

PENGEMBANGAN SISTEM INFORMASI BERBASIS WEB SEBAGAI PENGAWASAN PROSES PENYELESAIAN PERMOHONAN UPAYA HUKUM DI KANWIL DJP JATENG II

Muhammad Mu'tashim Billah; Devi Afriyantari Puspa Putri, S.Kom., M.Sc.
Program Studi Informatika, Fakultas Komunikasi dan Informatika,
Universitas Muhammadiyah Surakarta

Abstrak

Salah satu tugas Kantor Wilayah (Kanwil) Direktorat Jenderal Pajak (DJP) adalah melakukan proses penyelesaian permohonan upaya hukum yang diajukan wajib pajak sesuai dengan amanat dalam Undang-Undang Perpajakan. Kendala utama yang dihadapi kantor tersebut dalam menjalankan proses bisnis adalah sulit untuk memanajemen permohonan terutama dalam mengetahui proses perkembangan permohonan, kendala tersebut dapat menyebabkan beberapa permohonan yang tertinggal bahkan bahkan baru terlihat setelah mendekati waktu jatuh tempo. Maka untuk menunjang efektivitas dan efisiensi proses, dibuatlah sistem informasi berbasis web dengan bahasa pemrograman Python, *framework* Django, dan basis data PostgreSQL yang dapat memberikan informasi secara cepat dan akurat tentang perkembangan proses bisnis yang sedang dijalankan. Metode yang digunakan dalam pengembangan sistem informasi ini adalah pendekatan dengan model *waterfall*. Aplikasi diuji

menggunakan metode *black box testing* untuk pengujian fungsional dan *System Usability Scale* (SUS) untuk pengujian kegunaan pengguna. Hasil akhir penelitian ini merupakan sistem yang dapat dipantau kinerja para pegawai dan ketepatan waktu penyelesaian permohonan sehingga dapat menunjang kinerja organisasi dengan skor SUS yang didapatkan adalah 67,5.

Kata Kunci: black box testing, django, DJP, kanwil, python, sistem informasi, SUS, wajib pajak, waterfall.

Abstract

One of the duties of the Regional Office of the Direktorat Jendral Pajak (DJP) is to carry out the process of resolving requests for legal remedies submitted by taxpayers in accordance with the mandate in the Tax Law. The main obstacle faced by the office in carrying out business processes is that it is difficult to manage applications, especially in the process of revocation progress, this obstacle can identify several applications that are left behind and even only become visible after approaching the due date. So to support the effectiveness and efficiency of the process, a web-based information system was created with the Python programming language, the Django framework, and the PostgreSQL database which can provide fast and accurate information about the development of the business processes that are being carried out. The method used in the development of this information system is the waterfall model approach. The application is tested using the black box testing method for functional testing and the System Usability Scale (SUS) for testing user usability. The final result of this study is a system that can monitor the performance of employees and timely completion of requests so that it can support organizational performance with the SUS score obtained being 67.5.

Keywords: black box testing, django, DJP, information system, kanwil, python, SUS, taxpayers, waterfall.

1. PENDAHULUAN

Dampak teknologi informasi telah membawa perubahan dan perkembangan dari berbagai bidang (Puspaningrum & Sudarmilah, 2020). Salah satu dampaknya adalah penggunaan sistem informasi

