

Analisis Proses Manajemen Kualitas Perangkat Lunak Pada Sistem Informasi Universitas Al Azhar Indonesia

Umar¹⁾, Muhamad Nur Prayogi²⁾, and Setyawan Widyarto³⁾

¹²⁾ *Universitas Budi Luhur, Jl. Ciledug Raya, RT.10/RW.2, Petukangan Utara, Kec. Pesanggrahan, Kota Jakarta Selatan, DKI Jakarta 12260, Indonesia*

²³⁾ *Centre for Graduate Studies, Berlian Building, 4th Floor, Universiti Selangor, Jalan Zirkon A7/A Seksyen 7, Shah Alam, 40000, Selangor Darul Ehsan, Malaysia*

ABSTRAK

Seiring dengan berkembangnya teknologi, kebutuhan perangkat lunak di berbagai bidang terus meningkat setiap tahunnya. Perangkat lunak dibutuhkan di hampir semua sektor untuk membantu menjalankan proses bisnis yang ada. Keberhasilan suatu perangkat lunak dapat dilihat dari sejauh mana sistem tersebut mampu mendukung dan sesuai dengan proses bisnis yang berjalan. Dalam konsep rekayasa perangkat lunak, keberhasilan perangkat lunak tidak hanya ditentukan oleh kesesuaian produk yang dihasilkan terhadap kebutuhan pengguna, tetapi juga oleh proses manajemen dan kualitas perangkat lunak yang diterapkan selama pengembangannya. Namun, berdasarkan temuan dalam penelitian ini, proses manajemen dan kualitas perangkat lunak masih kurang mendapatkan perhatian yang memadai. Hasil studi literatur menunjukkan bahwa kualitas perangkat lunak tidak hanya dinilai dari produk akhir yang dihasilkan, tetapi juga dari proses penjaminan kualitas yang dilakukan selama proses pengembangan.

Pengamatan ini menunjukkan bahwa pengawasan terhadap proses pembangunan perangkat lunak serta penerapan faktor-faktor kualitas pada setiap perangkat lunak yang dikembangkan masih perlu ditingkatkan. Oleh karena itu, diperlukan identifikasi terhadap komponen penjaminan kualitas perangkat lunak yang ada guna memastikan bahwa proses manajemen kualitas dapat berjalan dengan baik serta mendukung kesiapan dalam membangun perangkat lunak yang berkualitas.

Kata Kunci: Kualitas Perangkat Lunak, Rekayasa Perangkat Lunak

ABSTRACT

Along with the rapid development of technology, the demand for software in various fields continues to increase each year. Software is required across many sectors to support and facilitate ongoing business processes. The success of a software system can be evaluated based on how well its functionality aligns with the business processes it is intended to support. In the discipline of software engineering, software success is assessed from several aspects, particularly the suitability of the developed product to the specified requirements. In addition, success can also be evaluated through effective management processes and the overall quality of the software. However, findings from this study indicate that software management processes and quality aspects are often not given adequate attention. Based on the literature review, software quality should not only be evaluated from the final product's conformity to requirements, but also from the quality assurance practices applied throughout the software development process. Observations reveal that supervision of the software development process and consideration of software quality factors in many developed systems still need significant improvement. Therefore, it is necessary to identify the existing software quality assurance components to ensure that quality management processes can function effectively and to support the readiness to develop high-quality software systems.

Keywords: Software Quality, Software Engineering

1. Pendahuluan

Untuk meningkatkan kualitas perangkat lunak diperlukan adanya Manajemen penjaminan kualitas dalam setiap tahap pengembangan perangkat lunak. Kebutuhan penilaian kualitas perangkat lunak memiliki beberapa karakteristik, beberapa diantaranya adalah semua proyek perangkat lunak yang baik dan berkualitas harus memenuhi perhitungan yang tepat untuk kebutuhan dasar.

