

## Rekayasa Kebutuhan Aplikasi *Learning Management System* (LMS) Berbasis Web Menggunakan *Kotonya and Sommerville Process Model*

Rani Kusuma Wardhani<sup>a</sup>, Nina Setiyawati<sup>b\*</sup>

<sup>ab</sup>Fakultas Teknologi Informasi Universitas Kristen Satya Wacana, Salatiga

<sup>a)</sup> [672018026@student.uksw.edu](mailto:672018026@student.uksw.edu), <sup>b\*)</sup> [nina.setiyawati@uksw.edu](mailto:nina.setiyawati@uksw.edu)

### ABSTRAK

Kemampuan berbahasa Inggris merupakan salah satu kemampuan dasar yang harus dimiliki pelajar pada era ini. Umumnya, ditemukan kendala dalam mencapai tujuan pembelajaran Bahasa Inggris, sehingga diperlukan suatu solusi dari permasalahan tersebut. Solusi tersebut dapat berupa suatu sistem yang terintegrasi yang mudah diakses dan menarik perhatian dalam bentuk aplikasi *Learning Management System* (LMS) berbasis web menggunakan *Kotonya and Sommerville Linear Process Model*. Pembangunan sistem tersebut memerlukan satu fase penting yaitu menganalisa kebutuhan, tujuan, fungsi dan batasan-batasan pengembangan. Perancangan rekayasa aplikasi tersebut terdiri dari *elicitation, analysis and negotiation documentation*, dan *validation*. Berdasarkan penelitian yang telah dilakukan, *prototype* dapat membantu mengkomunikasikan gambaran keseluruhan sistem yang akan dibangun kepada pengguna pada tahapan *validation*. Berdasarkan rekayasa kebutuhan yang sudah dilakukan, didapatkan bahwa dengan melakukan rekayasa kebutuhan ini sebelum membangun suatu sistem dapat mempermudah identifikasi masalah terkait pembangunan *learning management system* dapat dibangun sistem sesuai dengan kebutuhan.

**Kata kunci :** *Kotonya and Sommerville Linear, Learning Management System, Rekayasa Kebutuhan*

### ABSTRACT

*English skills are one of the basic skills that should be had by students in this era. Commonly, there is an obstacle in achieving the purpose of English learning, so a solution is needed to resolve that problem. The solution could be an integrated system that is accessible and gets interest in the form of a Learning Management System (LMS) web-based application using the Kotonya and Somerville Linear Process Model. The development system requires one important phase, that is, needs analysis, purpose, function, and negotiation. It also needs validation. Based on research that has been done, a prototype can help communicate a big picture of the system that would be developed to the user in the validation stage. Based on the needs engineering that has been done, it was found that by doing needs engineering before building a system, it can be easier to identify problems related to the development of a learning management system that can fit its requirements.*

**Keywords:** *Kotonya and Sommerville Linear, Learning Management System, Requirement Engineering*

### 1. PENDAHULUAN

Dalam perkembangan kurikulum pembelajaran, bahasa Inggris adalah salah satu mata pelajaran sekaligus kemampuan dasar yang sangat diperlukan dalam era

saat ini di mana orientasi pembelajarannya diarahkan kedalam kemampuan mendengarkan (*listening*), membaca (*reading*), berbicara (*speaking*), dan menulis (*writing*). Dalam penelitian Idham Syahputra banyak peneliti yang

mengelompokkan strategi belajar menjadi 4 jenis: kognitif, meta-kognitif, efektif, dan sosial. Oxford membagi strategi belajar menjadi dua bagian besar; langsung dan tidak langsung. Strategi langsung kemudian dirinci lebih lanjut menjadi tiga jenis; memori, kognitif, dan kompensasi. Strategi tidak langsung dibagi menjadi tiga: meta-kognitif, efektif, dan sosial. Menurut Burdo & Byrd mengemukakan beberapa strategi yang dapat dipilih guru dalam pembelajaran, yaitu strategi Deduktif-Induktif dan strategi Ekspositori langsung dan belajar tuntas [1].