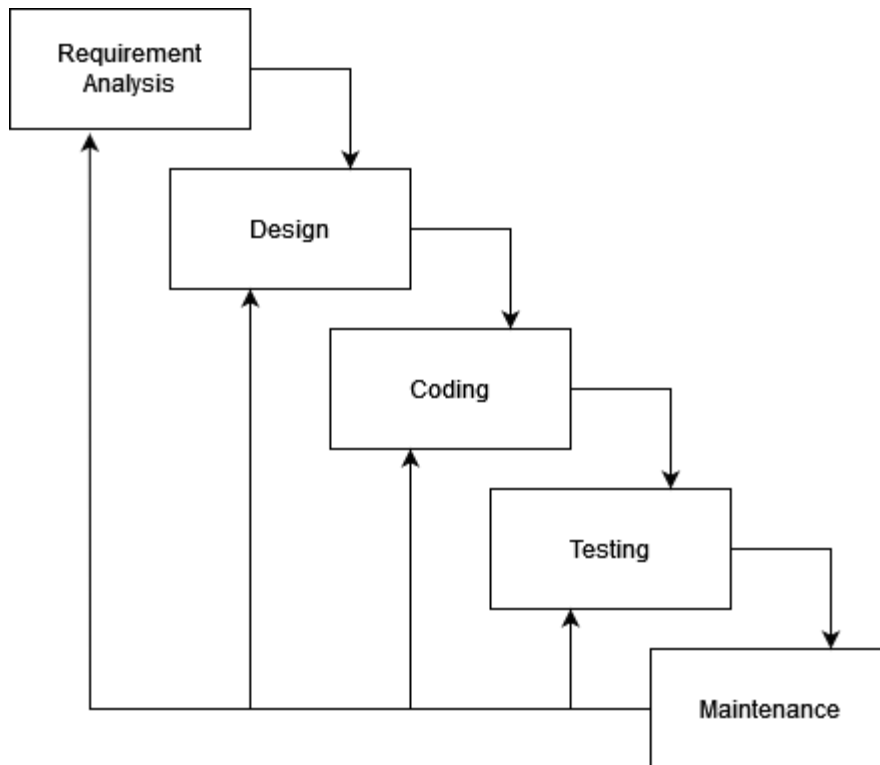
dalam kegiatan kantor. Sistem informasi yang baik sangat berpengaruh dalam kinerja yang efektif dan efisien (Ariawan & Ariyanto, 2018). Berdasarkan penelitian yang dilakukan oleh (Susanto et al., 2021), “Sistem Aplikasi Pengolahan Surat Masuk Keluar Dan Sppd di Kelurahan Jatijajar dapat mempermudah staff dalam mengelola surat, mengefisiensikan waktu serta mempermudah kinerja staff, meminimalisir kesalahan pencatatan, dan data yang disimpan lebih terstruktur dan lebih aman”. Menurut penelitian (Melliana & Nurgiyatna, 2021), dalam pembuatan Sistem Informasi Arsip Surat, “Sistem informasi ini dapat membantu aktivitas pada arsip surat-menyurat terarsip dengan baik, dapat melakukan cetak laporan berdasarkan tanggal dan kata kunci yang dimasukkan dan dapat melakukan pencarian data sewaktu-waktu dengan cepat dan tepat”.

Kanwil DJP Jawa Tengah II merupakan salah satu unit vertikal di DJP dengan wilayah kerja daerah administrasi sebagian Provinsi Jawa Tengah yang membawahi 12 Kantor Pelayanan Pajak (KPP) (Menteri Keuangan Republik Indonesia, 2020). Salah satu tugas dari kanwil tersebut adalah untuk menyelesaikan permohonan upaya hukum dari wajib pajak yang berada di area wilayah kerja yang telah ditetapkan. Setiap permohonan tersebut memiliki jenis kasus yang berbeda serta setiap kasusnya memiliki prosedur penyelesaian dan waktu jatuh tempo yang berbeda. Dalam penyelesaian permohonan, kanwil telah menggunakan sistem informasi berbasis web yang telah disediakan dari pusat. Namun, sistem tersebut belum dapat memenuhi semua kebutuhan kanwil dalam menyelesaikan permohonan seperti pelacakan proses permohonan dan informasi permohonan yang telah mendekati jatuh tempo.

Untuk menyelesaikan masalah tersebut, penulis diminta oleh kanwil untuk membuat sistem informasi berbasis web yang dapat membantu dalam mengelola permohonan dan memantau proses perkembangan permohonan di Kanwil DJP Jawa Tengah II. Diharapkan dengan adanya sistem informasi tersebut pegawai kanwil tersebut dapat melacak semua proses permohonan dan mengetahui informasi jatuh tempo permohonan dengan cepat dan akurat.