Sedangkan semua proyek perangkat lunak yang mengalami performansi buruk terutama di beberapa area-area yang penting yaitu perawatan, kehandalan, software reuse, pelatihan, dan penyebab dari performansi yang buruk tersebut adalah karena kurangnya definisi kebutuhan yang menunjang terbentuknya fungsional pada perangkat lunak tersebut.

Dilihat dari beberapa karakteristik diatas, maka diperlukan penilaian penjaminan kualitas perangkat lunak secara baik terukur. Masing- masing faktor kualitas harus dinilai secara detil dan lengkap untuk mendapatkan kualitas terbaik dari perangkat lunak yang digunakan.

Universitas Al Azhar Indonesia merupakan Institusi yang bergerak dibidang pendidikan yang telah menggunakan perangkat lunak sebagai alat bantu dalam melakukan kegiatan administrasinya. Perangkat lunak yang digunakan dibuat oleh salah satu divisi informasi teknologi yang ada di lingkungan Universitas Al Azhar Indonesia

Dengan banyaknya Perangkat Lunak Sistem Informasi yang dibuat oleh divisi informasi teknologi Universitas Al Azhar Indonesia, maka tentunya perlu diperhatikan kualitas dari perangkat lunak tersebut sehingga kinerjanya dapat berjalan maksimal dan terukur performansinya dengan baik serta sesuai dengan kebutuhan.

Dalam penulisan ini, maka dapat Tarik kesimpulan suatu rumusan yaitu “Bagaimana cara menganalisis penjaminan kualitas perangkat lunak dari Sistem Informasi Universitas Al Azhar Indonesia” yang dibangun dalam proses pembuatannya.

Adapun Maksud dari penelitian ini adalah : Membantu pihak Universitas Al Azhar Indonesia untuk mendapatkan gambaran *real* tentang kondisi kesiapan Universitas Al Azhar Indonesia dalam meningkatkan penjaminan kualitas dari perangkat lunak yang dibangun dan memudahkan dalam menentukan komponen apa saja yang dapat meningkatkan kualitas perangkat lunak yang ada terutama dari segi proses pembangunan perangkat lunaknya.

Sedangkan Manfaat Penelitiannya adalah menggambarkan tentang kondisi kesiapan Universitas Al Azhar Indonesia dalam melakukan penjaminan kualitas perangkat lunak dapat digunakan untuk pengembangan divisi Informasi Teknologi Universitas Al Azhar Indonesia sehingga perangkat lunak yang dibangun akan lebih berkualitas, dan perangkat lunak yang telah

ditentukan dapat dijadikan acuan bagi Universitas Al Azhar Indonesia untuk pembangunan perangkat lunak berikutnya.

2. Studi Literatur

Proses Manajemen Kualitas Perangkat Lunak adalah aktifitas pelindung yang diaplikasikan kepada seluruh proses perangkat lunak. Didalam proses Manajemen kualitas perangkat lunak terdapat model faktor. Model Faktor Kualitas Perangkat Lunak tersebut menjadi referensi utama dalam proses penjaminan kualitas perangkat lunak. Beberapa model faktor kualitas perangkat lunak dan kategorisasinya telah diusulkan selama bertahun-tahun. Model klasik dari faktor kualitas perangkat lunak dikemukakan oleh McCall yang terdiri dari 11 faktor [McCall et al, 1977].

Kemudian Model yang berikutnya dikemukakan oleh Deutsch dan Willis (1988) terdiri dari 12 sampai 15 faktor dan oleh Evans dan Marciniak (1987). Alternatif model yang ada tidak berbeda jauh dari model McCall. Perbedaannya berada pada penambahan sudut pandang yang dirasa belum dinilai pada model McCall. Pembahasan ini akan terbagi menjadi dua bagian besar, yaitu:

1. Model faktor McCall

Model faktor McCall mengklasifikasi-kan semua kebutuhan perangkat lunak ke dalam 11 faktor kualitas. Kesebelas faktor tersebut dibagi menjadi tiga kategori sebagai berikut:

- a. Faktor operasi produk
- b. Faktor revisi produk
- c. Faktor transisi produk.