Perkembangan dalam dunia teknologi yang sangat pesat membawa dampak juga dalam dunia pendidikan di mana penyampaian materi dan pembelajaran dapat dengan sistem *on-line learning*, *e-learning* ataupun *web based learning*. Oleh karena itu diperlukan suatu sistem pengelolaan pembelajaran *online* terintegrasi yaitu *learning management system* (LMS). Hiltz (1994) pernah melakukan studi yang membandingkan efektivitas pembelajaran yang dilakukan secara tatap muka dengan pembelajaran *on-line*. Hasil dari percobaan ini menunjukkan bahwa pembelajaran *on-line* lebih efektif dibanding dengan pembelajaran konvensional. Colorita (2001) menjelaskan bahwa pembelajar yang belajar secara *on-line* lebih tinggi tingkat partisipasinya dibanding dengan kelas konvensional. Hasil penelitian Munir (2006) menyatakan bahwa pembelajaran bahasa Inggris dengan menggunakan elektronik (multimedia) lebih signifikan hasilnya dibanding dengan pembelajaran konvensional. Begitu juga hasil penelitian Lengkanawati (2007) mengatakan bahwa pembelajaran bahasa Inggris dengan menggunakan elektronik (multimedia) hasilnya lebih signifikan dibanding dengan menggunakan pembelajaran konvensional [2].

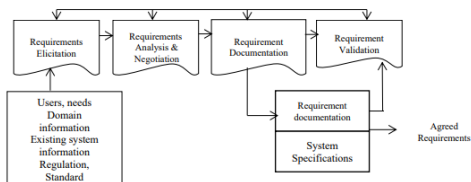
Dalam pengembangan perangkat lunak diperlukannya suatu tahapan menganalisa kebutuhan dari teknologi tersebut yaitu rekayasa kebutuhan atau dikenal sebagai *requirements engineering* salah satu fase kritis dan mendasar dalam pengembangan perangkat lunak. *Requirements engineering* juga merupakan fase pertama yang memengaruhi kualitas dan produktivitas dalam perangkat lunak. Pada fase ini dilakukannya analisa akan menentukan tujuan, fungsi, dan batasan dari sistem perangkat lunak yang dibangun. Ini menunjukkan bahwa fase *requirements engineering* tidak boleh dilewatkan dalam pengembangan perangkat lunak [3].

Dari permasalahan yang didapati perlu adanya suatu strategi baru dalam pembelajaran pada siswa hingga dapat menciptakan kesuksesan dalam pencapaian tujuan pembelajaran diperlukannya suatu solusi dengan menggunakan sistem yang terintegrasi dengan baik yaitu *learning management system* (LMS) yang dapat diakses dengan mudah dan menarik perhatian. Tentunya diperlukan satu fase penting yaitu menganalisa kebutuhan, tujuan, fungsi, dan batasan – batasan dalam pembangunan sistem perangkat lunak nantinya. Dengan demikian dilakukannya perancangan rekayasa kebutuhan aplikasi *learning management system* (LMS) berbasis web menggunakan *Kotonya and Sommerville Linear Process Model*. Untuk tahapan awal dalam pembuatan sistem yang diinginkan sehingga dapat menjadi solusi dari masalah yang ada.

## 2. METODE PENELITIAN

*Requirements engineering* suatu proses dengan tujuan untuk mengidentifikasi, menganalisis, mendokumentasikan dan memvalidasi persyaratan sistem yang akan dikembangkan[4]. Dalam penelitian ini model proses yang digunakan adalah *Kotonya and Sommerville* dengan gambaran tumpang tindih dan dieksekusi

secara berulang seperti yang ditunjukkan pada Gambar 1[3].



**Gambar 1.** Kotonya and Sommerville Linear Requirements Engineering Process model

#### a. Requirements Elicitation

Tahap mengumpulkan persyaratan dari kendala sistem, batasan, dan mengidentifikasi masalah dari sistem yang akan dibangun dengan berkonsultasi dengan pihak terkait[4][5][6]. Sering juga disebut dengan tahap “Requirements Gathering”. Pada tahap ini juga mempelajari dan memahami kebutuhan kemudian diterjemahkan kedalam bahasa yang lebih mudah dimengerti untuk pembangunan aplikasi[7][8].

#### b. Requirements Analysis dan Negotiation

Proses mengecek kebutuhan, konsistensi, kelengkapan, dan kelayakan pada sistem yang akan dibangun. Konflik yang terjadi diproses sebelumnya diselesaikan pada tahap negosiasi sebagai solusi sehingga mendapatkan persyaratan yang telah disepakati[4][5]. Dilakukannya pendekatan dengan menggambarkan sistem yang akan dibangun nantinya sesuai dengan teori jaringan komputer[9][10].

