

# **IMPLEMENTASI SISTEM PENJUALAN BAJU BERBASIS WEBSITE MENGGUNAKAN METODE EXTREME PROGRAMMING DENGAN STUDI KASUS TOKO HARTINI**

**Muhammad Ersan Aldo Alvinetino<sup>\*1)</sup>, Mohammad Sholik, S.Kom., M.Kom<sup>2)</sup>, dan Fidi Wincoko  
Putro, S.St., M.Kom<sup>3)</sup>**

Prodi Rekayasa Perangkat Lunak, Fakultas Teknologi Informasi Dan Bisnis, Institut Teknologi Telkom  
Surabaya, Jl. Ketintang No.156, Gayungan, Surabaya, 60231, Indonesia  
Email : zeronep960@student.ittelkom-sby.ac.id

## **Abstrak**

Toko Hartini merupakan toko kecil perseorangan yang berbisnis berjualan baju yang berada di kecamatan buduran, kabupaten sidoarjo. Toko Hartini ini masih beroperasi secara offline. namun seiring jaman yang serba online ini toko hartini menjadi sepi pengunjung, maka dari itu penulis mengusulkan pengembangan aplikasi toko baju online yang bertujuan agar toko hartini dapat melakukan transaksi secara online. Agar toko hartini ini dapat melakukan transaksi secara online maka di perlukan implementasi aplikasi penjualan baju pada toko hartini berbasis *Website*. Berdasarkan kriteria tersebut penulis akan menggunakan metode pengembangan *Extreme Programming*. penerapan metode extreme programming pada penelitian ini akan menggunakan standar *12 practice extreme programming*. Hasil penelitian tersebut dapat disimpulkan penerapan *extreme programming* terdapat beberapa ketidaksesuaian dengan standar *12 practice extreme programming*

**Kata kunci:** *Extreme Programming, Website, Toko Hartini*

## **1. Pendahuluan (Introduction)**

Toko Hartini merupakan sebuah toko sederhana yang berbisnis menjual baju yang bertempat di Kecamatan Buduran, Kabupaten Sidoarjo. Toko Hartini ini merupakan milik toko milik keluarga yang sudah berdiri sejak tahun 1994. Toko Hartini saat ini hanya memiliki 1 toko saja yang beroperasi.

Namun Seiring perkembangan jaman, hampir semua jenis usaha sudah beroperasi online secara penuh dimana penjual dan pembeli berinteraksi melalui online, bisa melalui social media ataupun Platform Ecommerce. Kegiatan Jual Beli pada Toko Hartini yang ada saat ini masih sepenuhnya menggunakan buku catatan. Sistem Toko Hartini tersebut memiliki beberapa kekurangan. Kekurangan yang ditemukan oleh penulis adalah :

1. Sistem Transaksi saat ini masih menggunakan kertas bon dan buku catatan
2. Sistem Manajemen Barang saat ini masih menggunakan buku catatan
3. Sistem ini tidak bisa menjangkau banyak orang

Dari kekurangan tersebut, Toko Hartini membutuhkan sebuah sistem penjualan baju yang dapat melakukan transaksi secara online serta dapat memanajemen barang secara portable sehingga dapat membantu dalam melakukan input data barang ke dalam database dan dapat di jangkau oleh banyak orang. Sistem penjualan baju akan dikembangkan dengan berbasis website sehingga dapat digunakan melalui semua device seperti handphone ataupun laptop serta sistem ini akan menggunakan metode pengembangan extreme programming dimana sangat cocok untuk aplikasi yang sering berubah seiring nya waktu.

Hasil yang diharapkan dari implementasi aplikasi penjualan baju berbasis website ini adalah aplikasi mampu menangani kebutuhan pengguna dalam melakukan transaksi jual beli dengan pemilik Toko Hartini.