2. Model faktor alternatif

Selain model faktor McCall terdapat beberapa model alternatif yang merupakan perkembangan dari model McCall. Seperti yang telah disebutkan pada poin awal bahwa ada dua model

alternatif yang dipergunakan, yaitu:

- a. Model Evans dan Marciniak
- b. Model Deutsch dan Willis.

Model Kualitas McCall terbagi menjadi 11 faktor kualitas. Penjelasan dari masing-masing faktor kualitas tersebut adalah sebagai berikut:

1. *Correctness*

Sebuah perangkat lunak dapat dikatakan benar jika memenuhi persyaratan sebagai berikut:

- a. Menghasilkan keluaran yang benar untuk setiap kemungkinan masukan oleh pengguna.
- b. Melakukan proses yang seharusnya (tidak kurang dan tidak berlebihan).
- c. Secara formal harus bisa dibuktikan secara matematis.

2. *Reliability*

Sudut pandang reliabilitas pada poin ini lebih menekankan pada kemungkinan dari *failure-free* suatu operasi perangkat lunak terhadap periode waktu tertentu di dalam lingkungan tertentu. Software reliability bukan fungsi langsung terhadap waktu.

3. *Efficiency*

Ada dua pengertian tentang efisiensi sebuah perangkat lunak, yaitu:

- a. Menurut McCall (1977)
Penggunaan sumber daya seperti waktu pemrosesan processor (eksekusi), pemakaian media penyimpanan (memori, space, bandwidth).
- b. Menurut ISO 9126 (1993)
Berkaitan dengan hubungan antara kinerja perangkat lunak dan jumlah sumber daya yang digunakan.

4. *Integrity*

Integritas perangkat lunak pada model McCall lebih menekankan kepada keamanan sebuah perangkat lunak.

Pihak Adam M. Bachtiar, Dian Dharmayanti, Mira Kania S. Developer harus mampu melihat kebutuhan akan hak akses perangkat lunak tersebut pada setiap penggunaannya.

5. *Usability*

Usability mempunyai unsur akademis seperti psikologis, ergonomi, dan human factors [Nielsen, 1993].

6. *Maintainability*

Maintainability adalah kemudahan dari perangkat lunak untuk dipelihara, seperti:

- a. Memperbaiki kerusakan
- b. Menemukan kebutuhan baru
- c. Membuat pemeliharaan selanjutnya lebih mudah
- d. Mengatasi lingkungan yang berubah. Sebuah perangkat lunak dikatakan dapat dipelihara jika koreksi dari minor bugs memerlukan usaha yang kecil.

7. *Flexibility*

Ada dua pengertian tentang faktor fleksibilitas perangkat lunak, yaitu:

- a. Menurut McCall
Kemudahan yang didalam membuat perubahan yang dibutuhkan akibat perubahan lingkungan.
- b. Menurut Boehm
Kemampuan melakukan modifikasi kode untuk memfasilitasi perubahan yang telah ditentukan.

8. *Testability*

Testability adalah kemampuan perangkat lunak untuk diuji. Selain itu testability adalah derajat yang dimiliki sebuah sistem untuk memfasilitasi kriteria pengujian dan performansi dari pengujian tersebut untuk mengukur sejauh mana kriteria tersebut dipenuhi [IEEE, 1990].

9. *Portability*

Perangkat lunak dikatakan portabel jika

biaya untuk memindahkannya (*transport* dan adaptasi) ke lingkungan yang baru lebih kecil jika dibandingkan dengan biaya untuk membangun perangkat lunak tersebut dari awal.

10. *Reusability*

Reusability adalah properti dari perangkat lunak yang memungkinkan perangkat lunak atau modul-modulnya digunakan kembali untuk sistem lain. Suatu perangkat lunak dikatakan *Reusable* yang baik jika modul-modulnya dapat digunakan kembali untuk aplikasi lainnya.