2. METODE

Metode yang dipilih penulis dalam pengembangan sistem informasi adalah metode *waterfal*. Penulis menggunakan metode *waterfall* karena dalam lapangan proses yang dilakukan dalam mengembangkan sistem informasi menggunakan metode *waterfall*. Model *waterfall* merupakan yang tertua dan yang paling dikenal dalam *Software Development Life Cycle (SDLC)*. Model ini banyak digunakan pada proyek pemerintahan dan perusahaan besar (Adel & Abdullah, 2015). Model *waterfall* yang digunakan penulis diperlihatkan pada gambar 1.



Gambar 1. Model *waterfall*

2.1 Requirement Analysis

Dalam tahapan ini penulis melakukan wawancara kepada pelaksana, penelaah keberatan, dan kepala bidang hingga penulis dan kanwil telah sepakat dalam menentukan kebutuhan yang diperlukan dalam sistem yang akan dikembangkan.

2.1.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional dalam sistem yang akan dikembangkan yaitu:

- Sistem dapat mendata dan memproses permohonan sesuai dengan alur *activity diagram*
- Sistem dapat menampilkan permohonan yang telah mendekati waktu jatuh tempo
- Sistem memberikan laporan perkembangan permohonan pada *dashboard*
- Sistem dapat melacak kegiatan pegawai dalam memproses permohonan
- Pegawai dapat melakukan *import* data permohonan dalam format xls atau xlsx
- Pegawai dapat memproses permohonan secara kolektif untuk beberapa tahapan proses permohonan
- Pegawai *login* untuk masuk ke dalam halaman pengguna dan mendapatkan hak akses yang sesuai untuk memproses permohonan sehingga menu *sidebar* yang tampil sesuai dengan hak akses yang pegawai dapatkan
- Admin mendapatkan seluruh hak akses pada halaman pengguna dan django admin

2.1.2 Analisis Kebutuhan Non Fungsional

Kebutuhan non fungsional dalam mengembangkan sistem ini terdiri atas perangkat keras dan perangkat lunak.

a. Analisis kebutuhan perangkat keras

Kebutuhan perangkat keras dalam mengembangkan sistem ini adalah laptop i5 6200u, RAM 16 GB, SSD 256 GB.

b. Analisis kebutuhan perangkat lunak

Kebutuhan perangkat lunak dalam mengembangkan sistem ini sebagai berikut:

- Python 3.8
- Visual Studio Code
- PostgreSQL
- Git
- Github
- Mozilla Firefox
- Microsoft Edge (berbasis chromium)

2.2 Design

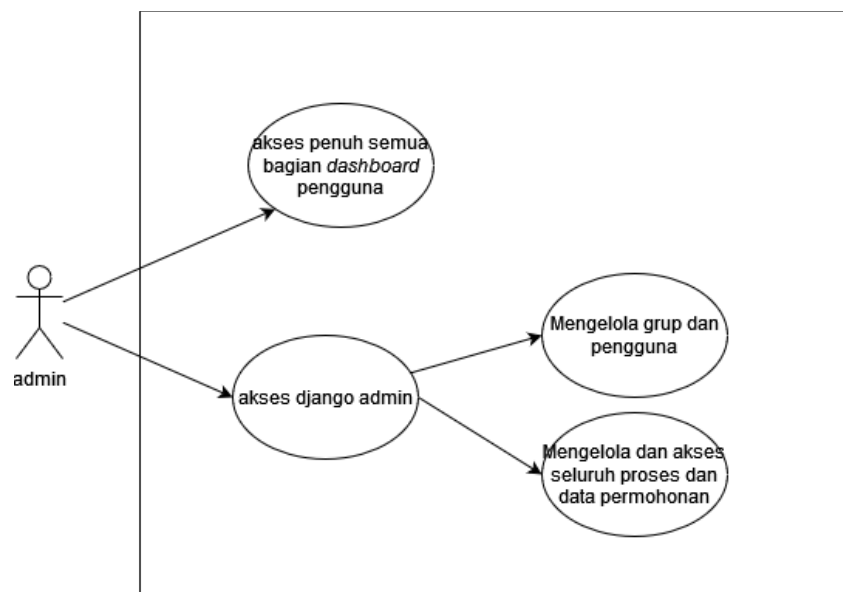
Pada tahapan ini, berdasarkan wawancara serta gambar diagram alur kerja dalam memproses permohonan yang diberikan perwakilan kanwil penulis serta menyepakati hasil *use case diagram* dan *activity diagram* menggunakan *Unified Modeling Language* (UML).

