adalah adalah pada kegiatan-kegiatan ekstrakurikuler sering mengalami benturan, sehingga santri yang mengikuti kegiatan ekstrakurikuler tersebut harus memilih salah satu ekstrakurikuler saja.

Solusi yang pernah dilakukan untuk mengatasi permasalahan tertib administrasi di Pondok Pesantren adalah penggunaan paket aplikasi sekolah(PAS) yang di bangun oleh depdiknas. Namun, permasalahan yang terjadi adalah kurang relevannya fitur tersebut dalam penggunaanya di pondok pesantren. Bagian administrasi kesantrian, dalam sistem kesantrian pada PAS hanya dapat menyajikan informasi dalam lingkup pendidikan akademik, sedangkan informasi yang dibutuhkan di Pondok Pesantren adalah mencangkup akademik dan kepesantrenan.

Berdasarkan permasalahan diatas maka dibutuhkan solusi yang mampu menyediakan sistem yang sesuai dengan kondisi sekolah pada pondok pesantren. Pada projek tugas akhir ini, studi kasus yang diangkat adalah Madrasah Aliyah Surabaya yang menjadi mitra dalam penggalian data dan informasi mengenai pembuatan projek.

* 1. **Perumusan Masalah**

Permasalahan yang dihadapi dalam tugas akhir ini antara lain:

1. Apa saja aktifitas-aktifitas administrasi kesantrian?
2. Bagaimana mengidentifikasi aktifitas tersebut dalam administrasi kesantrian?
3. Bagaimana menganalisis kebutuhan dari aktifitas-aktifitas yang ada pada administrasi kesantrian?
4. Bagaimana membuat desain sistem yang sesuai dengan kebutuhan administrasi kesantrian?
5. Bagaimana membangun desain sistem administrasi kesantrian yang telah dibuat sehingga dapat menjadi aplikasi?
6. Bagaimana memastikan aplikasi yang dibuat telah sesuai dengan desain dan kebutuhan pengguna?
   1. **Batasan Masalah**

Batasan masalah yang terdapat dalam pengerjaan tugas akhir ini antara lain:

1. Masalah yang diselesaikan adalah administrasi kesantrian yang berhubungan dengan akademik dan kepesantrenan.
2. Pengerjaan tugas akhir ini tidak termasuk pemasukkan data dan pelatihan.
3. Pengerjaan tugas akhir ini pada desain sistem dan pembuatan sistem.
   1. **Tujuan Tugas Akhir**

Tujuan dari pembuatan tugas akhir ini adalah manghasilkan rancangan dan Aplikasi Sistem Administrasi Kesantrian Pondok Pesantren yang dapat diimplementasikan pada Madrasah Aliyah di Pondok Pesantren sehingga dapat mengefisiensikan waktu operasional.

* 1. **Relevansi atau Manfaat Kegiatan Tugas Akhir**

Manfaat dari tugas akhir ini adalah menyediakan sistem Administrasi Kesantrian untuk pondok pesantren yang dapat bermanfaat sebagai sistem informasi pendidikan formal madrasah aliyah dan pendidikan pesantren. Sistem dapat ini dapat mengefisiensikan pengolahan informasi yang konsisten dan akurat tentang santri sehingga dapat memangkas kompleksitas dan waktu operasinal pada bagian kesantrian.

1. **TINJAUAN PUSTAKA**
   1. **Madrasah Aliyah Pondok Pesantren Amanatul Ummah.**

MA Unggulan Amanatul Ummah adalah salah satu lembaga pendidikan dibawah yayasan Pondok Pesantren Amanatul Ummah. MA Unggulan Amanatul Ummah didirikan pada tanggal 13 mei 2002 yang di resmikan oleh kementrian agama provinsi Jawa Timur. Madrasah Aliyah Unggulan Amanatul Ummah yang terletak di jalan SiwalanKerto Utara nomor 56 Surabaya dipimpin oleh seorang ketua madrasah yang bernaman Drs. H. Masyhadi, M.Ag.

MA Unggulan Amanatul Ummah memiliki 3 klasifikasi model madrasah, yaitu Madrasag Aliyah Reguler, Madrasah Bertaraf Internasional, dan Akselerasi. Berikut rekapitulasi siswa Madrasah Aliyah Amanatul Ummah pada tahun 2008/2009.

Table . Siswa Pada Madrasah Aliyah Amanatul Ummah

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tahun Ajaran 2008/2009** | **Jumlah** | | | **Jumlah** |
| **Siswa** | | |
| **MA Reguler** | **MBI** | **Akselerasi** |
| Kelas X | 85 | 120 | 54 | 259 |
| Kelas XI | 71 | 68 | - | 139 |
| Kelas XII | 81 | 49 | - | 130 |
| Total | 239 | 237 | 54 | 528 |

Sedangkan untuk tenaga administrasi dan pengajar yang ada untuk Madrasah Aliyah Pondok Pesantren Amanatul Ummah seperti yang di tampilkan pada tabel 2.

