Software Requirements Specification

for

**Easy Print**

Version 1.0 approved

Prepared by Kelompok 7

**11 Desember 2019**

Table of Contents

**Table of Contents ii**

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Tujuan dari dokumen ini adalah untuk menyajikan deskripsi terperinci tentang Sistem Penerbitan Web. Ini akan menjelaskan tujuan dan fitur sistem, antarmuka sistem, apa yang akan dilakukan sistem, kendala di mana ia harus beroperasi dan bagaimana sistem akan bereaksi terhadap rangsangan eksternal. Dokumen ini ditujukan untuk para pemangku kepentingan dan pengembang sistem yang akan diusulkan.

## Document Conventions

Teknologi Easy Print diperkenalkan Firsta di Windows Server 2008 sebagai alternatif untuk penggunaan subsistem pencetakan tradisional pada server Remote Desktop. Remote Desktop Easy Print menghindari instalasi driver untuk printer yang diarahkan pada server terminal (RDS) dan memungkinkan untuk dengan mudah memetakan printer yang dialihkan klien ke driver Easy Print. Ini secara signifikan meningkatkan stabilitas dan kinerja pekerjaan layanan Print Spooler dan server RD secara keseluruhan.

Teknologi EasyPrint sangat nyaman ketika beberapa lusinan perangkat pencetakan yang berbeda (dengan driver yang berbeda) digunakan pada komputer RDS pengguna, atau ketika printer tidak memiliki driver yang benar untuk server RDS (masalah umum dengan model printer rumah).

Fungsionalitas Easy Print didasarkan pada spesifikasi yang ditentukan dalam XPS standar (Spesifikasi Kertas XML) untuk driver printer. Keuntungan utama format XPS: format dokumen terbuka, portabel dan independen dari platform perangkat keras, persyaratan rendah untuk bandwidth saluran (melalui penggunaan standar XML yang lebih ringan). Pekerjaan cetak XPS melalui sesi RDP ditransfer ke PC klien dan diproses oleh driver cetak lokal.

Driver EasyPrint mengubah pekerjaan yang dikirim untuk dicetak ke format XPS dan mentransfernya ke perangkat klien, di mana ia dicetak menggunakan driver printer asli. Satu kelemahan penting dari teknologi EasyPrint dapat dicatat - biaya sumber daya tambahan (CPU dan RAM) pada server RDS untuk mengkonversi dokumen, dan dokumen melalui EasyPrint biasanya membutuhkan waktu sedikit lebih lama untuk dicetak.

## Intended Audience and Reading Suggestion

**Intended Audience :**

Semua personel yang bertanggung jawab atas implementasi cetak (keluaran dokumen) infrastruktur, strategi cetak atau pemesanan perangkat faks, cetak, pemindaian, atau fotokopi atau persetujuan pesanan semacam itu harus membaca dan memahami dokumen ini.

**Reading suggestions :**

* **Cetak Buku Saya Dapat Membacanya dalam PDF**
* **Memberi siswa Anda banyak latihan dalam membaca kosakata penglihatan dan menggunakan keterampilan decoding sangat penting untuk keberhasilan siswa Anda.**

**Ya ketergantungan yang berlebihan pada buku yang dapat didekodekan dapat membuat instruksi membaca menjadi kaku. Itu sebabnya instruksi membaca perlu menyertakan baca-aloud (untuk menyediakan model menarik dari tes otentik) beberapa buku yang mudah dibaca dan akhirnya buku yang dapat didekodekan yang membenamkan siswa Anda dalam bahasa yang dapat mereka akses.**

## easier to support remotelyProduct Scope

## Sistem perangkat lunak ini akan menjadi Sistem Penerbitan Web untuk editor lokal dari masyarakat sejarah regional. Sistem ini akan dirancang untuk memaksimalkan produktivitas editor dengan menyediakan alat untuk membantu mengotomatiskan tinjauan artikel dan proses penerbitan, yang seharusnya harus dilakukan secara manual. Dengan memaksimalkan efisiensi dan produksi editor, sistem akan memenuhi kebutuhan editor sambil tetap mudah dimengerti dan digunakan.

