
Sistem Push Notification Life@BankJateng Sebagai Penunjang Layanan Pegawai PT. Bank Jawa Tengah Menggunakan Socket.IO

Push Notification System Life@BankJateng to Support PT. Bank Jawa Tengah Using Socket.IO

Sinar Setyawan¹, Ajib Susanto²

^{1,2}Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
Jl. Imam Bonjol No. 207, Tlp. (024) 3517261
E-mail: ¹sinarsetyawan0@gmail.com, ²ajib.susanto@dsn.dinus.ac.id

Abstrak

Sistem Push Notification Life@BankJateng merupakan sistem yang dirancang untuk melayani kepegawaian Bank Jawa Tengah. Sistem tersebut berupa aplikasi web sebagai admin dan aplikasi mobile sebagai client yang dapat memudahkan pertukaran informasi antar keduanya. Sistem Life@BankJateng dapat diakses menggunakan komputer, android dan IOS dan perangkat lain yang mendukungnya. fitur-fitur yang disediakan bertujuan untuk memudahkan pengelolaan informasi karir. Fitur yang disediakan meliputi pengaksesan informasi berita terbaru, akses profil tentang pencapaian jabatan kerja, pengelolaan cuti, fasilitas kesehatan dan surat pengantar perjalanan dinas (SPPD) dan lain lain. Sebagai tambahannya sistem tersebut berjalan secara realtime dengan memanfaatkan library Socket.IO yang dapat memberikan pemberitahuan pesan khusus pada setiap pegawai yang telah terdaftar di database sistem. Sistem ini telah teruji kualitasnya dengan menggunakan pengujian black box testing dan white box testing.

Kata kunci: Push Notification, Socket.IO, Realtime, web service, Mobile.

Abstract

Push notification system Life@BankJateng is a system designed to serve employee Central Java Bank. The system is a web application as admin and mobile application as a client that can facilitate the exchange of information between sectors. Life@BankJateng system can be accessed using computer, android and iOS and others that support it. features provided in formal to facilitate career information management. The features provided include accessing the latest news information, access profiles on job positions, information management, health facilities and official travel service (SPPD) mail and others. In addition, the system runs in realtime by utilizing the Socket.IO library that can provide a special message to every employee who has been registered in the system database. This system has been tested by using testing black box and white box testing.

Keywords: Push Notification, Socket.IO, Realtime, web service, Mobile.

1. PENDAHULUAN

Bank merupakan lembaga keuangan yang memiliki tugas pokok sebagai penyalur dana kepada masyarakat[1]. Saat ini banyak bank yang berdiri di Indonesia. Salah satu bank Badan Usaha Milik Pemerintah Daerah (BUMD) adalah Bank Jawa Tengah (Jateng). Bank Jateng pertama kali didirikan di kota Semarang dengan persetujuan dari Menteri Pemerintah Umum dan Otonomi Daerah No. DU 57/1/35 tanggal 13 Maret 1963 dan mendapat ijin usaha No. 4/Kep/MUBS/63 tanggal 14 Maret 1963 sebagai landasan operasional Jawa Tengah. Bank Jawa Tengah didirikan bertujuan untuk mengelola keuangan daerah yaitu membantu meningkatkan ekonomi daerah dengan memberikan pinjaman kepada pengusaha kecil dan sebagai pemegang

kas daerah[2]. Salah satu faktor perbankan menuju keberhasilan bank yang dapat dipercaya masyarakat tergantung pada kualitas kerja pegawai. Pegawai harus memiliki kemampuan mengembangkan diri secara proaktif pada suatu perusahaan. Untuk itu pegawai perlu diperhatikan dengan memberikan fasilitas layanan yang baik dan memadai.

Saat ini pegawai PT. Bank Jawa Tengah belum memiliki layanan khusus untuk menunjang kinerja pegawai seperti pengajuan cuti, Surat Pengantar Perjalanan Dinas (SPPD), Fasilitas Kesehatan (FasKes), dan Berita/ *News*. Sedangkan untuk mendapatkan respon cepat dari atasan secara langsung, perlu adanya suatu sistem yang dapat menangani permasalahan diatas. Sehingga dapat meningkatkan kinerja pegawai. Oleh karena itu untuk menunjang kebutuhan dan layanan pegawai yang dapat membantu menyusun informasi yang lebih terstruktur dan menjadikan layanan terkomputerisasi yaitu dengan sistem informasi.