### 1.1. Rumusan Masalah

Berdasarkan latar belakang masalah tersebut maka dapat diperoleh rumusan masalah sebagai berikut :

1. Bagaimana membangun sistem penjualan toko hartini dengan menggunakan metode extreme programming?

### 1.2. Tujuan Dan Manfaat

Adapun tujuan dari penelitian ini yaitu :

1. Membangun sistem penjualan toko hartini dengan menggunakan metode extreme programming.

Adapun manfaat dari penelitian ini yaitu :

1. Toko Hartini mampu melakukan manajemen data barang dengan efisien.
2. Toko Hartini mampu melakukan transaksi dengan pembelinya.

## 2. Metode Penelitian (Methods)

dalam pengerjaan penelitian ini, peneliti menggunakan beberapa tahap yaitu :

### 2.1. Extreme Programming

Tahapan Extreme programming terdiri dari 4 tahap (Pressman, 2012) yaitu :

#### 2.1.1. Planning (perencanaan)

Tahap Planning berfokus untuk mendapatkan gambaran fitur dan fungsi dari perangkat lunak yang akan dibangun. pada tahap ini ada beberapa aspek yaitu : users story, users acceptance criteria.

#### 2.1.2. Design (Desain)

Tahap Design berfokus dalam membangun perangkat lunak yang didasari dari kebutuhan klien yang sudah di dapat pada tahap sebelumnya. Dalam XP, proses design terjadi sebelum dan sesudah aktivitas coding berlangsung.

#### 2.1.3. Coding (Pengkodean)

Tahap Coding yaitu tahap dimana dalam pengembangan akan menerapkan point story ke dalam kode serta dilakukan testing apakah setiap fitur sesuai permintaan.

#### 2.1.4. Testing (Pengetesan)

Tahap Testing yaitu tahap yang berfokus pada pengujian system, yang pertama yaitu unit test dimana untuk menentukan apakah fitur berfungsi sebagaimana yang dimaksud. Jika semua tes berjalan maka pengkodean selesai lalu akan dilakukannya acceptance test. Pada tahap ini, aplikasi akan langsung diuji coba oleh client agar mendapat tanggapan langsung mengenai penerapan gambaran dan permintaan dari klien yang telah dilakukan sebelumnya apakah sudah sesuai atau belum.

### 2.2. 12 Core extreme programming

12 core extreme programming merupakan aturan khusus yang membedakannya dengan metode lain (Ariaji, Utami and Sunyoto, 2021). Adapun aturan yang terdapat pada extreme programming adalah sebagai berikut :

#### 2.2.1. The Planning Game

The Planning Game ialah karakteristik karakteristik pada extreme programming yang digunakan buat mendiskusikan serta menyetujui fitur fitur pada produk serta diakhir perencanaan akan menciptakan iterasi serta perilisan yang hendak tiba.

#### 2.2.2. Small Release

Small Release di dalam extreme programming bermaksud untuk meningkatkan produk lebih lanjut dengan membuat pembaruan kecil serta bertahap. Perilisan kecil membolehkan pengembang buat

kerap menerima umpan balik, mengetahui bug lebih dini, serta memantau metode kerja produk dalam penciptaan.

#### 2.2.3. Metaphor

Metaphor merupakan salah satu ciri dari extreme programming yang memiliki arti bahwa desain atau strukturnya dapat dipahami oleh orang baru, dimana berarti program harus di buat se jelas mungkin agar orang baru dapat mengerti dengan cepat.

#### 2.2.4. Simple Design

Simple Design adalah menerapkan desain sesimpel mungkin dan tidak ada kompleksitas di dalamnya. yang dimana tidak ada redundansi, duplikat, dan tidak berisi metode yang tidak kegunaannya.

#### 2.2.5. Test-Driven Development

Test-Driven Development adalah dimana akan di lakukan pengujian setiap saat dalam pengembangannya.

#### 2.2.6. Code Refactoring

Code Refactoring merupakan proses untuk memperbaiki kode yang bertujuan untuk menghilangkan redundansi, menghilangkan fungsi yang tidak perlu, meningkatkan kejelasan kode.

