



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN

3.1 Objek Penelitian

PT Jasaku Adiguna Sejahtera Abadi merupakan perusahaan teknologi yang berfokus pada aplikasi *smartphone* yang bernama Jasaku ini didirikan pada 17 September 2015. Aplikasi Jasaku dirancang untuk pelaku penyedia jasa untuk berkembang dan memasarkan jasa yang disediakan secara langsung kepada pelaku yang membutuhkan jasa tersebut. Fungsi utama aplikasi Jasaku yaitu menghubungkan penyedia jasa dengan pelanggan yang membutuhkan jasa secara langsung atau real time dimana pelanggan dapat melihat profil perusahaan jasa, harga dan lokasi penyedia jasa tersebut.



Gambar 3.1 Logo aplikasi Jasaku

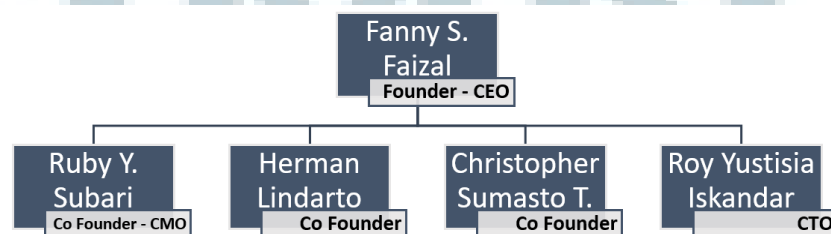
PT Jasaku Adiguna Sejahtera Abadi terus melakukan inovasi dengan terus menambahkan fitur-fitur baru untuk memenuhi kebutuhan pelanggan,

seperti contohnya fitur "Store", "Wallet", menyempurnakan fitur *cashless*, pelayanan *premium*, dan lain-lain.



Gambar 3.2 Tampilan setelah melakukan login pada aplikasi Jasaku

Pada Oktober 2016 Jasaku telah melakukan kerjasama dengan lebih dari 300 *vendor* yang telah tersebar di Jakarta, Bogor, Tangerang, Depok, Bekasi dan kini Jasaku terus melakukan ekspansi dengan target pasar secara nasional.



Gambar 3. 3 Struktur organisasi PT Jasaku Adiguna Sejahtera Abadi

Adapun visi dan misi PT Jasaku Adiguna Sejahtera Abadi yaitu:

1. Visi

Menjadi perusahaan yang efektif dan terdepan dalam menyediakan pelayanan jasa pada era digital.

2. Misi

1. Melayani dengan sepenuh hati dan memberikan pelayanan yang terbaik.

2. Memberdayakan penyedia layanan yang dapat membawa manfaat bagi masyarakat.

3. Inovasi dalam pengembangan teknologi dan berbagai layanan sesuai dengan kebutuhan pengguna dan untuk memfasilitasi penyedia layanan.

3.2 Metode Penelitian

3.2.1 Task-Centered System Design

Task-Centered System Design (TCSD) merupakan metode yang biasanya digunakan untuk mendeskripsikan kebutuhan serta pekerjaan penggunanya. Hasil deskripsi ini nantinya akan menjadi dasar dalam merancang antarmuka (UI) yang disertai dengan dukungan metode tambahan seperti observasi atau wawancara dengan pengguna. TCSD sendiri memiliki fokus utama yaitu perancangan konten serta tampilan informasi yang hendak dibuat. Dalam proses TCSD terdapat 4 Tahap, yaitu *identification*, *User-Centered*

Requirements Analysis, design as scenario, dan Evaluate via Task-Centered Walkthroughs.

1. Identification

Tahap pertama menentukan identifikasi permasalahan yang ada, penulis melakukan sebuah survei dengan kuesioner yang berasal dari jurnal John Brooke yang diunduh melalui website Usability.gov pada tanggal 25 Maret 2017. Dalam SUS - *A quick and dirty usability scale* milik John, terdapat 10 pertanyaan yang diterjemahkan ke dalam Bahasa Indonesia dengan 5 skala yang sudah ditentukan mulai dari sangat tidak puas hingga sangat puas.

Target kuesioner merupakan pengguna aplikasi dimana kuesioner nantinya akan disebar melalui *email* yang telah didapat dari pihak Jasaku. Setelah mendapat hasil kuesioner, data yang masuk diolah dan akan mendapat hasil penilaian akhir, kemudian dilakukanlah perancangan ulang dengan aplikasi Axure dan setelah selesai, akan disebarkan kembali kuesioner yang menilai prototip yang telah dirancang dengan prinsip dari *Google Material Design* dan langkah yang sama ketika hendak menyurvei tampilan sebelumnya. Langkah berikutnya apabila sudah menerima kuesioner yang menilai rancangan baru, data kemudian akan diolah untuk mendapatkan nilai akhir dan nantinya nilai akhir

ini-lah yang akan dibandingkan sehingga bisa dibuat sebuah kesimpulan.