Perkembangan teknologi yang sangat cepat berakibat pada sistem yang dibangun harus memiliki teknologi khusus yang harus ditanamkan, salah satu teknologi yang dibutuhkan yaitu dengan menerapkan informasi terbaru secara *realtime* atau biasa disebut *push notification*. *Push Notification*[3] memberi tahu pengguna dengan informasi baru tentang aplikasi, sehingga notifikasi yang didapat memberikan pembaruan informasi yang berharga dan relevan bahkan ketika aplikasi tidak berjalan

Untuk mendapatkan *realtime* notifikasi dengan menggunakan *Socket.IO*[4][5]. *Socket.IO* adalah paket perangkat lunak *Node.js* yang diimplementasikan pada sisi *server*, dan protokol *socket* web baru yang memungkinkan percobaan berkomunikasi dengan *Socket.IO* dibuat untuk *workstation* kontrol eksperimen [6]. *Socket.IO* dapat mengirim dan menerima permintaan dari *client-side* yang bisa akses secara bersamaan dan lebih stabil. Pada penelitian yang telah dilakukan oleh N. Wang, X. Chen, H. Parsaei dengan mengimplementasikan *Socket.IO* yang bertujuan untuk komunikasi waktu nyata antara aplikasi klien dan aplikasi server, protokol transmisi aplikasi berbasis *Socket.IO*. Selanjutnya, *Socket.IO* mendukung berbagai mekanisme lain (selain *socket TCP*) untuk mentransfer data, dan dalam kasus ini klien tidak akan (atau tidak bisa) mengirim pesan pengakhiran ke server[6]. Sedangkan pada penelitian lain membuat sistem dengan memanfaatkan *Node.js* menyatakan bahwa dengan menggunakan *Node.js* dan HTML 5 dapat diimplementasikan sebagai aplikasi web klien SIP sehingga *Node.js* dapat mengolah koordinat posisi dengan cepat menyesuaikan perubahan data dalam jeda waktu yang cukup singkat[7]. Implementasi *Socket.IO* dilakukan dalam penelitian untuk aplikasi lelang *online* di sosial media untuk menyelesaikan kasus *bid and run*[8] yang mampu menjembatani antara pelelang dan peserta lelang sehingga mampu meningkatkan kepercayaan pengguna.

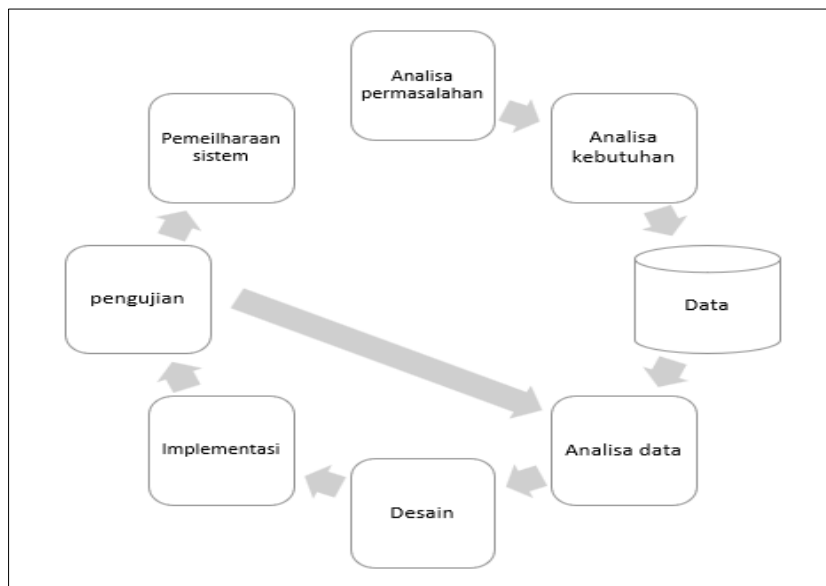
Sebagai upaya mengatasi permasalahan di manajemen Bank Jateng dibangun aplikasi yang bernama Life@BankJateng. Sistem Life@BankJateng yang dibagi menjadi *2platform* yaitu *mobile apps* dan *web service*. *Mobile apps* digunakan untuk pegawai *internal* Bank Jateng yang dapat mengakses sistem melalui *smartphone*-nya masing-masing sedangkan untuk pengelolaan sistem untuk memberikan informasi aplikasi *mobile* yaitu *web service*. *Web service*[9] merupakan sistem yang dirancang untuk mengatur komunikasi antar sistem melalui suatu jaringan berupa *HTTP, XML, SSL, SMTP, SOAP*, dan *JSON*[10]. *Web Service* telah menawarkan kinerja yang baik dan adanya integritas data[11]. Aplikasi *mobile* dan *web service* dapat memberikan pelayanan informasi secara *realtime* yang disediakan pada fitur notifikasi cuti, Fasilitas Kesehatan (FASKES), Surat Pengantar Perjalanan Dinas (SPPD), Penghasilan, dan Berita. Dengan kebutuhan sistem tersebut dapat mengurangi anggaran belanja perusahaan karena masih banyaknyapenggunaan alat tulis kantor. Sehingga dibutuhkan suatu Sistem *Push Notification* untuk Bank Jateng yang mengelola semua keperluan pegawai secara terstruktur didukung dengan pemanfaatan *Socket.IO* secara *realtime*.