11. *Interoperability*

Interoperability adalah kemampuan suatu perangkat lunak untuk bekerja dengan perangkat lunak lainnya tanpa mengalami kesulitan.

Model Kualitas Alternatif ada beberapa faktor kualitas yang akan dibahas di sini, antaralain:

1. *Verifiability*

Menggambarkan semudah apa memverifikasi performa dari suatu program. Beberapa sub faktor pada *verifiability* adalah sebagai berikut:

- Coding and documentation guidelines* berfokus untuk memberikan panduan dalam menuliskan kode dalam berbagai bahasa pemrograman dan petunjuk untuk mendokumentasikan suatu perangkat lunak dengan baik.
- Compliance (Complexity)* berfokus untuk menjaga kompleksitas kode program yang dibangun sehingga tingkat verifikasi tetap terjaga.
- Document Accessibility* berfokus terhadap kemudahan untuk mengakses dokumentasi yang sudah disebutkan pada sub bab sebelumnya.
- Traceability* berfokus terhadap kemudahan developer untuk melakukan penelusuran suatu dokumentasi yang dimiliki oleh perangkat lunak tersebut.

- Modularity* berfokus kepada kefleksibelan sistem Informasi teknologi Universitas Al Azhar Indonesia. Yaitu kemandirian fungsional dan spesifikasi dari suatu komponen program yang berjalan.
- Generality* adalah seberapa bisa perangkat lunak tersebut bisa menyelesaikan masalah pada domainnya.
- Simplicity* mengenai tingkat dimana perangkat lunak dapat dimengerti tanpa kesulitan.
- Safety* dapat diartikan sebagai kemampuan untuk memperkecil resiko yang dapat membahayakan ke tingkat / level yang dapat diterima. *Safety* juga dapat didefinisikan sebagai kemampuan untuk melakukan: Identifikasi, Analisis, Mempelajari, Mengontrol Terhadap software hazard atau fungsi berbahaya (data & command) untuk memastikan melakukan operasi yang aman [NASA Software Assurance]. *Safety* dapat dipecah menjadi bagian: Pertama Identifikasi, mencari dan menentukan hazard yang mungkin terjadi. Kedua Analisis, menganalisa hazard yang ditemukan untuk mengetahui resiko yang dapat terjadi. Ketiga Mempelajari, mempelajari hasil analisa untuk mencari solusi yang dapat digunakan dan terakhir mengontrol, mengontrol hazard yang telah ditemukan untuk meminimalisasi resiko yang mungkin terjadi.
- Manageability* adalah *Manageability* dapat didefinisikan sebagai kemampuan untuk melakukan tindakan administrasi, melakukan pengawasan serta memperoleh informasi yang relevan dengan tindakan yang terkait. Beberapa

kaitan *manageability* antara lain:
Monitoring: Berkaitandengan aktifitas pemantauan (termasuk pencatatan)

2. *Tracking* : Berkaitan dengan aktifitas penelusuran
3. *Control* : Dengan aktifitas pengendalian/pengubahan.
4. *Survivability* Terdapat dua pengertian untuk *survivability*, yaitu: Kehandalan sistem untuk memberikan layanan ketika terkena bencana, dan kehandalan sistem diukur dari lamanya waktu *failure* dan lamanya waktu *recovery*.

Beberapa langkah yang bisa dilakukan untuk pengendalian bencana, yaitu:

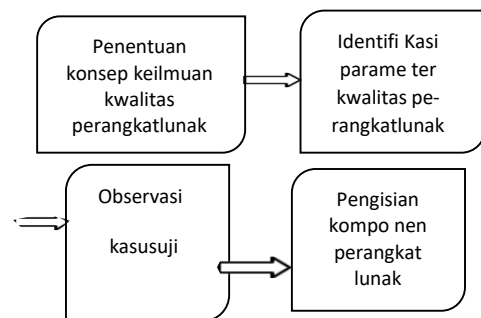
1. *Identify the Business Continuity Components That You Will Focus On (people, property, system, data).*
2. *Define What You're Protecting.*
3. *Prioritize Business Functions.*
4. *Classify Outage Types, Frequencies, and Duration.*
5. *Calculate The Cost of Downtime.*