#### c. Requirements Documentation

Memiliki tujuan utama untuk pihak *stakeholders* dan *developers* berkomunikasi untuk kepentingan pengembangan sehingga dapat mengurangi masalah dengan jangka panjang nantinya. Dokumen harus jelas, konsisten, ringkas, dan lengkap[4][5][11].

#### d. Requirements Validation

Tahapan di mana proses memastikan persyaratan akan sistem yang akan dibuat sudah sesuai dengan kebutuhan pengguna dan harapan dari pihak terkait [4][5][12]. Metode yang dapat digunakan dalam tahap ini dengan permodelan dengan *prototype* sebagai pendekatan paling efektif dalam mendemokrasi konsep, menemukan permasalahan dan solusi perbaikan[13][12].

### 3. HASIL DAN PEMBAHASAN

#### • Requirements Elicitation

Pada tahap ini untuk dapat mengidentifikasi masalah dilakukannya wawancara dan membaca beberapa artikel. Sehingga didapatkan beberapa masalah yaitu :

- 1) Banyak pelajar yang tidak dapat belajar lebih mendalam akan bahasa inggris karena biaya kursus yang mahal[14].
- 2) Waktu belajar bahasa inggris yang terbatas di sekolah[15][16].
- 3) Keterbatasan media pembelajaran untuk belajar bahasa inggris[17].
- 4) Metode pembelajaran yang diajarkan di sekolah yang kurang maksimal[16][18].
- 5) Keterbatasan kosakata yang diketahui untuk belajar[16][19].
- 6) Pengucapan *Pronunciation* yang kurang akibat kurangnya digunakan[16][18].
- 7) Perlunya pemberian *reward* dalam meningkatkan motivasi belajar siswa[20][21][22].

Dari permasalahannya yang didapatkan perumusan kebutuhan sistem pada Tabel 1.

**Tabel 1.** Eksplorasi alternatif solusi aplikasi

Tujuan	Kebutuhan	Alternatif Solusi	Kebutuhan	Fitur
Membantu menyediakan pembelajaran dengan akses yang dapat dilakukan kapan saja.	Dibutuhkan media yang dapat diakses kapan dan di mana saja.	Membangun aplikasi berbasis web yang dapat diakses kapan dan di mana saja.	Pengguna dapat melakukan registrasi maupun <i>login</i> dengan <i>Google mail</i> ataupun secara manual.	Registrasi dan <i>Login</i>
Menambah pembelajaran baru yang tidak didapatkan di sekolah.	Materi baru yang belum ada pada kurikulum di sekolah.	Mengembangkan materi baru yang mudah diakses	Pengguna dapat mengubah <i>password</i> jika diperlukan.	
Membantu lebih memahami materi yang telah pelajari di sekolah.	Materi sesuai dengan kurikulum yang ada di sekolah.	Mengembangkan materi yang ada dengan tampilan dan isi yang lebih mudah dipahami.	Admin dapat melakukan <i>login</i> akun yang sudah dibuatkan.	
Membantu pelajar menghemat biaya dalam pembelajaran di luar sekolah.	Media dengan biaya yang terjangkau/gratis.	Mengembang aplikasi dengan biaya maupun tanpa biaya dengan tampilan yang menarik.	Pengguna dapat melihat kelas-kelas yang tersedia dan melihat keterangan dalam kelas tersebut.	<i>List Class</i> dan <i>Detail Class</i>
Menambah semangat dalam belajar dengan pemberian <i>reward</i> .	<i>Reward</i> yang bermanfaat.	Memberikan fitur sertifikat/ <i>badge</i> ketika dapat menyelesaikan kelas yang diambil.	Admin dapat membuat dan mengubah keterangan dari kelas yang ada.	<i>Master Class</i> dan <i>Master Topic</i>
			Pengguna dapat mengubah data yang ada.	<i>Dashboard</i> dan <i>Myclass</i>
			Pengguna dapat melihat kelas-kelas yang telah diambil baik itu sudah selesai maupun sedang dalam pembelajaran.	
			Pengguna dapat mengakses sertifikat kelas yang sudah diselesaikan.	
			Pengguna dapat melihat dan mengakses materi pembelajaran yang ada.	Ruang Kelas
			Pengguna dapat mengambil tes tentang pembelajaran yang sedang diambil.	
			Pengguna mendapatkan feedback dari hasil tes yang dikerjakan.	
			Pengguna dapat mengakses history dari kelas yang diambil dan score dari setiap tes yang sudah dilakukan pada kelas tersebut.	
			Admin dapat menginput materi pembelajaran berupa text, video, maupun gambar.	Input materi
			Admin dapat membuat <i>test</i> dengan pilihan jawaban yaitu	<i>Create Test</i>