2. *User-Centered Requirement Analysis*

Tahap berikutnya yaitu melakukan Analisa nilai rata-rata setiap pertanyaan berdasarkan hasil dari jawaban yang telah diberikan oleh responden atas survei yang telah diisi sebelumnya. Nilai yang dibawah rata-rata inilah yang nanti akan menjadi perhatian utama penulis untuk melanjutkan ke tahap berikutnya.

3. *Design Through Scenarios*

Tahap ketiga hal yang harus disiapkan oleh perancang yaitu:

a) *Desain flowchart*

Flowchart merupakan sebuah bagan yang menjelaskan alur tahap penyelesaian masalah hingga aliran data dengan gambar berupa simbol-simbol yang masing-masing diantaranya memiliki arti.

b) *Desain prototype*

Prototype atau purwarupa merupakan model awal sebuah entitas dimana model ini nantinya akan menjadi acuan sebuah produk yang akan dikembangkan atau diteliti. Dalam penelitian ini,

prototype dibuat dengan menggunakan *software* Axure RP berdasarkan prinsip *Google Material Design*.

4. Evaluate via Task-Centered Walkthroughs

Tahap ini merupakan tahap akhir dalam proses desain sistem, dimana penulis melakukan evaluasi final terhadap sistem yang telah dibuat dan memastikan semuanya proses di dalam sistem berjalan dengan seharusnya dengan cara melakukan survei untuk *prototype* yang telah dibuat kemudian dibandingkan dengan nilai aplikasi yang saat ini sudah ada.

3.2.2 8 Golden Rules

Perancangan *user interface* sendiri memiliki ketentuan, salah satunya yaitu 8 *Golden Rules* yang ditulis oleh Schneiderman:

1. *Strive for consistency*

Konsistensi dibutuhkan didalam sebuah aplikasi guna agar pengguna tetap mengenali halaman yang masih berada dalam satu lingkup sehingga akan membuat pengguna lebih nyaman dalam menggunakan aplikasi.

2. *Cater to universal usability*

Dalam merancang sebuah antarmuka, perancang harus mengenal tiap-tiap pengguna aplikasi yang bersangkutan, baik berdasarkan latar belakang, Bahasa hingga tingkat kefasihan

pengguna dalam memakai aplikasi. Penerapan *universal usability* bermanfaat untuk memudahkan pengguna awam hingga professional, dengan cara menerapkan sebuah *shortcut* suatu fungsi untuk pengguna yang sudah sering menggunakan aplikasi tersebut.

3. *Offer informative feedback*

Informative feedback tidak harus selalu dengan jawaban dari aplikasi ke pengguna, bisa saja berupa perubahan tampilan ketika melakukan aksi, sehingga pengguna mengerti jika perintah yang di-*input* sudah direspon oleh aplikasi.

4. *Design dialogs to yield closure*

Pada poin ini, pengguna diberikan informasi dan paham bahwa proses yang tadinya dijalankan oleh pengguna telah selesai sehingga tidak perlu menunggu apakah masih ada langkah lagi yang harus dijalani setelah menyelesaikan *task*-nya.

5. *Prevent errors*

Dalam perancangan antarmuka, pengguna dibuat sejauh mungkin melakukan kesalahan ketika sedang menggunakan sebuah aplikasi. Seperti contoh, didalam sebuah form registrasi dibuat sebuah petunjuk atau *clue* yang membuat pengguna dapat mengisi dengan tepat pada pengisian pertama.

6. *Permit easy reversal of actions*

Poin ini menjelaskan bahwa pengguna seharusnya diberikan sebuah fungsi 'pembatalan'. Contohnya, ketika pengguna melakukan transaksi pembelian dan kemudian berubah pikiran, pengguna dapat membatalkan transaksi karena pengguna tidak langsung diarahkan ke proses pembelian, serta adanya pilihan pembatalan transaksi sehingga pengguna dapat bereksplorasi dengan aplikasi lebih lanjut.

7. *Support internal locus of control*

Fungsi ini memberikan benefit kepada pengguna untuk melakukan *custom* terhadap tampilan sesuai keinginan, sehingga pengguna akan merasa lebih puas dan memberi nilai tambahan untuk UX aplikasi bersangkutan.

8. *Reduce short-term memory load*

Penerapan poin ini pada rancangan antarmuka berfungsi untuk membuat pengguna agar tidak perlu mengingat identitas apa yang harus dimasukkan kembali kedalam sistem, karena sistem sendiri telah secara otomatis mengingat dan mengisi data tersebut dengan sendirinya.