2. METODE PENELITIAN

Teknik wawancara dilakukan untuk mendapatkan data yang *valid* dan bisa dipertanggung jawabkan sehingga tidak merugikan pihak lain jika terdapat salah informasi. Oleh karena itu penulis melakukan wawancara terhadap pihak yang berkaitan dengan pekerjaannya. Divisi

SDM dengan Pegawai IT merupakan pihak yang terkait dengan pengelolaan pembuatan sistem yang memberikan informasi berbagai permasalahan untuk membuat aplikasi pada PT. Bank Pembangunan Daerah Jawa Tengah. Metode pengembangan sistem ditunjukkan pada gambar 1. dimana dalam metode ini ada beberapa tahapan pengembangan sistem, yaitu :

1. Analisa permasalahan dimana proses ini digunakan untuk menganalisa masalah dan menghasilkan suatu solusi menjadi tujuan pada obyek penelitian.
2. Selanjutnya membuat analisa kebutuhan yaitu menganalisis kebutuhan dan mengumpulkan kebutuhan dengan berbagai sumber literatur agar menghasilkan sebuah data untuk mencapai tujuan awal tersebut.
3. Tahap berikutnya adalah analisis data dari data yang didapat pada tahap sebelumnya dimana data tersebut dilakukan analisis pada data agar data tersebut dapat diolah pada tahap selanjutnya yaitu tahap desain.
4. Tahap desain dimaksudkan untuk melakukan perancangan *database*, *interface* dan alur sistem.
5. Selanjutnya adalah tahap implementasi yang dilakukan dengan cara meng- implementasikan rancangan yang telah di bangun sebelumnya ke dalam bahasa pemrograman dengan menggunakan *Framework CodeIgniter*.
6. Setelah mengimplementasikan rancangan pada sistem, selanjutnya adalah tahap testing atau pengujian yang dilakukan pada sistem. Ketika pengujian dilakukan apabila terdapat sebuah kesalahan, maka dapat dilanjutkan dengan melakukan tahap pemeliharaan atau perbaikan.
7. Tahap perbaikan dilakukan ketika proses pengujian terjadi sebuah kesalahan. Maka tahap tersebut dapat memperbaiki sistem yang bermasalah.