Table . Karyawan Madrasah Aliyah Amanatul Ummah

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Guru dan Karyawan | Jenjang Pendidikan | | | | Usia | | |
| > S1 | S1 | <=D3 | JML | 22-50 | 51-59 | JML |
| 1 | Guru | 24 | 60 |  | 84 | 80 | 4 | 84 |
| 2 | Tata Usaha |  | 12 |  | 12 | 12 |  | 12 |
| ­Jumlah | | 24 | 72 |  | 96 | 92 | 4 | 96 |

* 1. **Definisi santri**

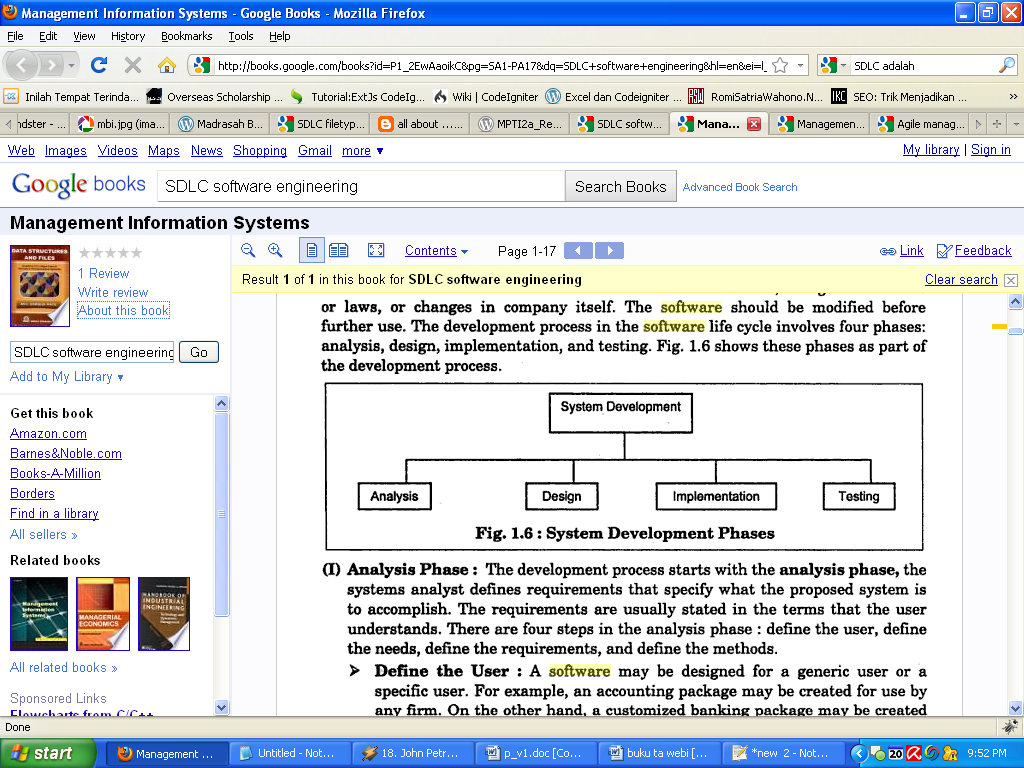
Berdasarkan kamus bahasa indonesia, istilah santri adalah orang yang mendalami agama islam, orang yang beribadah dengan saleh dan sungguh-sungguh. Sedangkan secara istilah istilah santri berasal dari bahasa ‘tamil’ yang berarti ‘guru mengaji’, ada juga yang menilai kata santri berasal dari kata india ‘shastri’ yang berarti ‘orang yang memiliki pengetahuan tentang kitab suci.(Halim, 2010)

Berdasarkan literatur yang telah dipelajari, definisi santri adalah seseorang yang menuntut ilmu pada lembaga atau badan keagamaan islam serta mempelajari ilmu agama dan atau ilmu pengetahuan dengan tujuan menggali ilmu dan berguru kepada seseorang yang lebih ahli.

* 1. **System Development Life Cycle(SDLC)**

Dalam pembuatan rekayasa perangkat lunak, terdapat metode siklus kerja atau lebih dikenal dengan *System Development Life Cycle*(SDLC). Dalam siklus ini terdapat tahapan-tahapan yang diperlukan untuk membuat atau merekayasa perangkat lunak.(Antastina,2010)

Tahapan-tahapan umum dalam SDLC dapat dilihat pada gambar 1.