## Lebih khusus lagi, sistem ini dirancang untuk memungkinkan editor mengelola dan berkomunikasi dengan sekelompok pengulas dan penulis untuk menerbitkan artikel ke situs web publik. Perangkat lunak ini akan memfasilitasi komunikasi antara penulis, pengulas, dan editor melalui E-Mail. Formulir balasan yang telah diformat digunakan dalam setiap tahap kemajuan artikel melalui sistem untuk menyediakan proses peninjauan yang seragam; lokasi formulir ini dapat dikonfigurasi melalui opsi perawatan aplikasi. Sistem ini juga berisi basis data relasional yang berisi daftar Penulis, Peninjau, dan Artikel.

## References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

# Overall Description

## Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

## Product Functions

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

## User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

## Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

## Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>

## User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

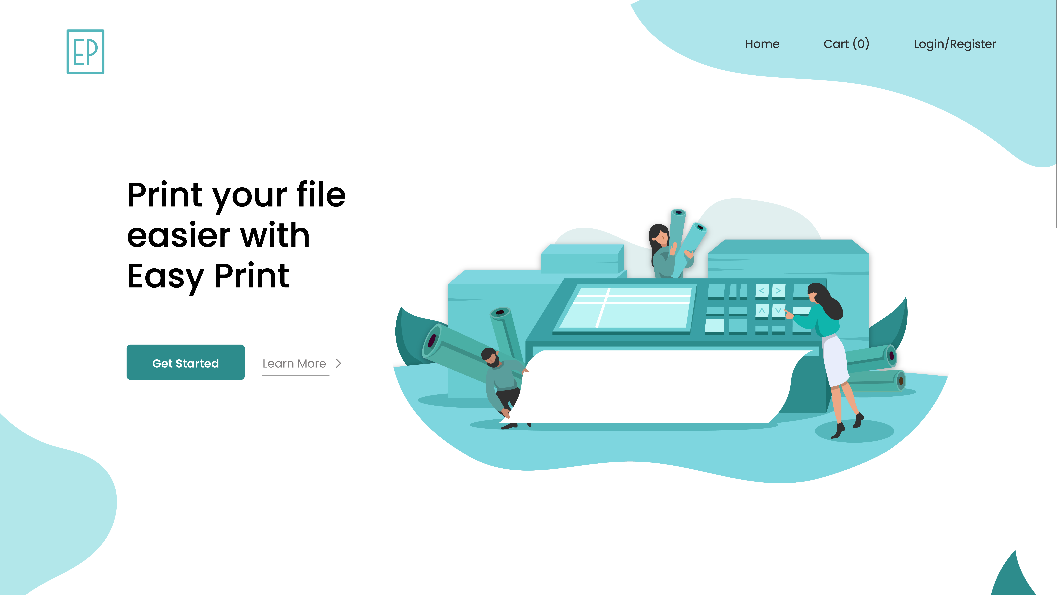
## Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