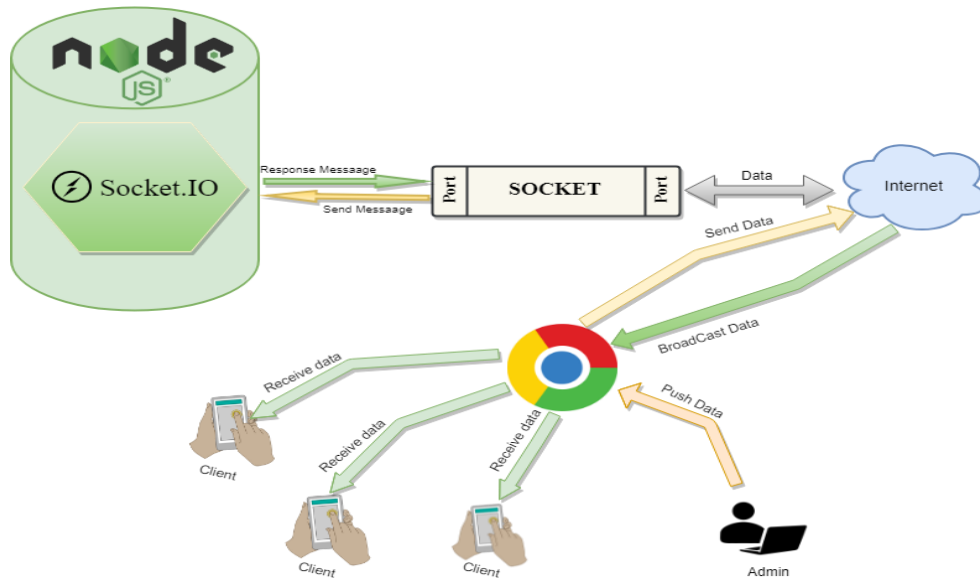


Gambar 1. Metode Pengembangan Sistem

3. HASIL DAN PEMBAHASAN

Untuk membuat aplikasi sistem manajemen kepegawaian Bank Jateng agar berjalan secara *realtime* dibutuhkan perancangan *Socket.IO* dan *Node.js* terhadap sistem. *Socket.IO* adalah sebuah *library javascript* untuk aplikasi web yang *realtime* yang ditujukan untuk setiap *browser* dan *mobile device*. Untuk mengatur jalannya *Socket.IO* dibutuhkannya *Node.js* sebagai server. Sedangkan sistem yang dibuat dibagi 2 hak akses yaitu akses sebagai user yang digunakan *client* dan admin yang di kelola oleh pegawai IT untuk mengelola sistem Life@BankJateng.

Menurut Ancona *Node.js* adalah *platform* aplikasi *JavaScript* sisi server yang didasarkan pada *non-event-driven* memblokir model *I/O*, dirancang untuk mendukung pengembangan *server* ringan yang mudah dan cepat untuk mengelola permintaan yang lebih besar [12]. Saat ini *Node.js* sudah banyak digunakan oleh para *developer* terutama *developer website* karena dapat memudahkan untuk mengambil *source code* kontribusi yang sudah bertebaran di internet untuk diunduh dengan menuliskan *script* pada *command line (CMD)* dan proses komunikasi *client server* yang sangat cepat seperti menangani banyak permintaan yang dapat ditangani secara bersamaan.



Gambar 2. Perancangan Socket.IO

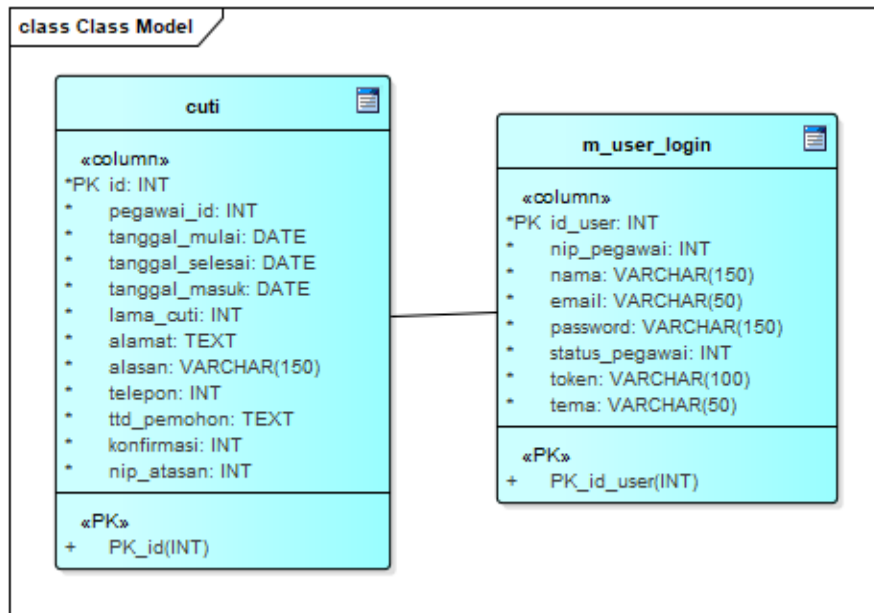
Pada gambar 2 dijelaskan tentang penggunaan *Socket.IO* yang bekerja dengan melibatkan komponen lainnya. Dibuktikan bahwa setiap pertukaran informasi harus melalui tahap tertentu yang harus dilalui. Salah satu contoh komponen yang melakukan interaksi/ pertukaran informasi yaitu hubungan antar admin dengan *client*. Pertukaran informasi diatas dimulai dari admin yang akan mengirim pesan menggunakan *browser chrome*. Lalu dari chrome harus memiliki sambungan internet sebagai jalur pengiriman datanya. Setelah itu internet menuju ke socket port yang dikenali sebagai alamat tujuan. Kemudian *socket port* menuju ke *socket server* yang berjalan diatas *Node.js* untuk menerima pesan secara *realtime* dan pesan diteruskan kembali melalui alamat *socket port* yang dituju. Pengiriman diteruskan melalui jalur internet yang menuju kembali ke *chrome* sebagai media penggunaanya. Kemudian *chrome* mengirim ke *smartphoneclient* untuk menerima pesan dari admin. Dari beberapa tahapan diatas penulis menjelaskan *code* yang digunakan sebagai pengirim dan penerima dengan melampirkan *code Socket.IO* sebagai berikut :