Gambar 2. System Development Prase

* Fase analisis: fase pertama yang dilakukan dalam proses rekayasa perangkat lunak adalan fase ini. Dalam fase ini dilakukan pendefinisian secara spesifik terhadap tujuan sistem dibuat. Kebutuhan-kebutuhan dalam fase ini terdiri dari 4 langkah, yaitu:
  + Mendefinisikan pengguna
  + Mendefinisikan keinginan pengguna
  + Mendefiniskan kebutuhan pengguna
  + Mendefinisikan metode-metode
* Fase desain, dalam fase ini dilakukan definisibagaimana sistem akan tercapai dengan mendefinisikannya dalam fase analisis. Dalam fase ini, akan ditentukan desain dari file dan atau database secara lengkap.
* Fase implementasi, dalam fase ini dilakukan penulisan kode program.
* Fase uji coba, saat program pertama kali dibuat, butuh dilakukan pengujian. Pada fase ini akan cukup membosankan dan menghabiskan banyak waktu untuk pembuatan program. Dengan adanya sistem uji coba dapat dilakukan koreksi terhadap validitas sistem. Uji coba dilakukan untuk memastikan bahwa sistem dapat berjalan dengan normal. Terdapat 2 tipe dari uji coba, yaitu:
  + Black box(Alpha), uji coba ini dilakukan oleh developer dan pengguna.
  + White box(Beta), uji coba ini dilakukan oleh progremmer sebagai rasa tanggung jawab terhadap program.
  1. **Viewpoint-Oriented Requirements Definition (VORD)**

VORD adalah salah satu cara untuk manganalisa suatu kebutuhan. Perangkat lunak. Metode ini ini dikembangkan oleh Gerald Kotonya and Ian Sommerville pada tahun 1996. VORD mempresentasikan kebutuhan sistem berdasarkan entitas viewpoint. Dalam view point terbagi menjadi 2 bagian yaitu *direct Viewpoint*(viewpoint langsung) dan *Indirect* *Viewpoint*(viewpoint tidak langsung). Dalam direct viewpoint menggambarkan entitas yang berkorespondensi langsung dengan client, sedangkan indirect viewpoint menggambarkan entitas yang memiliki kepentingan untuk menerima layanan dari sistem namun tidak melakukan interaksi secara langsung.(wibowo, 2010)

Dalam sistem VORD memiliki 4 tahap utama, yaitu:

1. *Viewpoint identification*

Melakukan identifikasi pada viewpoint meliputi penemuan viewpoint berdasarkan layanan yang akan diterima oleh setiap viewpoing.

1. *Viewpoint Structuring*

Melakukan strukturisasi viewpoint meliputi pengelompokkan viewpoint menjadi sebuah hirarki. Layanan yang bersifat umum digambarkan pada level yang paling tinggi dan selanjutnya diwarisi oleh viewpoint di levewl yang lebih rendah.

1. *Viewpoint Documentation*

Dalam Dokumentasi viewpoint meliputi cara untuk mendeskripsikan setiap viewpoing dan layanan yang telah ditentukan sebelumnya.

1. *Viewpoint System Mapping*

Tahap ini meliputi implementasi dokumentasi viewpoint menjadi *object-oriented design.*

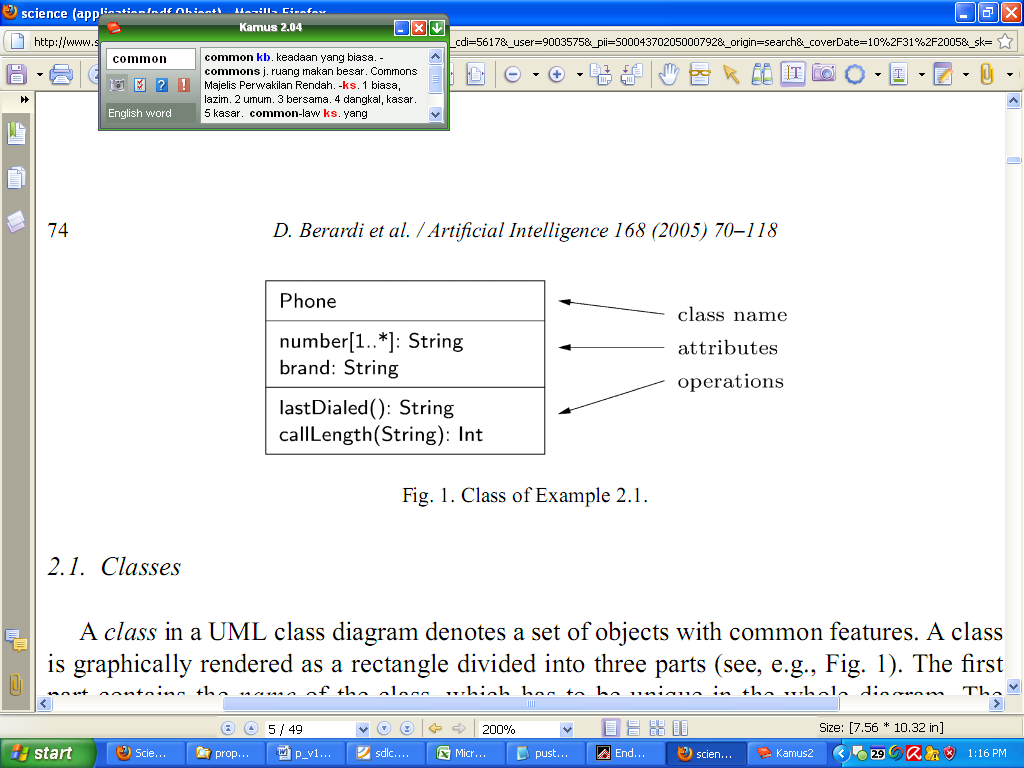
* 1. **Unified Modelling Languages(UML)**