#### 2.2.7. Pair Programming

Pair Programming adalah dimana membutuhkan 2 programmer untuk bekerja sama dalam satu kode yang sama yang memiliki tugas masing-masing yaitu dimana programmer pertama melakukan penulisan kode sementara programmer kedua meninjau kode yang dilakukan oleh programmer pertama.

#### 2.2.8. Collective ownership

Collective Programming yang dimana semua seluruh tim desain sistem dapat meninjau dan memperbarui kode agar tidak terjadi kondisi dimana tim tidak mengetahui letak kode untuk menambahkan fitur baru.

#### 2.2.9. Continous integration

Continous Integration merupakan kondisi dimana tim pemrogram melakukan menulis kode beberapa kali dalam sehari. pemrogarm mendiskusikan bagian mana yang dapat digunakan kembali dan yang tidak dapat digunakan kembali sehingga seluruh tim tahu persis fungsi apa yang perlu dikembangkan.

#### 2.2.10. 40-Hour week

40-Hour Week adalah kondisi dimana jam kerja dalam seminggu adalah 40 jam seminggu. dilakukan kondisi ini agar seluruh tim sehat dan dapat beristirahat.

#### 2.2.11. On-Site Customer

On-Site Customer adalah dimana klien harus sepenuhnya berpartisipasi dalam pengembangan, lebih tepatnya klien harus hadir setiap saat pada saat tim membutuhkan jawaban sehingga menentukan prioritas pengerjaan.

#### 2.2.12. Coding Standards

Coding Standards merupakan aturan dalam format dan gaya untuk penulisan kode. aturan ini dibuat agar semua tim dapat membaca, berbagi, dan memfaktor ulang kode dengan mudah, melacak siapa yang mengerjakan potongan kode tertentu, serta membuat pembelajaran lebih cepat untuk programmer lain.

### 2.3. *Black Box Testing*

Metode Blackbox Testing merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di diharapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid (Anggana *et al.*, 2018).

### 3. Hasil dan Pembahasan (Results and Discussions)

#### 3.1. Hasil

Berikut akan dijelaskan hasil penelitian berdasarkan metode yang diambil yaitu Metode Extreme Programming.

##### 3.1.1. Iterasi 1

Pada Tahap Iterasi Pertama, Peneliti melakukan pengerjaan *user story* yang terdaftar dalam iterasi pertama yang terdapat pada *iteration planning*.

##### 3.1.1.1. Planning

Pada tahap *planning* ini akan dilakukan perencanaan yang telah didapat pada *user stories* pada tahap iterasi pertama ini. berikut merupakan tabel daftar fitur yang akan di kerjakan pada iterasi pertama ini :

Tabel **Error! No text of specified style in document..**1 Release Planning

No.Prioritas	Topik	Iterasi				
		Iterasi 1	Iterasi 2	Iterasi 3	Iterasi 4	Iterasi 5
1	<i>Register (3)</i>					
2	<i>Login (3)</i>					
3	<i>Forgot Password (3)</i>					
4	<i>Logout (2)</i>					
5	<i>Dashboard Admin (1)</i>					
6	<i>Add Item (2)</i>					
7	<i>Update Item (2)</i>					
8	<i>Delete item (2)</i>					
9	<i>View all order (3)</i>					
10	<i>Edit Status Order (2)</i>					
11	<i>Report Print (2)</i>					

No.Prioritas	Topik	Iterasi				
		Iterasi 1	Iterasi 2	Iterasi 3	Iterasi 4	Iterasi 5
12	<i>Change Password (3)</i>					
13	<i>Show list item (2)</i>					
14	<i>Searching item (3)</i>					
15	<i>Cart (4)</i>					
16	<i>Transaction (4)</i>					
17	<i>History (3)</i>					
18	<i>Profile (2)</i>					
19	<i>Home customer (2)</i>					

**Tabel 3.2 Iterasi 1**

No	ITERASI 1	BOBOT POINT
1	<i>Register</i>	3
2	<i>Login</i>	3
3	<i>Forgot Password</i>	3

Berikut ini merupakan daftar *users story* yang terdapat pada iterasi pertama yaitu :

1. *Register* : Pembeli dapat melakukan registrasi akun agar dapat masuk ke dalam Toko Hartini.