Tabel 1. Code untuk pengirim dan penerima

No	Source Code	Keterangan
1	<code>var socket = io.connect('alamat tujuan');</code>	Langkah pertama digunakan untuk mendaklarasikan alamat ip dan port yang dituju
2	<code>socket.emit(nama_event',{atribut data bentuk array});</code>	Langkah kedua dikenali sebagai pengiriman data
3	<code>Socket.on(nama_event',function(data){deklarasi isi data });</code>	Langkah ke 3 berguna untuk menerima pesan dari beberapa event pengirim

Dari proses yang telah dilalui maka dapat disimpulkan bahwa *Socket.IO* terbukti efektif dan cepat dalam melakukan pertukaran data secara *realtime*.

Perancangan *data modeling* bertujuan untuk menggambarkan konsep relasi antar entitas yang dapat dibentuk menjadi struktur data sebuah sistem yang jelas yang dapat membantu memudahkan pengembangan atau pembuatan sistem. Perancangan *data modeling* yang dibuat sebatas entitas yang memiliki relasi saja. Berikut rancangan *data modeling* dari sistem :



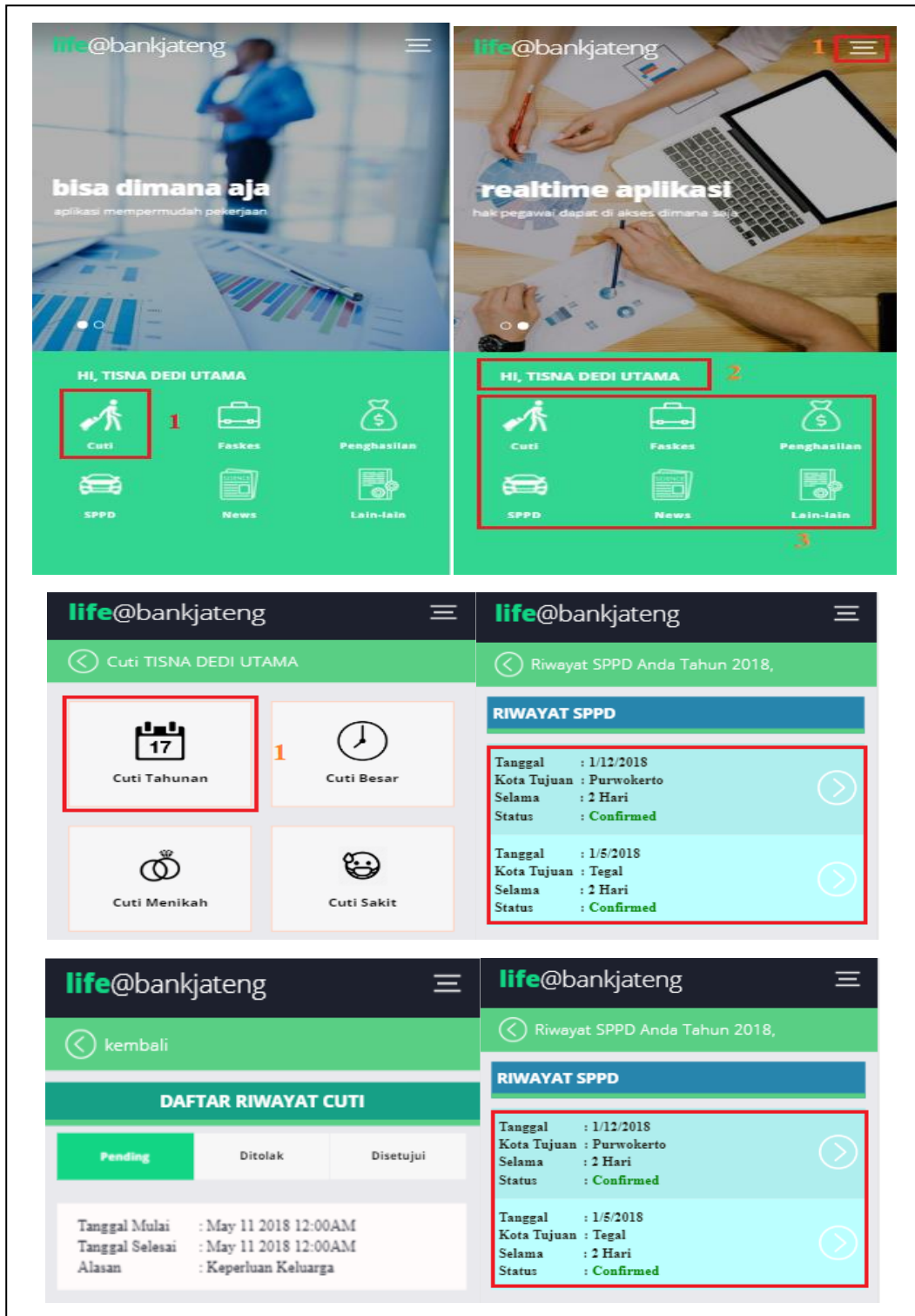
Gambar 3. Perancangan *Data Modeling*

Pada gambar 3 di atas menjelaskan hubungan data antara cuti dan pegawai yang sedang masuk ke sistem. Pengguna dapat melakukan pengajuan cuti yang nantinya data pengajuan juga akan dibaca oleh atasan sampai notifikasi yang diberikan diterima oleh pegawai.