UML adalah bahasa strandar dalam permodelan perangkat lunak dan pembuatan sistem yang berbasis objek. UML digunakan untuk menentukan, menvisualisasikan, membangun dam mendokumentasikan suatu sistem. **(Patil,2007)**

Dalam UML terdapat diagaram-dagram yang menggambarkan dari aplikasi yang akan dibangun, diantaranya *Class diagram*, *Usecase diagram, Sequence diagram.*

1. *Class diagram*

Notasi *Class*  *diagram* padaUML digambarkan sebagai gambaran keadaandari fitur-fitur. Secara tampilan class diagram terbagi menjadi 3 bagian, seperti yang terlihat pada gambar. Pada bagian pertama berisi nama klass, lalu kotak kedua berisi atribut-atribut dari klass dan bagian lainnya berisi nama operasi dari klas atau biasa disebut sebagai fungsi.

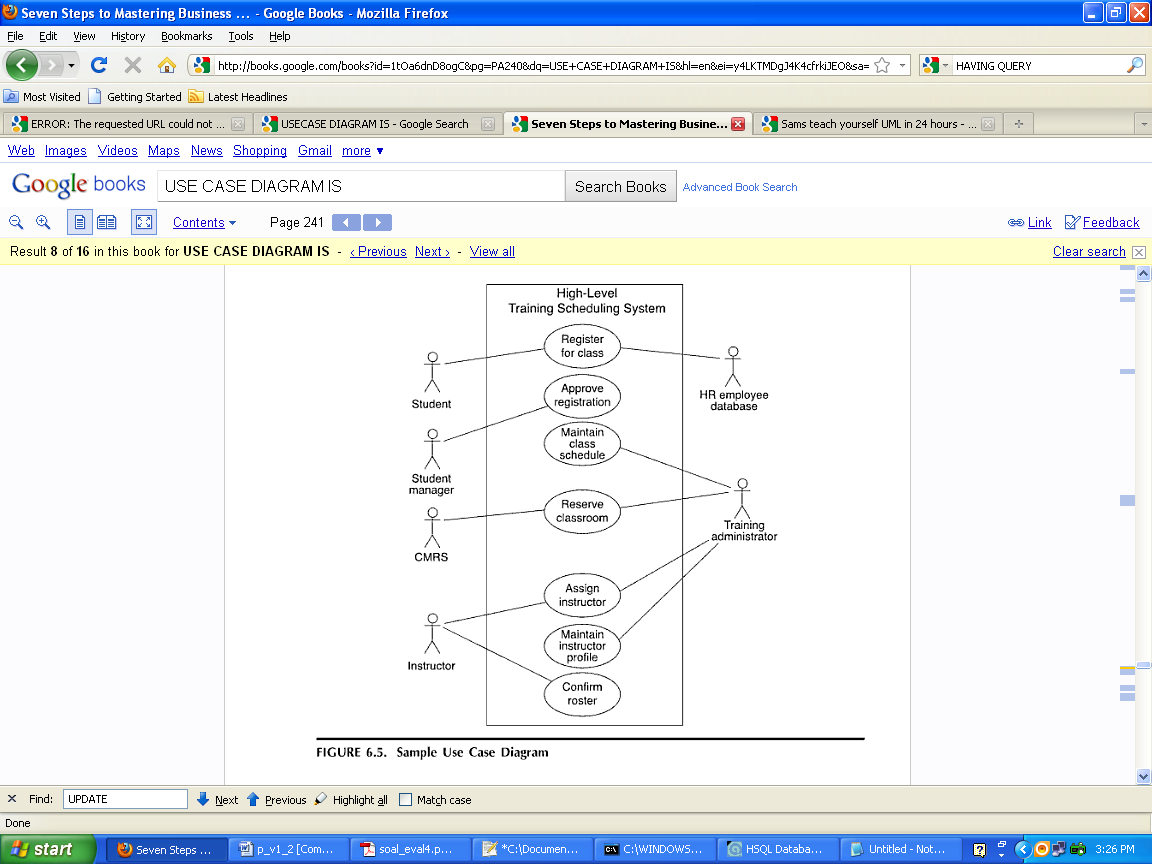


Gambar 3. Struktur Class Diagram

*.*

1. *Usecase diagram*

Dalam *usecase* adalah tujuan dari sistem perangkat lunak. Dalam usecase diagram menggambarkan situasi dari penggunaan atau *usage case*. Dalam usecase menggambarkan sistem berinteraksi dengan pengguna atau aktor.