3. Metodologi Penelitian

Adapun jenis penelitian yang telah dilakukan dalam pelaksanaan penelitian ini ialah penelitian analisis (*analythical research*). Jenis penelitian analisis yaitu jenis penelitian yang melibatkan konsep ilmu pengetahuan dalam menilai suatu kasus. Penelitian analisis lebih memilih konsep yang akan dianalisis yang berhubungan dengan masalah yang ada pada suatu organisasi atau lingkungan.
 Metode Penelitian: Metodologi penelitian yang digunakan pada

penelitian ini adalah: Penentuan Konsep Keilmuan Proses Manajemen Kualitas Perangkat Lunak, Identifikasi Parameter Proses Manajemen Kualitas Perangkat Lunak, Observasi Kasus Uji, Pengisian komponen proses manajemen kualitas perangkat lunak.

Ilustrasi dari metode penelitian ini dapat dilihat pada Gambar 1.



Gambar-1 Metodologi Penelitian

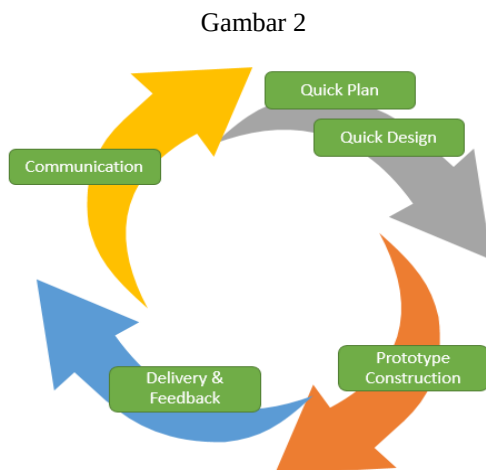
4. Hasil dan Pembahasan

Hasil Observasi Terhadap Sistem Informasi Universitas Al Azhar Indonesia merupakan salah satu institusi pendidikan yang dalam kegiatan sehari-harinya menggunakan perangkat lunak sebagai alat bantu dalam melakukan kegiatannya. Perangkat lunak yang ada di Universitas Al Azhar Indonesia dibangun sendiri oleh Divisi Informasi Teknologi Universitas Al Azhar Indonesia. Perangkat lunak yang dihasilkan oleh divisi Informasi Teknologi Universitas Al Azhar Indonesia merupakan perangkat lunak Sistem Informasi yang umumnya berbasis Web. Berikut ini adalah beberapa contoh dari aplikasi yang ada di lingkungan Universitas Al Azhar Indonesia:

1. Sistem Informasi Akademik (SIKAD UAI)
2. Sistem Informasi Penerimaan Mahasiswa Baru

3. Perwalian/Pengisian KRS Online
4. KHS mahasiswa Online
5. Bagian Administrasi dan Keuangan
6. Kegiatan kemahasiswaan
7. Alumni Online
8. Perkuliahan Online
9. Perpustakaan Online
10. Kegiatan kemahasiswaan Online
11. Manajemen Aset/Inventaris
12. Informasi terupdate (POP UP)
13. Blog Universitas Al Azhar Indonesia
14. Dosen dan Karyawan.
15. Informasi beasiswa.

Model proses yang sering digunakan Universitas Al Azhar Indonesia dalam mengembangkan perangkat lunak adalah model *prototyping* dan *incremental*. Ilustrasi dari model *prototyping*, dapat dilihat pada gambar di bawah ini.