Pada sistem ini, user berperan sebagai pengguna yang mengelola informasi pada sistem sisi *client* untuk menerima informasi tentang karir, riwayat dan pengajuan cuti. Sistem ini hanya bisa digunakan oleh pegawai PT. Bank Pembangunan Daerah Jawa Tengah. Penggunaan aplikasi sebagai *client* dapat diakses dengan cepat, hemat dan bisa diakses dimana saja. Aplikasi untuk client adalah sistem *mobile apps* supaya memudahkan penggunaannya

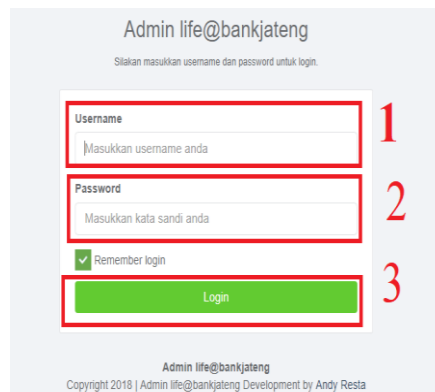
Gambar 4 menunjukkan tampilan layar bagi *user* sebagai *client*. Berikut keterangan dari gambar yang menunjukkan urutan langkah pada menu utama:

1. Pada masing-masing ikon (diwakili dengan ikon 'Cuti' yang diberi tanda kotak), digunakan untuk menampilkan menu *side bar*.
2. Pada bagian ini (didalam kotak), menunjukkan nama user yang telah *login*.
3. Ikon-ikon dalam kotak nomor 3 merupakan daftar fitur menu utama yang di *display*.



Gambar 4. Tampilan User sebagai client

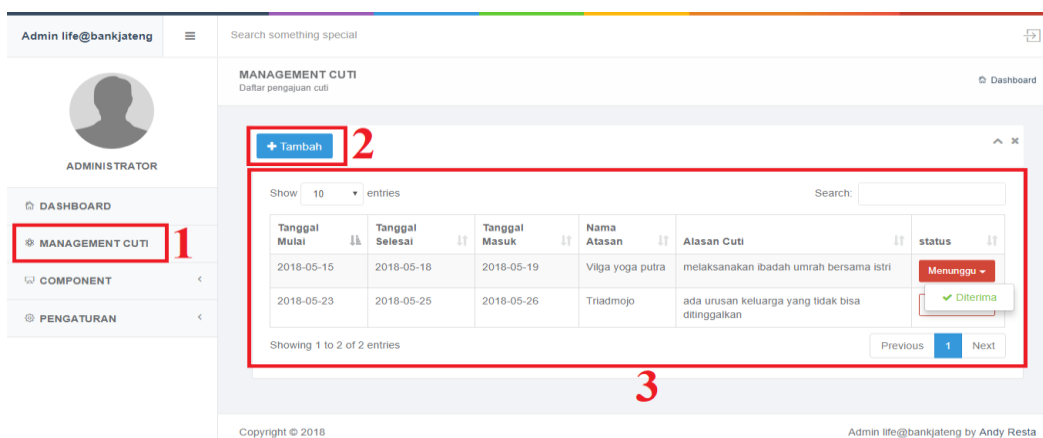
Sedangkan implementasi sistem dimana user sebagai *admin* yang berperan untuk mengatur pengelolaan sistem manajemen pada *backend* website Life@bankJateng, terdapat menu login seperti ditunjukkan pada gambar 5. Untuk masuk pada sistem Life@BankJateng, *admin* harus mengisi *username* pada kolom 1 dan *password* pada kolom 2. Setelah mengisi *username* dan *password*, klik *login* yang ditunjukkan pada kolom 3.