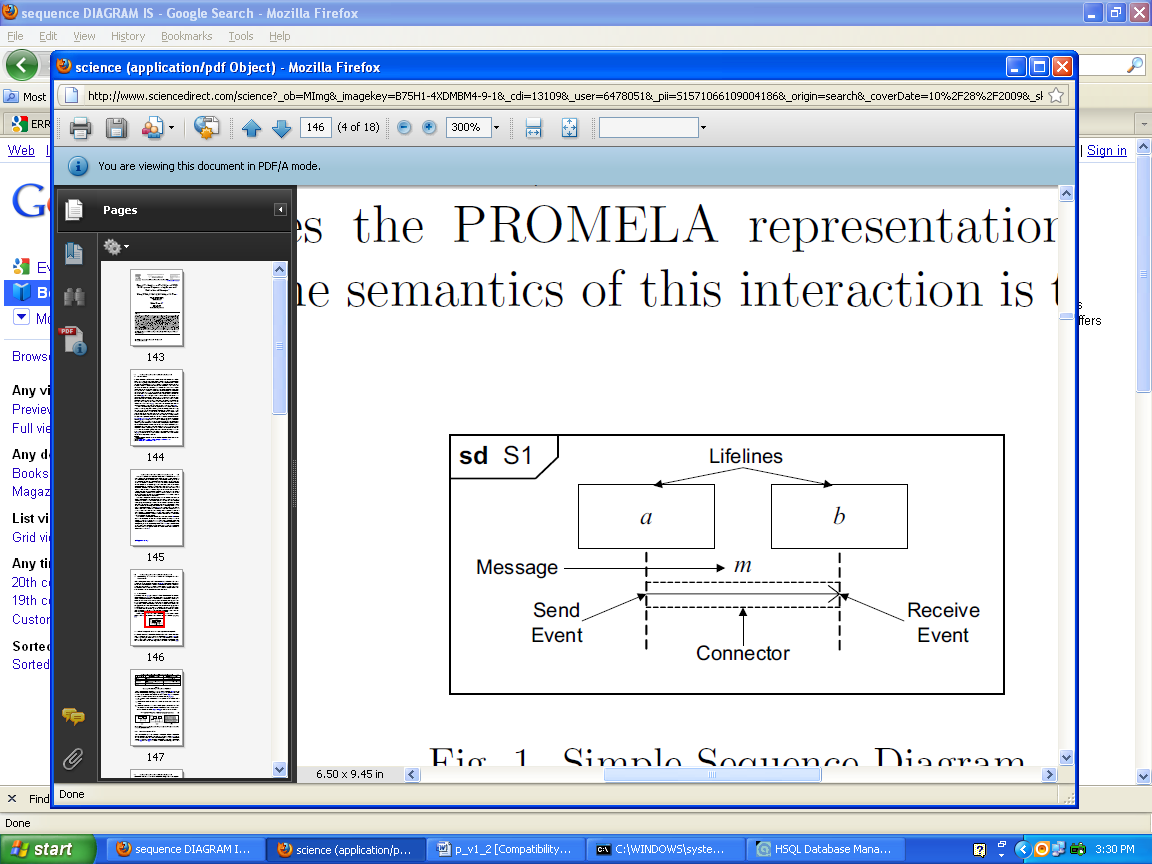


Gambar 4. contoh Usecase

1. *Sequence diagram*

Sequence diagram adalah suatu tipe bagian dari diagram permodelan UML yang menyajikan diagram interaksi objek. Saat projek yang dibuat memiliku struktur yang kompleks, dengan sequence diagram dapat dengan jelas memperlihatkan interaksi asal dan keterikanan objek, beserta kapan objek tersebut akan dieksekusi.( Lima, 2009)

Secara sederhana, sequence diagram seperti yang ditunjukkan pada gambar 4.

****

Gambar 5. Sequence diagram sederhana

* 1. **Management verifikasi dan validasi**

Verifikasi dan Validasi merupakan aktivitas yang penting dalam membangun aplikasi. Aktifitas pada bagian verifikasi dan validasi bertujuan untuk memastikan bahwa kualitas produk yang dibuat telah sesuai dengan kebutuhan pelanggan.

Standart kebutuhan biasanya telah di jelaskan dalam kontrak oleh supplier dan pelanggan. Dengan verifikasi dan validasi memastikan bahwa kontrak yang dibuat sudah terpenuhi atau belum.(colin hood et all, 2008)