Tugas :

- 1) membuat halaman input untuk registrasi
- 2) menyimpan data registrasi

2. *Login* : Pembeli atau admin dapat melakukan login ke dalam Toko Hartini.

Tugas :

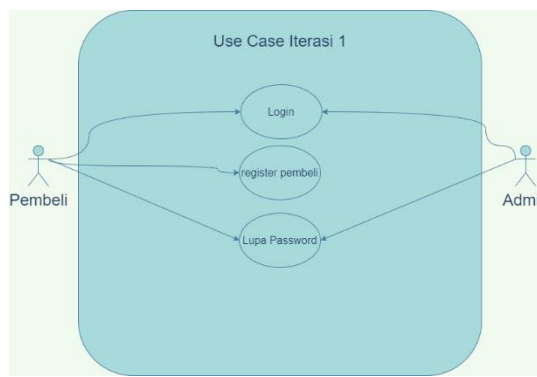
- 1) membuat halaman input login
- 2) memvalidasi data login
3. *Forgot Password* : Pembeli atau admin dapat melakukan lupa password jika lupa dengan password akun mereka.

Tugas :

- 1) membuat halaman input email untuk lupa password
- 2) mengirim pesan untuk lupa password ke email

### 3.1.1.2. Design

Pada tahap *Design* ini akan mendesain rancangan tampilan serta *Use Case* Iterasi 1. berikut merupakan rancangan tampilan serta *Use Case* Iterasi 1 :



Gambar 3.1 Use Case Iterasi 1

### 3.1.1.3. Coding

Pada tahap *Coding* ini akan mengimplementasikan desain yang telah dibuat ke dalam kode. berikut merupakan daftar *class* yang telah peneliti buat serta penjelasannya :

**Tabel 3.1 Coding Iterasi 1**

No	Nama Class	deskripsi
1	<i>Model Akun</i>	<i>Model akun</i> adalah sebuah <i>class model</i> dimana akan berfungsi menghubungkan ke database pada tabel akun
2	<i>Controller Register</i>	<i>Controller register</i> adalah sebuah <i>class Controller</i> yang berfungsi untuk memvalidasi serta menyimpan data-data register.
3	<i>Controller Login</i>	<i>Controller Login</i> adalah sebuah <i>class Controller</i> yang berfungsi untuk menyimpan serta memvalidasi data-data login.

No	Nama Class	deskripsi
4	<i>Controller Forgot Password</i>	<i>Controller Forgot Password</i> adalah sebuah <i>class Controller</i> yang berfungsi untuk mengirim <i>email</i> lupa <i>password</i> serta memvalidasi untuk menyimpan <i>password</i> baru.

#### 3.1.1.4. Testing

Tahapan *Testing* ini untuk menguji coba semua fitur yang telah di buat pada iterasi pertama. berikut merupakan hasil uji coba pada iterasi pertama yang menggunakan pengujian *black box* :

**Tabel 3.2 Pengujian Login Admin**

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengetikkan Email dan password (diisi), kemudian klik tombol Login	Username : zeronep960@gmail.com  Password :alvin	Sistem menerima akses login dan kemudian menampilkan halaman utama Admin	Sesuai harapan	Valid

**Tabel 3.3 Pengujian Login Customer**

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengetikkan Email dan password (diisi), kemudian klik tombol Login	Username : zeronep9600@gmail.com  Password : alvin	Sistem menerima akseslogin dan kemudian menampilkan halaman utama Admin	Sesuai harapan	Valid