Dari solusi yang telah dirumuskan pada Tabel 1 menjadikan landasan untuk menentukan kebutuhan fungsional aplikasi yang akan dibangun ditunjukkan pada Tabel 2.

**Tabel 2.** Kebutuhan fungsional aplikasi

pilihan ganda dan check box jika jawaban lebih dari satu.

Admin dapat memberikan feedback dari semua jawaban yang ada sebagai bahan pembelajaran lagi pada pengguna.

Admin dapat menentukan tes yang dikategorikan sebagai syarat kelulusan maupun tidak.

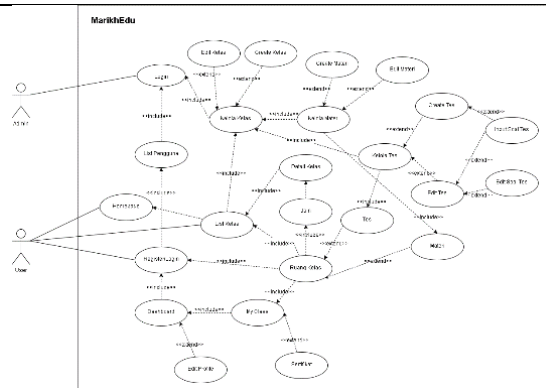
Admin dapat mengatur skor dari setiap *test* yang ada dan poin untuk pertanyaan dalam tes yang ada.

Admin dapat melihat *List Class* dan perkembangan pengguna yang *List Pengguna* ada.

Admin dapat melihat siapa saja pengguna yang mengambil kelas dan sejauh mana dalam menyelesaikan kelas tersebut.

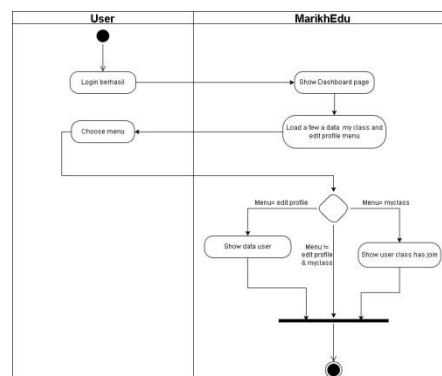
• *Requirements Analysis dan Negotiation*

Tahapan ini dilakukan analisa dari kebutuhan fungsional aplikasi dan diterjemahkan ke dalam bentuk sistem model kemudian dilakukannya juga negosiasi agar aplikasi yang dibutuhkan dapat terealisasi. *Use case diagram* tahap awal di mana mengidentifikasi dan menggambarkan kebutuhan sistem yang akan dibangun[23]. *Use case diagram* juga menjelaskan bagaimana alur dari aplikasi ketika digunakan oleh *user* dengan visualisasi aktor di dalamnya. [24]. *Use case diagram* aplikasi dapat dilihat pada Gambar 2.



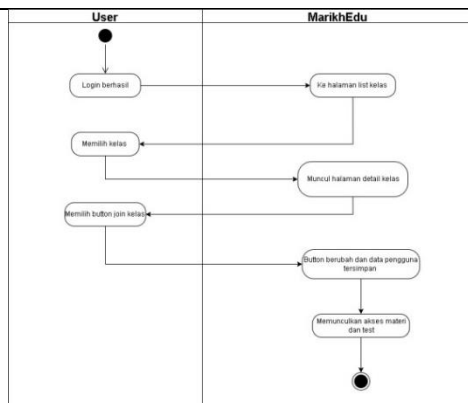
**Gambar 2.** *Use case Diagram Application*

Tahapan analisa selanjutnya adalah pembuatan *activity diagram*. *Activity diagram* adalah suatu proses pemodelan yang menggambarkan aktivitas yang terjadi pada sistem suatu aplikasi[25]. *Activity diagram* pada aplikasi ini dapat dilihat pada Gambar 3, Gambar 4, dan Gambar 5.