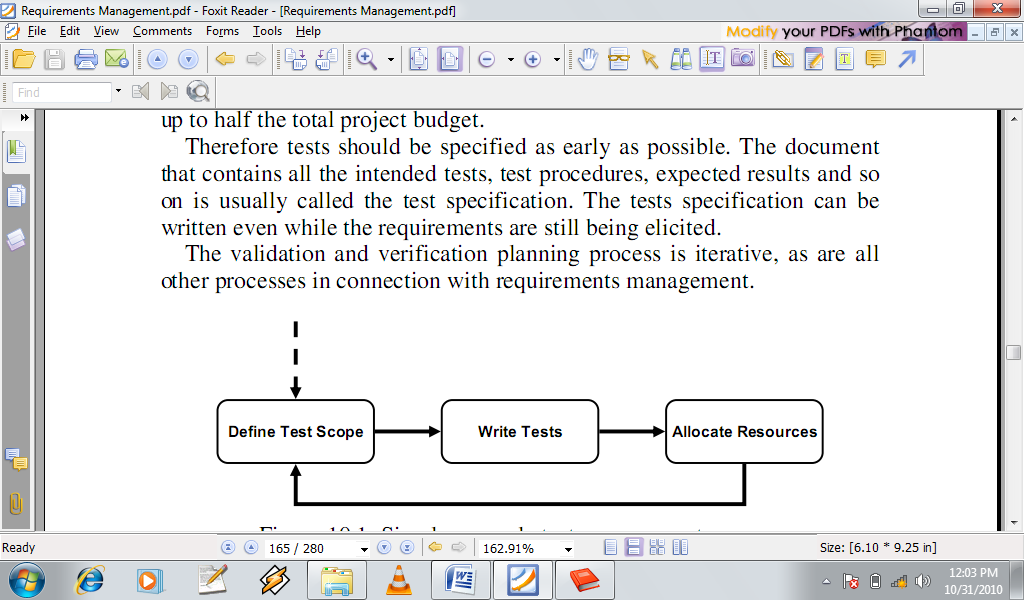
Velidasi adalah melakukan pengecekkan kesesuaian sistem yang dibuat sesuai dengan kebutuhan pelanggan. Sedangkan verifikasi adalah melakukan pengecekan terhadap alur prodak secara spesifik.

Untuk membedakan verifikasi dan validasi dapat dijelaskan seperti pada pertanyaan berikut:

*Are we building the right thing?* (validasi)

*Are we building the thing right?* (verifikasi)

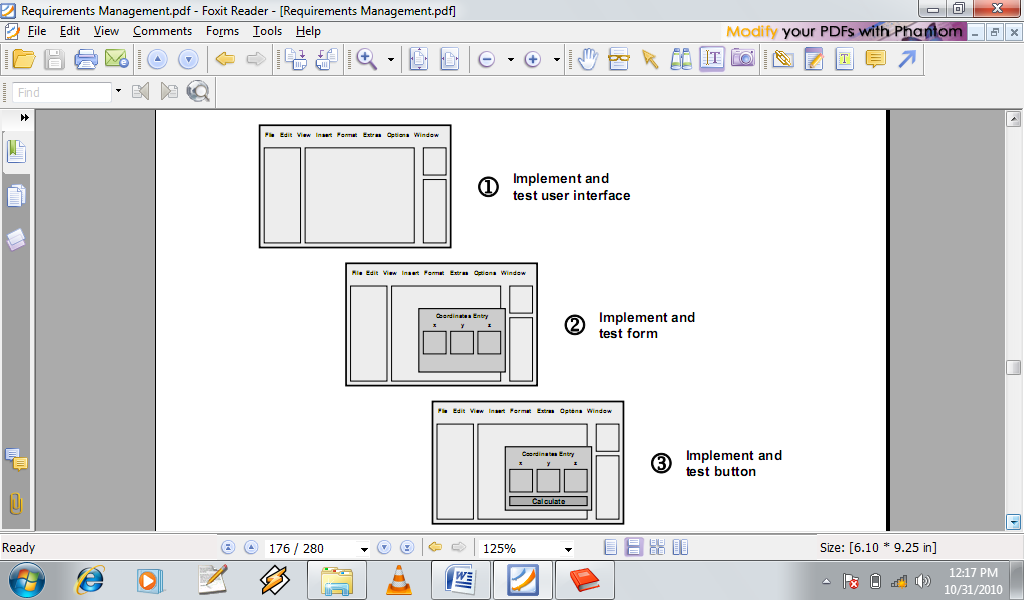
Alur perencanaan iterasi proses validasi dan verifikasi dapat ditunjukan pada gambar.



Gambar 6. Contoh Alur   
manajemen ujicoba proses(Hood, 2008)

Pada gambar diatas memperlihatkan perbedaan spesifikasi kebutuhan pada masing-masing level. Bermula dari kebutuhan pelanggan yang abstrak dan berakhir pada kebutuhan implementasi yang sangat mendetil.

Pada tampilan muka, level yang abstrak harus di ferifikasi kembali mnjadi *project schedule.* Didalam satu kebutuhan terhadapat suatu *seuqence*(aliran), dan urutan ini juga harus didefinisikan. Pada setiap informasi dalam relasi yang ada merupakan sesuat yang sangat bernilai. Berikut adalah implementasi dan urutan ujicoba pada antarmuka.