2.1.1 Use Case Diagram

Diagram yang menggambarkan aktor, *use case*, dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor (Haviluddin, 2011). Dalam memproses data permohonan aplikasi dibagi menjadi 2 bagian yakni bagian halaman pengguna dan django admin. Halaman pengguna merupakan halaman yang digunakan untuk memproses permohonan sesuai alur yang telah ditetapkan. Django admin merupakan aplikasi *user friendly* untuk mengelola konten basis data relasional yang ditautkan ke proyek Django (Rubio, 2017). Halaman pengguna akan tampil sesuai dengan tugas masing-masing pengguna, sedangkan admin mendapatkan seluruh akses serta django admin untuk kebutuhan darurat. Hasil dari *use case diagram* dapat ditampilkan pada gambar 2 dan gambar 3.



Gambar 2. Use case diagram pada dashboard pengguna

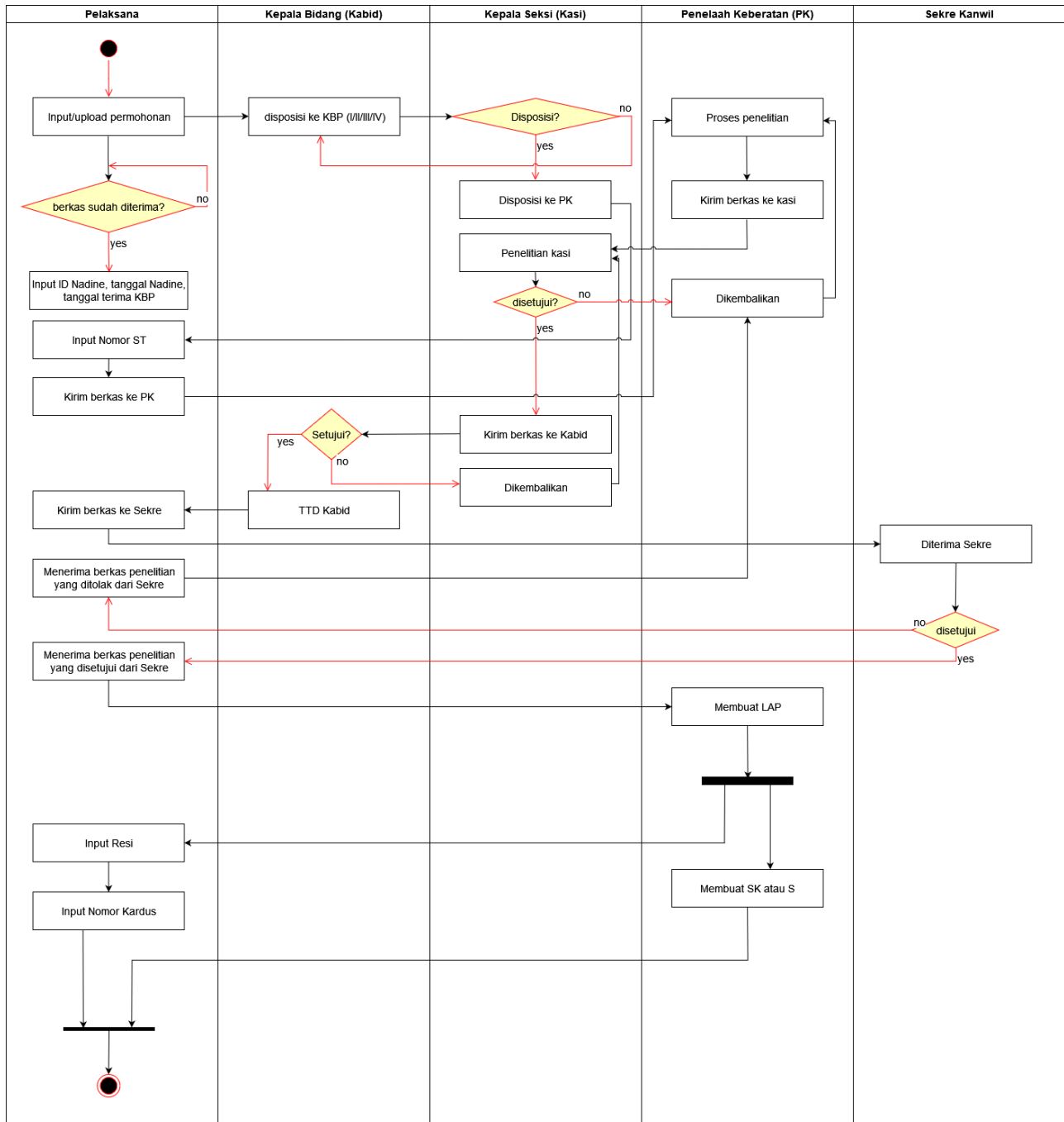


Gambar 3. Use case diagram pada admin

2.2.2 Activity Diagram

Activity diagram menggambarkan aktifitas-aktifitas, objek, state, transisi state, dan event (Haviluddin, 2011). Pada activity diagram yang dibuat penulis memperlihatkan alur kerja dalam

menyelesaikan permohonan beserta bagian yang bertugas atas proses tersebut dimulai dari masuknya permohonan baru hingga selesai yakni munculnya LAP, Surat atau Surat Keputusan, resi, dan nomor kardus. *Activity diagram* buat berdasarkan dengan kesepakatan akhir dengan kanwil. Proses tersebut dapat dilihat pada gambar 4.