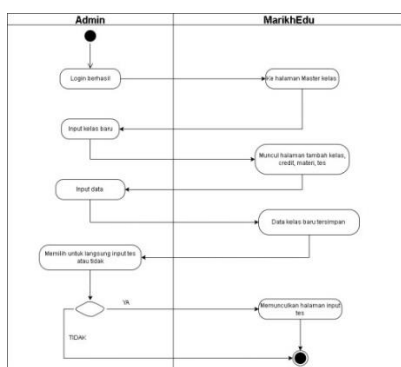
**Gambar 3.** *Activity Diagram User Dashboard*

Gambar 3 menunjukkan proses di mana *user* dalam mengakses halaman *dashboard* di mana terdapat menu untuk mengetahui data diri dari *user* tersebut. Lalu mengakses kelas yang sudah diambil maupun sedang diambil. Dengan kondisi di mana *user* tersebut sudah berhasil melakukan *login*.



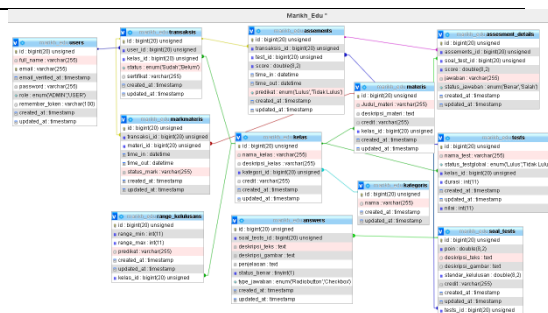
**Gambar 4. Activity Diagram User Join Class**

Pada Gambar 4 digambarkan alur *user* dalam mengambil suatu kelas. Syarat yang harus dipenuhi adalah keberhasilan dalam *login* kemudian mencari kelas yang diinginkan pada halaman *list class*. Kemudian memilih kelas yang sesuai dengan harapan dimunculkannya detail dari kelas tersebut untuk menginformasikan hal-hal yang akan dilakukan pada kelas. Ketika *user* memilih kelas tersebut untuk bergabung di kelas maka akan langsung dapat mengakses materi dan *test* yang terdapat di dalamnya.



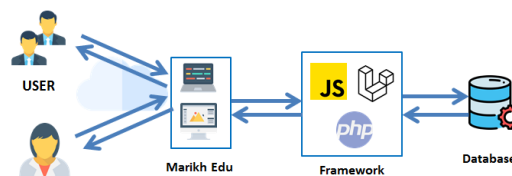
**Gambar 5. Activity Diagram Admin Master Class**

Jika pada Gambar 3 dan Gambar 4 menjelaskan dari sisi *user*. Pada Gambar 5 dijelaskan alur dari segi admin untuk membuat suatu kelas yang nantinya akan tersedia dan dapat diakses oleh *user*. Tidak hanya itu tapi admin juga memasukkan data dan mengatur data *test* dan materi pada kelas tersebut.



**Gambar 6. Database Design**

Database desain tahap di mana menentukan data - data yang terdapat pada aplikasi akan tersimpan di mana dan memiliki hubungan dengan data yang lain atau tidak[25]. Gambar 6 menjelaskan bagaimana hubungan dari satu tabel ke tabel lain dalam aplikasi sehingga data dapat tersimpan sesuai dengan kebutuhan.



**Gambar 7. Architecture Design Application**

Arsitektur aplikasi adalah perancangan aplikasi melalui desain di mana untuk menjelaskan kebutuhan dan komponen yang diperlukan dalam pembangunan aplikasi nantinya[26]. Dapat disebut juga sebagai pondasi dalam pembangunan aplikasi di mana dibutuhkan komputer server sampai ke database yang akan mengintegrasikan aplikasi nantinya[27]. Pada Gambar 7 dijelaskan bahwa pada aplikasi menggunakan *hypertext* PHP dan JS kemudian *Laravel framework* dalam membangun aplikasi tersebut. Aplikasi dapat diakses oleh *user* dan admin melalui *platform* web kemudian dihubungkan melalui *cloud* dan terhubung ke database MySQL untuk penyimpanan dari aplikasi ini.