Gambar 7. Implementasi dan urutan ujicoba   
pada antarmuka aplikasi(Hood, 2008)

Tahapan ujicoba yang dilakukan yaitu pada bagian *userinterface* atau antar muka aplikasi, kemudian dilakukan validasi terhadap *form* yang ada pada interface, setelah melawati tahapan pengujian terhadap *form* kemudian dilakukan uji coba kembali terhadap tombol-tombol yang ada. Jika dalam tahapan ini dapat dilewati, maka pengujian tes validasi telah selesai.

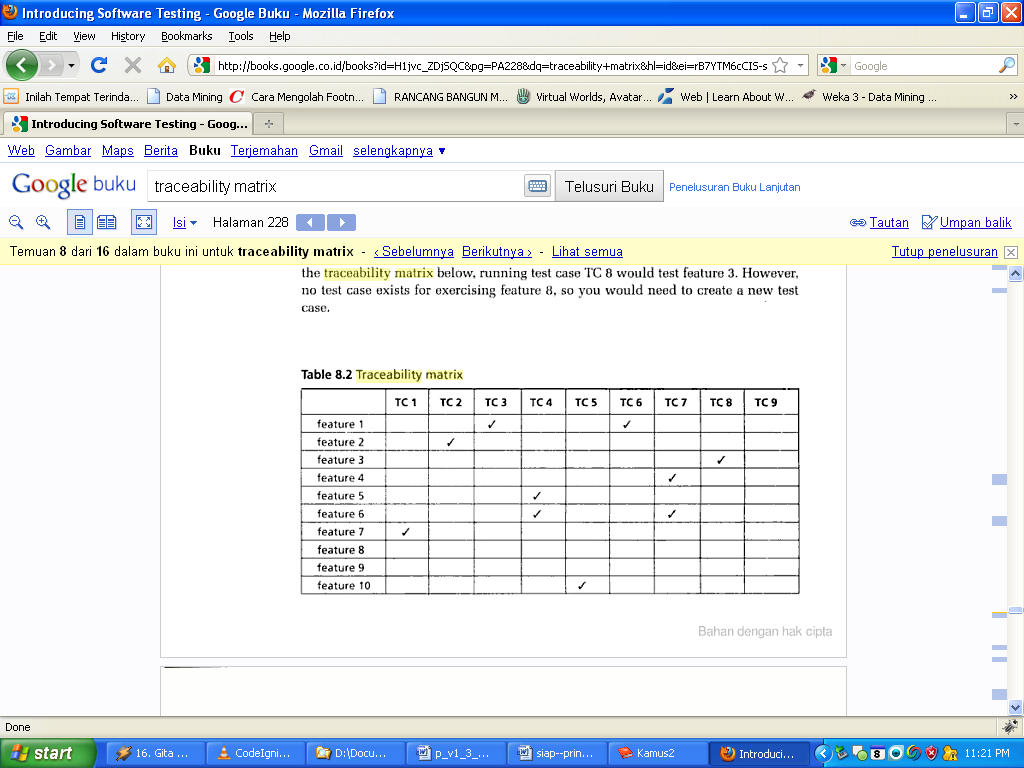
* 1. **Requirements Traceability matrics(matrik kerunutan)**

Matrik kerunutan merupakan bagian perencanaan proses dari pengujian verifikasi. untuk memprioritaskan skema dari kegiatanyang telah ada dilist fitur untuk dilakukan pengujian. pengujian harus dapat memperlihatkan keterhubungan test cases dengan fitur-fitur target. kebutuhan matrik kerunutan membantu memetakan kebutuhan-ekbutuhan untuk dilakukan uji coba, untuk mencapai tujuan dan menyediakan informasi-informasi:

* pengujian test case mana yang dilakukan pada fitur yang spesifik
* kebutuhan mana yang tidak memiliki korespondensi dengan test case
* fitur mana yang butuh dilakukan pengujian
* test case mana yang disimulasikan dengan perubahan-perubahan pada kebutuhan

Berikut merupakan contoh tabel *traceability matrics*

Table . Traceability matrik



Tabel diatas memperlihatkan *traceability matrics,* baris pada tabel merupakan fitur aplikasi, dan bagian colom merupakan komponen test case yang diujikan. setiap fitur aplikasi minimal memiliki korespondensi dengan sebuah test case. (Tamres, 2009)

1. **METODE PENELITIAN**



Gambar 9. Tahapan Penelitian Tugas Akhir

Metode penelitian yang dipakai dalam tugas akhir ini, yaitu:

1. **Pengumpulan data**

Pada metode penelitian ini, dilakukan pengumpulan data dokumen yang dibutuhkan dalam melakukan rekayasa perangkat lunak, dalam metode pengumpulan data dilakukan 3 teknik, yaitu:

* Interview: melakukan wawancara terhadap mitra untuk mengetahui sistem.
* Review dokumen: melihat dokumen yang telah ada, seperti standar operasional dan dokumentasi proses bisnis mitra.
* Observasi: mengamati secara langsung jalannya sistem.