Gambar 5. Tampilan Login Admin

Gambar 6 merupakan menu *dashboard*, yang didalamnya terdapat beberapa menu seperti :

1. *Dashboard* digunakan untuk menampilkan konten *dashboard*.
2. Jumlah *user* adalah banyaknya jumlah pegawai yang telah terdaftar pada sistem Life@BankJateng.
3. Pengajuan cuti adalah banyaknya pegawai yang mengajukan cuti untuk selanjutnya dapat diproses oleh *admin*.

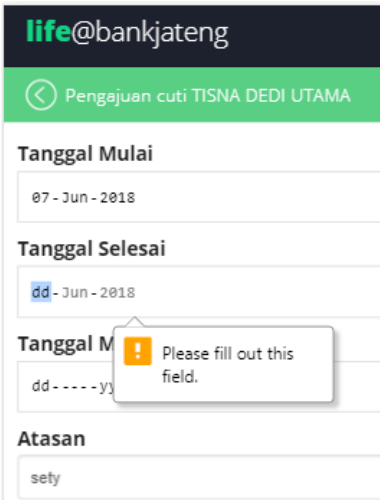
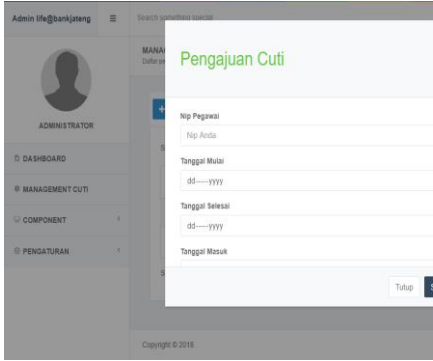
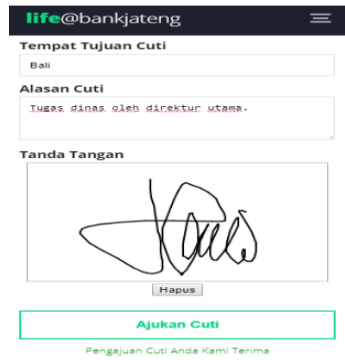
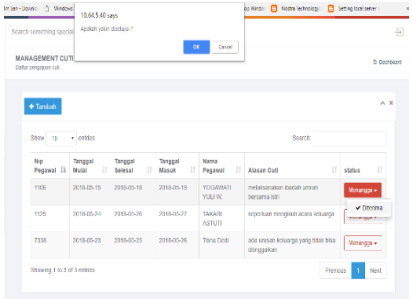
Gambar 6 : Menu *Dashboard* adminGambar 7. Tampilan *Admin* Pengelola Sistem

Pada gambar 7 menjelaskan salah satu pengelolaan yang diperankan *admin* yaitu menu cuti, berikut tahapan dari penggunaan menu cuti:

1. Manajemen cuti digunakan untuk menampilkan konten dari Manajemen cuti
2. Tombol Tambah berguna untuk menambahkan pengajuan cuti manual oleh admin. Berikut adalah tampilan dari tombol Tambah yang berisi Nip Pegawai, Tanggal Mulai, Tanggal Selesai dan Tanggal masuk dll. Lalu setelah selesai mengisi semuanya klik Simpan.
3. Berisi informasi pengajuan cuti oleh pegawai. Pengajuan cuti pegawai dapat ditindak lanjuti dengan mengubah status menunggu menjadi diterima oleh admin yang berarti pengajuan cuti dari pegawai telah disetujui.

Untuk menguji kelayakan sistem life@bankjateng menerapkan pengujian *black box* dan *white box*. Pengujian *black box* adalah pengujian yang memeriksa fungsional *software* dan dapat diamati melalui data uji yang telah dieksekusi. Sedangkan *white box* di gunakan untuk mendeteksi kesalahan logika sistem seperti *looping* dan kondisi perhitungan yang membuat sistem dapat bekerja penuh. Menguji perangkat lunak dari segi tampilan dan *syntax* program apakah mampu menghasilkan alur prosedural, masukan dan keluaran yang sesuai dengan rancangannya. Sehingga dengan pengujian keduanya maka sistem layak digunakan dan berkualitas. Berikut adalah pengujian untuk cuti pegawai oleh pegawai sendiri dan pengujian pengajuan cuti manual yang dilakukan oleh admin.