- *Requirements Documentation*

Kebutuhan fungsional sebagai salah satu cara dalam membantu dalam pengembangan proses pembangunan aplikasi nantinya sehingga mempermudah dalam *developer* membangun aplikasi tersebut[28].

### USER

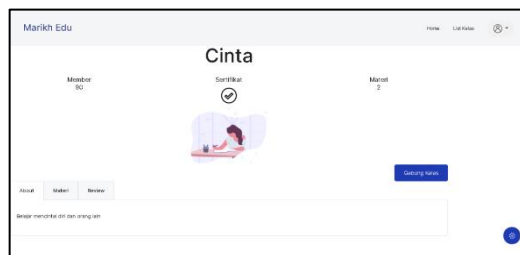
1. Sistem menyediakan fasilitas mengakses materi dan *test*.
2. Sistem menyediakan fasilitas *feedback* ketika tes berlangsung.
3. Sistem menyediakan fasilitas *history* dari kelas yang sudah diambil.
4. Sistem menyediakan fasilitas durasi saat tes berlangsung dan memunculkan nilai.

### ADMIN

1. Sistem menyediakan fasilitas memilih tes yang dijadikan syarat kelulusan dalam kelas.
  2. Sistem menyediakan fasilitas menambah maupun menghapus kelas.
  3. Sistem menyediakan fasilitas tipe jawaban dalam tes.
  4. Sistem menyediakan fasilitas pemberian poin pada tiap nomor tes.
  5. Sistem menyediakan fasilitas untuk *input* total nilai untuk lulus.
- *Requirements Validation*  
Tahap validasi *requirement* dilakukan dengan pendekatan *prototype*. *Prototype* adalah permodelan dari suatu sistem yang ingin dibangun nantinya yang berisikan *storyboard*, tata letak dari fitur yang ingin dibangun[25][29]. Dapat dikatakan juga sebagai gambaran dasar dari program yang ingin dibangun [30].

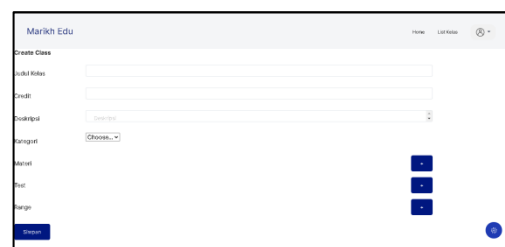
Pembangunan *prototype* dapat membantu menemukan dan memahami persyaratan pengguna serta memungkinkan pengguna untuk terlibat dengan validasi *requirement*. Adapun pada penelitian ini dilakukan

pembangunan *prototype* yang bersifat *high fidelity* [30].



**Gambar 8.** *Prototype Detail Class*

Gambar 8 menggambarkan sistem detail kelas nantinya akan menampilkan seberapa banyak siswa yang sudah mengambil kelas tersebut kemudian terdapat informasi terkait sertifikat apakah tersedia atau tidak dan berapa banyak materinya. Didalam materinya nanti akan dipisahkan sesuai dengan tipe materinya yaitu *listening*, *reading*, *speaking*, *writing*. Dijelaskan juga tentang kelas yang menjelaskan berapa banyak materi terkait *listening*, *reading*, *speaking*, *writing* maupun tes yang perlu dilalui untuk lulus dari kelas tersebut dan ulasan dari siswa yang sudah ambil. Kemudian terdapat *button* gabung kelas di mana ketika nantinya akan berubah ketika sudah gabung.



**Gambar 9.** *Prototype Create Class*

Gambaran membuat kelas pada admin di *prototype* Gambar 9 disediakan untuk *input* judul kelas, guru, deskripsi, dan kategori apakah SMP atau SMA. kemudian diberikan untuk menambahkan materi dan tes lalu pada *range* untuk memberikan ketentuan kelulusan pada kelas tersebut.

## 4. KESIMPULAN

Dari proses rekayasa kebutuhan yang dilakukan, disimpulkan bahwa *prototype* sebagai salah satu pembangunan sistem pada tahap validasi *requirement* dapat membantu mengkomunikasikan gambaran keseluruhan sistem yang akan dibangun kepada pengguna sehingga kebutuhan *user* bisa terpenuhi.