1. **Identifikasi permasalahan**

Pada tahapan ini, data-data yang telah terkumpul digunakan sebagai landasan identifikasi permasalahan pada. Kemudian dilakukan pendaftaran informasi berdasarkan data-data yang telah diperoleh. Dalam identifikasi permasalahan, Infromasi yang dimaksud adalah proses-proses utama yang ada pada Kesantrian Madrasah Aliyah Amanatul Ummah.

1. **Analisa kebutuhan**

Dalam analisa kebutuhan dilakukan dengan pendekatan view-point berbasis objek atau VORD. Pada bagian ini akan menggambarkan interaksi dari antarmuka dari class yang berhubungan dengan pengguna akhir sistem ataupun dengan dengan sistem lain. Keluaran dari langkah ini adalah Spesifikasi Kebutuhan Perangkat Lunak(SKPL)

1. **Membuat desain Mock-up**

Pada tahapan desain sistem, dilakukan spesifikasi dari kebutuhan-kebutuhan yang telah dirumuskan pada bagian analisa kebutuhan, sehingga dapat menjadi model aplikasi yang dapat diimplementasikan. Dalam tahapan ini digunakan UML sebagai standart metode yang digunakan dalam membuat desain sistem. Diagram yang dibuat adalah usecase diagram, sequence diagram dan class diagram. Hasil dari tahapan ini adalah Deskripsi Perancangan Perangkat Lunak(DPPL). Jika pada desain mock-up telah sesuai dengan SKPL maka proses implementasi dapat dilakukan, namun jika desain belum sesuai maka perlu dilakukan peninjauan kembali dokumen SKPL dan DPPL hingga menjadi sesuai dan siap untuk diimplementasikan.loh

1. **Implementasi**

Pada tahapan implementasi, dilakukan pembangunan aplikasi berdasarkan fungsional yang telah didefinisikan pada usecase diagram, class diagram dan sequence diagram sesuai desain Mock-up.

1. **Verifikasi dan Uji validitas**

Pada tahapan validasi dilakukan pengujuan terhadap fungsional antarmuka aplikasi dan memastikan bahwa sudah sesuai dengan kebutuhan-kebutuhan yang telah di rumuskan pada SKPL pada analisa kebutuhan.

Pengukuran perbandingan waktu proses dan operasional sebelum dan sesudah menggunakan sistem administrasi santri.

1. **Penyusunan buku tugas akhir**

Tahapan penyusunan bukut tugas akhir ini merupakan bagian teakhir dari proeses pengerjaan penelitian ini. Dalam tahapan ini, semua hasil yang telah diperoleh dari tahapan-tahapan sebelumnya didokumentasikan secara sistematik sesuai dengan standar yang berlaku dalam penulisan buku tugas akhir di Jurusan Sistem Informasi ITS.

1. **DAFTAR PUSTAKA**

**Antastina**, 2010. review of processing modeling, http://imeldas.blog.ittelkom.ac.id/blog/files/2010/03/MPTI2a\_Review-of-SDLC.pdf, diakses tanggal 20 Oktober 2010

**Hood**, colin. wedermann, simon. Fichtinger stefan. Pautz, urte,. 2008 Requirement Management, the Interface Beteween Requirement Development and all other System Engineering Proccess, Hall 157-174

**Lima,** C. Talhi, D. Mouheb, M. Debbabi, and L. Wang**,** 2009, **“**Formal Verification and Validation of UML 2.0 Sequence Diagrams Using Source and Destination of Messages”, http://www.sciencedirect.com/science, diakses 29 Oktober 2010

**Larasati,** amelia pradipta dewi, 2010**, “**Perancangan Sistem Informasi Manajemen Kapita Selekta di Jurusan Sistem Informasi its’’, http://digilib.its.ac.id/detil.php?id=10283 diakses tanggal 29 Oktober 2010

**Patil**, Mrs Varsha, 2007. data structure and files, http://books.google.com/books?, di akses tanggal 20 Oktober 2010

**Tamres**, Louise. 2009, "introducing software testing", hall 228-229, http://books.google.co.id/books, diakses tanggal 8 November 2010

**Wibowo**, hapsara, 2010. “rancang bangun sistem monitoring perkuliahan dengan metode pembelajaran Student Centered Learning (SCL) di Jurusan Sistem Informasi ITS” hal 15-19

1. **JADWAL KEGIATAN**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Kegiatan** | **Bulan** | | | | | | | | | | | | | |
| **Nopember** | | | **Desember** | | | | **Januari** | | | | **Febuari** | | |
| 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 |
| 1 | Survey dan Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Identifikasi Permasalahan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Analisa Kebutuhan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Membuat Desain Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Verifikasi dan Uji Validitas |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Surabaya, 9 November 2010

Mengetahui/Menyetujui

# Dosen pembimbing I, Dosen pembimbing II

Ir. Achmad Holil Noor Ali, M.Kom Feby Artwodini Muqtadiroh, S.Kom, M.T

NIP. 196606021992031002 NIP. 198302232009122002