Gambar 2. Model Proses Prototyping

4.1 Hasil Pencocokan Komponen Penjaminan Kualitas Perangkat Lunak pada Universitas Al Azhar Indonesia

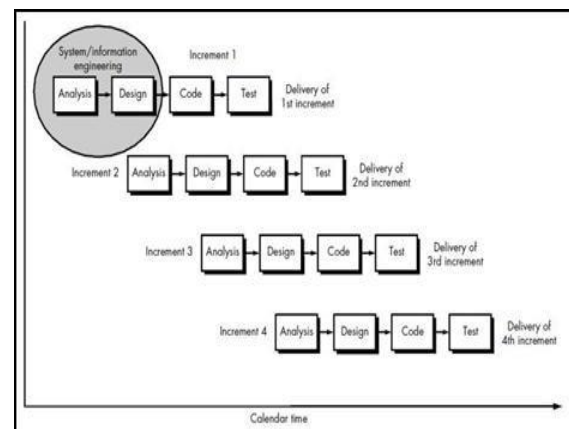
Setelah melakukan observasi menyeluruh terhadap perangkat lunak yang dibangun dan kondisi yang ada di divisi Informasi teknologi, hal berikutnya adalah mengisi borang untuk mengukur sejauh mana kesiapan

Universitas Al Azhar Indonesia dalam melakukan penjaminan kualitas perangkat lunak yang dibangun pada Perencanaan strategis sistem Informasi yang terdapat dalam borang Universitas Al Azhar Indonesia yang digunakan dibentuk berdasarkan konsep penjaminan kualitas perangkat lunak yang telah dipelajari.

4.2 Fakta Kepedulian Universitas Al Azhar Indonesia terhadap Manajemen Kualitas Perangkat lunak

Dari hasil wawancara dan observasi yang kami lakukan di Universitas Al Azhar Indonesia didapatkan fakta sebagai berikut: Di Universitas Al Azhar Indonesia belum memiliki tim sendiri untuk mengurus masalah penjaminan kualitas perangkat lunak yang telah dijalankan, akan tetapi sudah ada divisi tersendiri untuk membangun produk/perangkat lunak, yaitu Divisi Informasi teknologi. Berdasarkan kondisi tersebut maka kondisi organisasi Penjamin Kualitas PL dapat dijelaskan pada Tabel 1.

Gambar 3. Model Proses Incremental



Tabel-1. Organisasi Penjamin Kualitas Perangkat Lunak

SQA Organisasi	Ketersediaan	Penjelasan
SQA manajemen	Tidak ada	---
SQA Unit	Tidak ada	---
SQA Trustees	Tidak ada	---
SQA commiteess	Tidak ada	---
SQA Forums	Tidak ada	Forum khusus SQA belum ada namun pada saat akan membangun perangkat lunak dilakukan rapat koordinasi yang dilakukan pada akhir tahap SDLC.

Tabel 2. Komponen Infrastruktur Kualitas PL

SQA Infrastructure	Ketersediaan	Penjelasan
SQA Procedures	Ada	SQA procedure yang digunakan berupa prosedur maupun standar yang dibuat oleh Universitas Al Azhar Indonesia sendiri yang berorientasi pada produk bukan pada proses SDLC .
SQA Supporting Devices	Tidak ada	-----
SQA Training Instructions	Tidak ada	-----
SQA Infrastructure	Ketersediaan	Penjelasan
SQA Preventive Actions	Tidak ada	-----
SQA Configuration Managements	Tidak ada	-----
SQA Documentation Control	Tidak ada	-----

Tabel 3. Manajemen Kualitas PL

SQA Manajemen	Komponen	Ketersediaan
SQA Project Progress Control	Pengontrolan aktifitas manajemen resiko	Tidak ada
	Pengontrolan jadwal proyek.	Tidak ada
	Pengontrolan sumber daya proyek	Tidak ada
	Pengontrolan biaya proyek.	Tidak ada
SQA Project Progress Control	Metrics process	Tidak ada
	Metrics product	Tidak ada
	Timetable	Tidak ada
	Efektifitas penghilangan error	Tidak ada
	Produktifitas proses software	Tidak ada
	Help desk services	Tidak ada
	Corrective maintenance	Tidak ada
SQA Cost	services SQ Costs	Tidak ada
	Prevention (CC)	Tidak ada
	Appraisal (CC)	Tidak ada
	Manajerial Preparation control cost (C and C)	Tidak ada
	Internal failure costs (FoCC)	Tidak ada
	External failure costs (FoCC)	Tidak ada
	Managerial failure costs (FoCC)	Tidak ada