## DAFTAR PUSTAKA

- [1] I. Syahputra, "Syahputra, I. (2006). Strategy of English Learning to Improve Students' language Skill. Kutubkanah: Jurnal Penelitian Sosial Keagamaan, 17(1), 127–145. Strategy of English Learning to Improve Students' language Skill," *Kutubkanah J. Penelit. Sos. Keagamaan*, vol. 17, no. 1, pp. 127–145, 2006.
- [2] M. Munir, "Penggunaan Learning Management System (Lms) Di Perguruan Tinggi: Studi Kasus Di Universitas Pendidikan Indonesia," *J. Cakrawala Pendidik.*, vol. 1, no. 1, pp. 109–119, 2010, doi: 10.21831/cp.v1i1.222.
- [3] A. J. Ekanem, U. I. Valentine, and I. P. Nwakaego, "A Participatory Requirement Engineering Process (PREP) Model for Software Development Projects," vol. 6, no. 9, pp. 10789–10796, 2019.
- [4] F. Paetsch, A. Eberlein, and F. Maurer, "Requirements engineering and agile software development," *Proc. Work. Enabling Technol. Infrastruct. Collab. Enterp. WETICE*, vol. 2003-Janua, no. July, pp. 308–313, 2003, doi: 10.1109/ENABL.2003.1231428.
- [5] N. Ramadan and S. Megahed, "Requirements Engineering in Scrum Framework," *Int. J. Comput. Appl.*, vol. 149, no. 8, pp. 24–29, 2016, doi: 10.5120/ijca2016911530.
- [6] D. Gobov and I. Huchenko, "Software requirements elicitation techniques selection method for the project scope management," *CEUR Workshop Proc.*, vol. 2851, no. Itpm, pp. 1–10, 2021.
- [7] D. Zowghi and C. Coulin, "Requirements elicitation: A survey of techniques, approaches, and tools," *Eng. Manag. Softw. Requir.*, pp. 19–46, doi: 10.1007/3-540-28244-0\_2.
- [8] M. A. Ramdhani, D. S. adillah Maylawati, A. S. Amin, and H. Aulawi, "Requirements elicitation in Software Engineering," *Int. J. Eng. Technol.*, vol. 7, no. 2.29 Special Issue 29, pp. 772–775, 2018, doi: 10.14419/ijet.v7i2.29.14254.
- [9] S. Fricker, "Specification and analysis of requirements negotiation strategy in software ecosystems," *CEUR Workshop Proc.*, vol. 505, pp. 19–33, 2009.
- [10] Universitas Dian Nuswantoro, "Analisis Kebutuhan ( Requirement Analysis )," pp. 44–49, 2011.
- [11] D. W. Nugraha, "Software Requirement Dalam Membangun," *Softw. Requir. Dalam Membangun Sist. Inf. Pelayanan Publik*, no. Faktor utama yang menjadi penghambat dalam pelayanan publik yang baik dapat dianalisa dari dua sisi, yakni birokrasi dan standar pelayanan publik. Sudah menjadi rahasia umum bahwa dalam tubuh pemerintahan negara Indonesia pada semua jenjang dan jenisnya m, p. 11, 2011.