**Tabel 3.4 Pengujian Register**

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Password, Email, Alamat, Nama dan Nomor Telepon diisi tetapi dengan Email yang belum terdaftar kemudian klik tombol Register	Nama: alvin  Alamat: sukodonoro  Nomor Telepon: 0881036370068  Email: zeronep96001@gmail.com	Sistem akan menyimpan data dan mengirimkannya kode verifikasi akun melalui email	Sesuai harapan	Valid



No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		Password :alvin			

**Tabel 3.5 Pengujian Halaman Verifikasi Email**

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	<i>Kode Verifikasi</i> tidak diisi kemudian klik tombol Verifikasi	Kode Verifikasi: (kosong)	Sistem akan menolak	Sesuai harapan	Valid
2	Klik tombol resend		Sistem akan mengirimkan kode verifikasi melalui email dan menampilkan pesan “berhasil mengirimkan kode verifikasi”	Sesuai harapan	Valid
3	Memasukkan kode verifikasi yang benar kemudian klik tombol Login	Kode Verifikasi: 2148	Sistem akan berhasil memverifikasi akun dan	Sesuai harapan	Valid

No.	Skenario Pengujian	<i>Test Case</i>	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
			menampilkan pesan “Berhasil Memverifikasi akun”		

**Tabel 3.6 Pengujian Halaman Lupa Password**

No.	Skenario Pengujian	<i>Test Case</i>	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	<i>Email</i> diisi dan email sudah terdaftar di dalam sistem kemudian klik tombol kirim	Email: zeronep960@gmail.com	Sistem akan mengirim email untuk mengubah password baru dan menampilkan pesan “Email untuk lupa password sudah terkirim”	Sesuai harapan	Valid

**Tabel 3.7 Pengujian Halaman Password Baru Untuk Lupa Password**

No.	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	<i>password</i> baru dan konfirmasi <i>password</i> baru diisi dengan isi yang sama kemudian klik tombol kirim	Password Baru: alvin1  Konfirmasi  Password Baru: alvin1	Sistem akan menyimpan password baru dan kembali ke halaman login	Sesuai harapan	Valid

### 3.2. Pembahasan

penerapan metode extreme programming pada penelitian ini menggunakan standar 12 core practice extreme programming. berikut merupakan hasil penerapan berdasarkan standar 12 core practice extreme programming :

#### 1. *The Planning Game*

Pada prakteknya, peneliti melakukan diskusi di awal dan di akhir iterasi untuk memastikan bahwa fitur sudah sesuai dengan permintaan. Dalam praktek The Planning Game ini sudah sesuai dengan standar extreme programming.

#### 2. *Small Release*

Pada prakteknya, peneliti tidak melakukan rilis secara berkala pada setiap akhir iterasi dan tetap menggunakan pengembangan lokal dalam melanjutkan pengembangannya lalu merilis ke production ketika sudah selesai keseluruhan program. Dalam praktek Small Release ini tidak sesuai dengan standar extreme programming.

#### 3. *Metaphor*

Pada prakteknya, peneliti melakukan penyederhanaan pada kode dengan menggunakan seperti menggunakan variabel yang cukup mudah untuk dipahami karena nama variabel sesuai dengan yang disimpan. Dalam praktek Metaphor ini sesuai dengan standar extreme programming.

#### 4. *Simple Design*

Pada prakteknya, peneliti menerapkan desain sesimpel mungkin dalam desain ui ataupun desain kode serta sudah tidak ada redundansi, duplikat fungsi ataupun berisi metode yang ada kegunaannya karena sudah menerapkan tahapan code refactoring. Dalam praktek Simple Design ini sesuai dengan standar extreme programming.

#### 5. *Test-Driven Development*

Pada prakteknya, peneliti menerapkan pengujian setiap saat dalam pengembangannya agar memastikan saat production tidak terjadi bug. Dalam praktek Test-Driven Development ini sesuai dengan standar extreme programming.