# External Interface Requirements

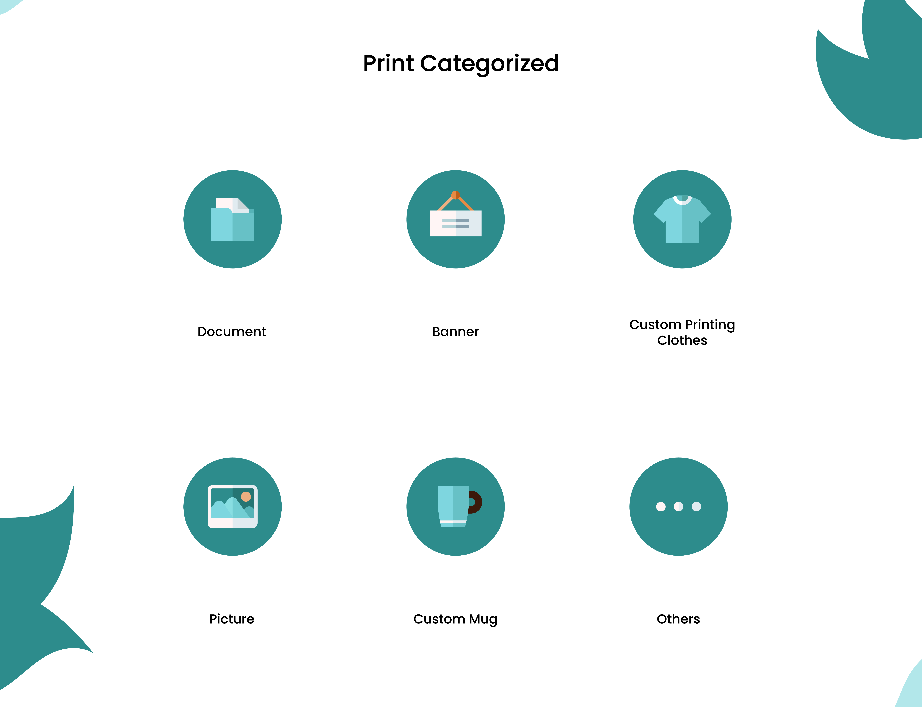
## User Interfaces

***Easy Print*** menggunakan antarmuka berbasis web. Desain dari antarmuka ini digunakan untuk mempermudah user dalam mencari atau melakukan antrian di jasa percetakan yang mereka inginkan dengan cepat dan efisien.

* Rancangan antarmuka Home website Easy Print
* Penjelasan tombol :
  1. Mari Mulai  
     Tombol yang mengarahkan langsung pada laman Print Categorized
  2. Pelajari Lagi

Tombol yang mengarahkan langsung kepada lama How to Order via Easy Print

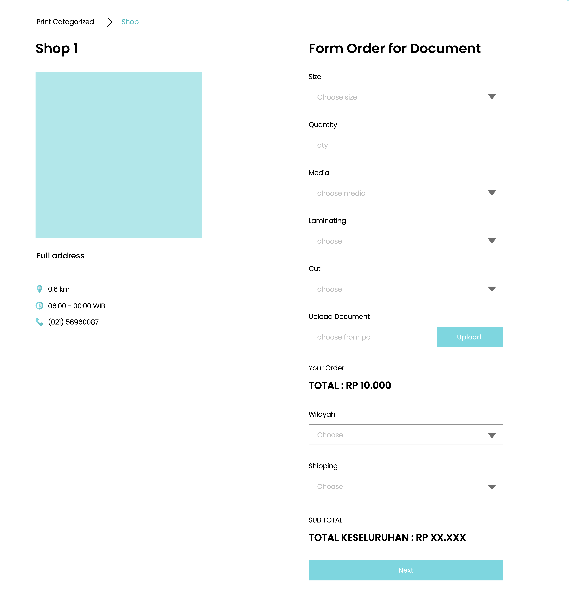
* Rancangan Tombol Print Categorized



* 1. Dokumen

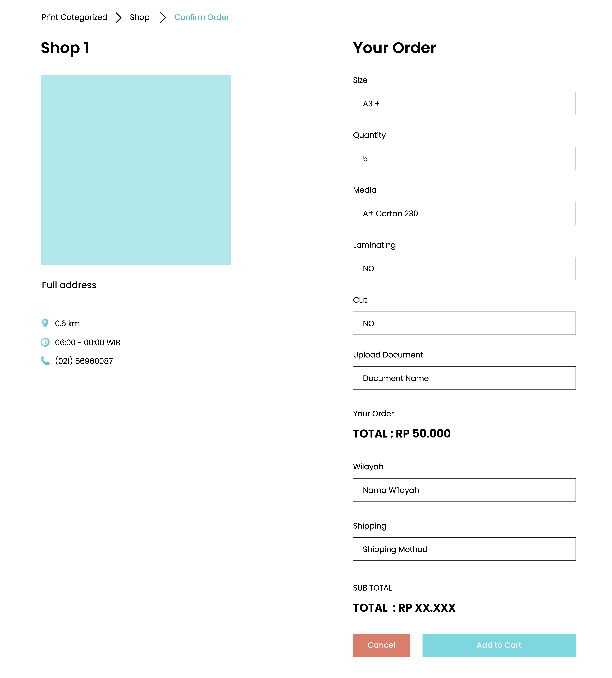
Digunakan untuk menampilkan jasa percetakan dokumen