- [12] X. Fei, C. Bin, and Z. Siming, “A Methodology of Requirements Validation for Aviation System Development,” *Proc. 32nd Chinese Control Decis. Conf. CCDC 2020*, no. 525, pp. 4484–4489, 2020, doi: 10.1109/CCDC49329.2020.9164301.
- [13] Y. Yang, W. Ke, and X. Li, “RM2PT: Requirements validation through automatic prototyping,” *Proc. IEEE Int. Conf. Requir. Eng.*, vol. 2019-September, pp. 484–485, 2019, doi: 10.1109/RE.2019.00067.
- [14] R. Riyati, “Tingkat Kesulitan Belajar Bahasa Inggris Pada Siswa Kelas XI SMA Xaverius I Jambi,” no. July, pp. 1–23, 2016.
- [15] I. Widyawati, “Analisis Kesulitan Belajar Siswa Pada Mata Pelajaran Bahasa Inggris Kelas VIII Di SMP N 1 Sawahan Madiun Semester Genap Tahun Ajaran 2010-2011,” *Univ. Sebel. Maret*, pp. 1–106, 2012.
- [16] Parlindungan Pardede, “<https://eeduki.com/2018/11/27/mengapa-banyak-siswa-gagal-belajar-bahasa-asing/>,” *English Education Department (Pend. Bhs. Inggris) UKI*, 2018.  
<https://eeduki.com/2018/11/27/mengapa-banyak-siswa-gagal-belajar-bahasa-asing/> (accessed May 12, 2022).
- [17] “HAMBATAN PENGAJARAN BAHASA INGGRIS, BAGAIMANA MENGATASINYA?,” *SMP NEGERI 8 SURAKARTA*, 2019.  
<https://smpn8solo.sch.id/hambatan-pengajaran-bahasa-inggris-bagaimana-mengatasinya/> (accessed Mar. 20, 2022).
- [18] Gunawan Tambunsaribu and Yusniaty Galingging, “Masalah Yang Dihadapi Pelajar Bahasa Inggris Dalam Memahami Pelajaran Bahasa Inggris,” *Dialekt. J. Bahasa, Sastra Dan Budaya*, vol. 8, no. 1, pp. 30–41, 2021, doi: 10.33541/dia.v8i1.3110.
- [19] “HAMBATAN PENGAJARAN BAHASA INGGRIS, BAGAIMANA MENGATASINYA?,” *SMP NEGERI 8 SURAKARTA*, 2019.
- [20] R. Aksa, “The influence of giving reward on students’ motivation in learning english at senior high schools in barru,” 2018.
- [21] K. Phunghai and A. Boonmoh, “Students’ Perception towards the Use of Rewards to Enhance Their Learning Behaviours and Self-Development,” *JEE (Journal English Educ.)*, vol. 7, no. 1, pp. 39–55, 2021, doi: 10.30606/jee.v7i1.637.
- [22] J. S. M. Nasir, “Reinforcement of Rewards as Motivation for Students in Art Learning: A Case Study at a Local University,” *Adv. Soc. Sci. Educ. Humanit. Res.*, vol. 502, no. Imdes, 2020, doi: 10.2991/assehr.k.201202.082.
- [23] A. Y. Aleryani, “Comparative Study between Data Flow Diagram and Use case Diagram,” *Int. J. Sci. Res. Publ.*, vol. 6, no. 3, pp. 124–127, 2016.
- [24] D. H. Bangkalang, N. Setiyawati, R. Tanone, H. P. Chernovita, and Y. T. B. Tacoh, “a Requirement Engineering in Reporting and Counseling- Based Assistance Application for Victims of Violence,” *J. Ris. Inform.*, vol. 3,

---

pp. 311–318, 2021.

- [25] A. Dennis, B. H. Wixom, and D. Tegarden, *SYSTEMS ANALYSIS & DESIGN An Object-Oriented Approach with UML*. 2015.
- [26] T. J. Betri, E. Utami, and H. Al Fatta, “Perancangan Arsitektur Aplikasi Learning Management System Di Universitas Slamet Riyadi,” *Indones. J. Appl. Informatics*, vol. 2, no. 1, p. 17, 2017, doi: 10.20961/ijai.v2i1.16606.
- [27] U. Novianti S and T. Tristiyanto, “Perencanaan Arsitektur Aplikasi Universitas Lampung Menggunakan Zachman Framework,” *J. Komputasi*, vol. 5, no. 1, pp. 50–59, 2017, doi: 10.23960/komputasi.v5i1.1465.
- [28] S. Hartono and A. R. Tanaamah, “PENERAPAN SOFTWARE REQUIREMENT SPECIFICATIONS PADA SISTEM WEBSITE SINODE GKJ DI SALATIGA ( STUDI KASUS PADA BAGIAN PDILM SINODE GKJ SALATIGA ),” 2018.
- [29] F. Lang and A. Mjöberg, “Prototyping as a Requirements Engineering Technique. (Thesis),” *Lu-Cs-Ex*, 2020.
- [30] R. Puspita, “Pengembangan Prototipe Aplikasi Community Aggregator Beskem dengan Pendekatan UCD Menggunakan Balsamiq Mockup dan Figma,” *Univ. Islam Negeri Syarif Hidayatullah*, p. 189, 2020.