#### 6. *Code Refactoring*

Pada prakteknya, peneliti memperbaiki kode agar tidak redundansi serta sudah tidak ada sebuah fungsi yang tidak perlu dalam pengembangan. Dalam praktek Code Refactoring ini sesuai dengan standar extreme programming.

#### **7. Pair Programming**

Pada prakteknya, peneliti tidak menerapkan 2 programmer dalam penelitian ini dikarenakan ini dikerjakan sendiri oleh peneliti. Dalam praktek Pair Programming ini tidak sesuai dengan standar extreme programming.

#### **8. Collective ownership**

Pada prakteknya, peneliti tidak memiliki tim desain sistem maka tidak ada yang dapat meninjau dan memperbarui kode karena hanya peneliti sendiri yang mengetahui letak setiap kode. Dalam praktek Collective ownership ini tidak sesuai dengan standar extreme programming.

#### **9. Continous integration**

Pada prakteknya, peneliti menulis kode beberapa kali dalam sehari dan mengetahui setiap bagian yang dapat digunakan kembali dan yang tidak. Dalam praktek Continous integration ini sesuai dengan standar extreme programming.

#### **10. 40-Hour week**

Pada prakteknya, peneliti mengerjakan kurang lebih 6 jam per hari dalam seminggu penuh. Dalam praktek 40-Hour week ini sesuai dengan standar extreme programming.

#### **11. On-Site Customer**

Pada prakteknya, peneliti menerapkan kondisi dimana klien sepenuhnya berpartisipasi dalam pengembangan sistem ini agar klien dan peneliti tidak terjadi kondisi dimana salah paham dan agar sesuai permintaan. Dalam praktek On-Site Customer ini sesuai dengan standar extreme programming.

#### **12. Coding Standards**

Pada prakteknya, peneliti menggunakan konsep standarisasi dalam penulisan kode seperti menggunakan konsep oop dan menggunakan nama yang koheren sesuai dengan fungsinya sehingga dapat dengan mudah di pahami oleh orang lain dengan cepat. Dalam praktek Coding Standards ini sesuai dengan standar extreme programming.

Berdasarkan pada hasil implementasi metode extreme programming dalam pengembangan sistem penjualan baju pada Toko Hartini belum optimal. Perihal ini bisa dilihat dari 9 practice yang cocok dengan standar serta 3 practice yang tidak cocok dengan standarnya. Sebagian practice yang tidak cocok tersebut merupakan Small Release, pair programming, Collective ownership.

### **4. Kesimpulan (Conclusion)**

Berikut adalah kesimpulan dari pengerjaan jurnal ini:

1. Dalam membangun sistem penjualan toko hartini dengan menggunakan metode extreme programming, peneliti menggunakan 12 core practice extreme programming dan memiliki hasil dimana terdapat 9 core practice extreme programming yang berhasil di terapkan serta terdapat 3 core practice extreme programming yang belum sama sekali di terapkan. dikarenakan terdapat beberapa penyimpangan dalam membangun sistem , sehingga dapat disimpulkan penjualan toko hartini dengan menggunakan metode extreme programming masih belum maksimal.

### **Daftar Pustaka**

- Anggana, A.S. *et al.* (2018) 'Pengujian Black Box pada Aplikasi Admin PT. World Trans Berbasis Web Menggunakan Metode Equivalence Partitioning', *Informatika Universitas Pamulang*, 5(1), pp. 95–99.
- Ariaji, T., Utami, E. and Sunyoto, A. (2021) 'Evaluasi Sistem Informasi Yang Dikembangkan Dengan Metodologi Extreme Programming', *DASI*, 15(4), pp. 53–62.
- Pressman, R.S. (2012) *Rekayasa Perangkat Lunak ( Pendekatan Praktisi Edisi 7 ) Buku 1*. 7th edn. Edited by R.S. Pressman. Jakarta: Andi.