Tabel 2. Pengujian *Black Box* Sistem

Pengujian Pengajuan Cuti		Pengujian cuti manual oleh Admin	
Kondisi Awal	Mengisi formulir cuti	Kondisi Awal	Admin login kemudian masuk ke manajemen cuti, kemudian meng-klik Tambah
			
Proses	User menekan Ajukan Cuti	Proses	User menekan tombol Simpan
Keluaran	Sistem akan memberikan respon pengajuan cuti user	Keluaran	Halaman admin ditampilkan daftar pengajuan cuti dan status pengajuan dapat diubah.
			
Hasil	Sesuai	Hasil	Sesuai

4. KESIMPULAN

Berdasarkan pembahasan yang telah dilakukan dalam menggunakan *Socket.IO* pada sistem manajemen Life@BankJateng dapat ditarik kesimpulan bahwa:

1. Hasil pengujian *black box* bahwa sistem telah berjalan sesuai dengan rancangan sehingga sistem layak digunakan pada pengoperasian *Sistem Push Notification* Life@BankJateng.
2. Pemanfaatan sistem *pushnotification* yang diterapkan pada Life@BankJateng menggunakan *Socket.IO* dengan dibantu oleh *Node.js* sebagai sisi server dapat diterapkan dengan baik sebagai sistem *pushnotification* dapat berjalan secara otomatis dan realtime baik sistem dalam keadaan *online* atau *offline*.

DAFTAR PUSTAKA

- [1] W. R. Santosa, B. Jeni Susyanti, SE., MM., and M. Afi Rachmat Slamet, SE., "Aanalisis Kinerja Keuangan Perbankan Ssebelum dan Selama Menggunakan SMS Banking (Studi Kasus pada PT Bank Rakyat Indonesia (Persero) Tbk.)," pp. 1–14, 2015.
- [2] B. Jateng, "Sejarah Didirikan Bank PT. Bank Pembangunan Daerah Jawa Tengah," *Bank Jateng*, 2018. [Online]. Available: <http://bankjateng.co.id/tentang-kami/profil/tentang-perusahaan/>. [Accessed: 07-Nov-2018].
- [3] E. Isikligil, S. Samakay, and D. Kılınc, "A Prototype Framework for High Performance Push Notifications," *Int. J. Comput. Appl.*, vol. 166, no. 10, pp. 8–11, 2017.
- [4] Socket.IO, "What Socket.IO is." .
- [5] R. Somya, "Perancangan Aplikasi Chatting Berbasis Web di PT . Pura Barutama Kudus Menggunakan Socket . IO dan Framework Foundation," *khazanah Inform.*, vol. 4, no. 1, pp. 8–15, 2018.
- [6] N. Wang, X. Chen, and H. Parsaei, "Remote Experiment Development Using an Improved Unified Framework," *Proc. World Conf. E-Learning Corp. Gov. Heal. High. Educ. 2014*, no. November 2016, pp. 2003–2010, 2014.
- [7] H. Studiawan, M. C.R., Iqbal, and M. Husni, "Implementasi Klien SIP Berbasis Web Menggunakan HTML5 dan Node.js," *J. Tek. Its*, vol. 1, pp. 242–245, 2012.
- [8] D. Arga Anggara and A. Susanto, "Lelang Online Secara Realtime Dengan Protokol Websocket Menggunakan Socket . IO," 2017.
- [9] Y. R. Mustofa and A. Susanto, "Rancang Bangun Prototype REST Web service Untuk Menghubungkan Sistem Pembayaran UDINUS Dengan Sistem Perbankan," *Tugas Akhir*, 2013.
- [10] F. N. Rofiq and A. Susanto, "Implementasi RESTful Web Service untuk Sistem Penghitungan Suara Secara Cepat pada Pilkada," *Eksplora Inform.*, pp. 159–168, 2017.
- [11] D. Raniastri and A. Susanto, "Implementasi Web Service Dengan Metode Representational State Transfer (Rest) Untuk Integrasi Data Transaksi Pada Dua Web Server Yang Berbeda," 2017.
- [12] D. Ancona, L. Franceschini, G. Delzanno, M. Leotta, M. Ribaud, and F. Ricca, "Towards Runtime Monitoring of Node.js and Its Application to the Internet of Things," *Electron. Proc. Theor. Comput. Sci.*, vol. 264, pp. 27–42, 2018.