4.3. Evaluasi SQA Universitas Al Azhar Indonesia

Dari fakta yang didapat dilakukan evaluasi di beberapa hal. Hasil evaluasi tersebut adalah sebagai berikut:

1. Belum tersedianya tim tersendiri untuk mengurus masalah QA di Universitas Al Azhar Indonesia, ~~mmn~~ Universitas Al Azhar Indonesia memiliki Divisi khusus yang menangani pembangunan Perangkat Lunak. Berdasarkan kondisi diatas maka dilihat dari jumlah personil tim SQA kurang memenuhi standar dan tidak ada pembagian pekerjaan yang jelas. Di Universitas Al Azhar Indonesia terkadang 1 orang anggota divisi bertugas lebih dari 1 pekerjaan.
2. Organisasi penjaminan perangkat lunak masih perlu dibentuk.
3. Manajemen kualitas pada Universitas Al Azhar Indonesia masih ada yang belum terpenuhi.
4. Template dokumen yang digunakan merupakan hasil kustomisasi dari setiap kebutuhan perangkat lunak di Universitas Al Azhar Indonesia.
5. Tidak ada *Coding Standard* yang harus ditaati programmer sehingga mempersulit kerja dalam tim.
6. Belum tersedia perhatian khusus terhadap model kualitas perangkat lunak. Baik model Mc Call maupun model alternative

4.4 Kesimpulan dan saran

Dari pembahasan persoalan penanganan kualitas perangkat lunak di Universitas Al Azhar Indonesia dapat kami tarik kesimpulan sebagai berikut :

1. Universitas Al Azhar Indonesia belum memiliki tim SQA yang independen untuk menjamin kualitas setiap perangkat lunak

yang akan dikembangkan

2. Penerapan SQA di Universitas Al Azhar Indonesia sangat berbeda dengan konsep kualitas perangkat lunak yang ada.
3. Untuk mencapai kinerja yang lebih baik, Universitas Al Azhar Indonesia harus membentuk Organisasi SQA secara khusus, sehingga perangkat lunak yang dihasilkan terjamin kualitasnya. Sedangkan saran untuk penelitian ini adalah sebagai berikut :
 1. Standar borang yang digunakan untuk mengukur kesiapan penjaminan kualitas perangkat lunak pada penelitian ini masih didasarkan pada konsep secara umum sehingga perlu dilakukan penelitian lebih lanjut untuk isian borang yang digunakan dalam pengukuranterse
 2. Pengukuran kesiapan penjaminan kualitas perangkat lunak pada penelitian ini dilakukan terhadap perangkat lunak yang sudah dibangun sehingga dibutuhkan penelitian lebih lanjut untuk melakukan pengukuran terhadap pembangunan perangkat lunak yang baru akan dibangun untuk menilai kesiapan dari segi proses

5. Daftar Pustaka

Adler, Paul S. *Practice and Process: The Socialization Software Development*. 2006.

D,Galin, *Software Quality Assurance from Theory to Implementation*, Eng-land: Pearson, 2004

Eko,R,Indrajit,*KiatMemilih Software*. EbizzAsia, 2004.

Program Studi Teknik Informatika Universitas Al Azhar Indonesia, *Perencanaan Strategis Sistim Informasi Universitas Al Azhar Indonesia*, Jakarta, 2023

Divisi Informasi Teknologi Universitas Al Azhar Indonesia, *Plan and Action Sistim Informasi Universitas Al Azhar Indonesia*, Jakarta, 2023