* 1. Banner
  2. Digunakan untuk menampilkan jasa percetakan banner
  3. Custom Printing Cloth
  4. Digunakan untuk menampilkan jasa percetakan baju kustom
  5. Picture
  6. Digunakan untuk menampilkan jasa percetakan gambar
  7. Custom Mug
  8. Digunakan untuk menampilkan jasa percetakan mug
  9. Others
* Rancangan Tombol Selanjutnya pada Form Order



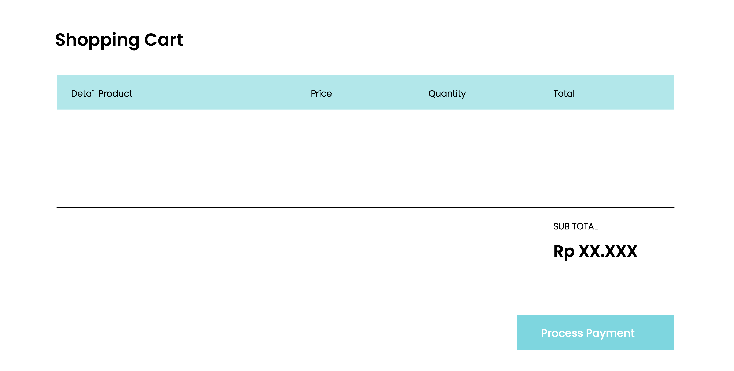
Digunakan untuk mengirimkan data order user kepada sistem

* Rancangan Tombol Masukkan ke Kerajang



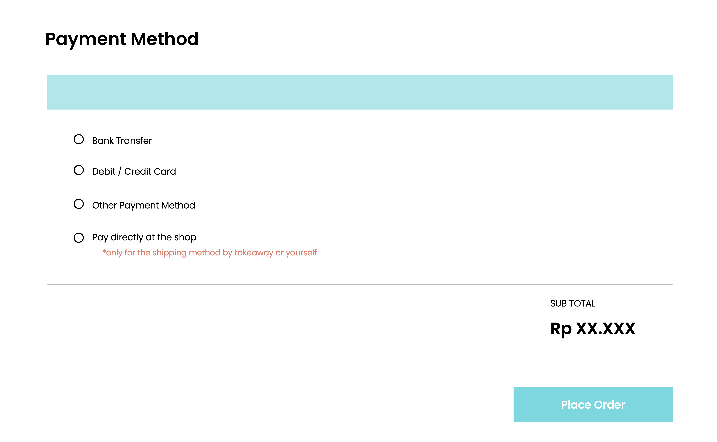
Digunakan untuk menyimpan data order user kepada system

* Rancangan Tombol Process Payment



Untuk meneruskan user kepada proses pembayaran

* Rancangan Tombol Place Order



Untuk menyimpan data pembayaran user ke pada system sehingga, order dapat diterus kan ke jasa cetak.

## Hardware Interfaces

* Personal Computer

1. Monitor

Wadah untuk menampilkan tampilan aplikasi kepada user.

1. Keyboard

Alat untuk mengetik/menginput data ke aplikasi

1. Mouse

Alat yang digunakan untuk men-submit data yang diisi oleh user pada aplikasi

1. Kabel USB/Flashdisk  
   Alat yang digunakan untuk mentransfer data perangkat keras lain ke personal computer
2. Kabel LAN

Menguhubungkan komputer ke jaringan internetworking.

## Software Interfaces

|  |  |  |
| --- | --- | --- |
| NO. | Jenis Software | Kebutuhan Hardware |
| 1. | Sistem Operasi |  |
| 2. | Bahasa Pemgrograman | HTML |
| 3. | Pemodelan Sistem | Adobe Illustrator |
| 4. | Perancang Antarmuka | Adobe Illustrator |

## Communications Interfaces

Aplikasi Easy Print ditujukan sebagai jembatan antara user dan jasa percetakan agar proses pemesanan jauh lebih mudah, bebas antrian dan juga terpercaya.

# System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

## System Feature 2 (and so on)

# Other Nonfunctional Requirements

## Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

## Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

## Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

## Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

# Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>