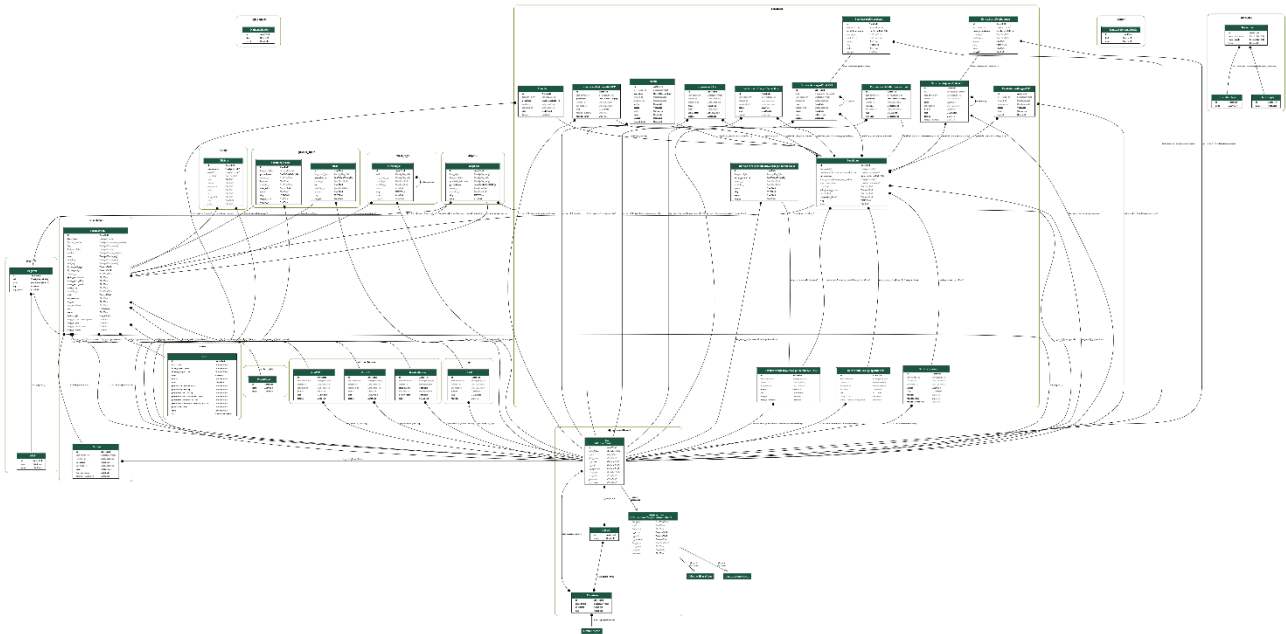


Gambar 4. *Activity diagram* penyelesaian permohonan

2.2.3 Database Relational

Entity Relationship Diagram (ERD) adalah cara grafis untuk memberikan informasi tentang *database* dalam sistem (Al-masree, 2015). Penulis melakukan desain ERD berdasarkan wawancara dari perwakilan kanwil kemudian melakukan implementasi pada *django model class*. Setelah *model class* sudah dirancang, dan migrasi *database* PostgreSQL telah dilakukan, aplikasi web yang

dikembangkan di Django dapat terhubung dengan *database*, *framework Object Relational Mapping* (ORM) dapat digunakan sebagai akses untuk melakukan operasi *create*, *read*, *update*, dan *delete* (Pérez et al., 2021). *Django* yang memproses *model* menjadi *query database* melalui Django ORM. Django ORM adalah seperangkat teknik khusus yang sangat kompleks yang tujuan utamanya untuk mempermudah pengembang (Kulanda Duisebekova, Roman Khabirov, 2021). Hasil *database model* dapat dilihat pada gambar 5.



Gambar 5. *Database model*

2.3 Implementation

Pengembangan sistem informasi ini dilakukan sesuai dengan analisis kebutuhan dan perancangan desain yang telah disusun. Pengembangan sistem menggunakan bahasa pemrograman *Python 3.8* dan *web framework Django 3.1*, selama pengembangan menggunakan *database SQLite* sedangkan saat penggunaan dalam produksi nanti menggunakan *database PostgreSQL*. *Front-end template* yang digunakan adalah *Gradient Able Django* dari Appseed. Adapun modul tambahan python yang dibutuhkan untuk mempercepat pengembangan aplikasi yaitu *django-extensions*, *django-autocomplete-light*, *django-import-export*, *django-tables2*, *django-filter*, *django-crispy-forms*, *django-widget-tweaks*, dan *django-annoying*. Untuk keamanan pengguna selain masuk menggunakan autentikasi kata sandi ditambahkan juga fitur autentikasi kedua berupa *time-based one-time password (TOTP)*. Kode TOTP pada sistem yang dibuat salah satunya menggunakan sebuah metode berbasis waktu (Permana et al., 2020). Dalam pengimplementasinya penulis menggunakan modul *django-two-factor-auth*.

2.4 Testing

Arti *software testing* sendiri yaitu sebuah metode yang berfungsi untuk mengetahui apakah perangkat lunak yang dirancang dapat berfungsi dengan sebagaimana mestinya dan sesuai dengan kebutuhan

pengguna (Amalia et al., 2021). Penulis menguji fungsi web yang telah dikembangkan menggunakan metode *black box testing*. *Black box testing* merupakan suatu bentuk pengujian aplikasi secara fungsional (Khafid & Putri, 2020). Pengujian dilakukan menggunakan 2 buah web *browser* yaitu Mozilla Firefox dan Microsoft Edge (berbasis chromium). Pengujian dengan 2 *browser* tersebut dipilih penulis karena Microsoft Edge (berbasis chromium) sudah cukup untuk mewakili semua aplikasi *browser* berbasis chromium lainnya seperti Google Chrome, Brave, dan Opera. Sedangkan Mozilla Firefox menggunakan *browser engine* yang berbeda dengan web *browser* berbasis chromium.

Selain itu ada pula pengujian *usability* yang melibatkan pengguna. Metode yang digunakan dalam menguji *usability* adalah metode SUS. SUS adalah kuisioner standar yang banyak digunakan untuk penilaian kegunaan yang dirasakan (Lewis, 2018).

2.5 Maintenance

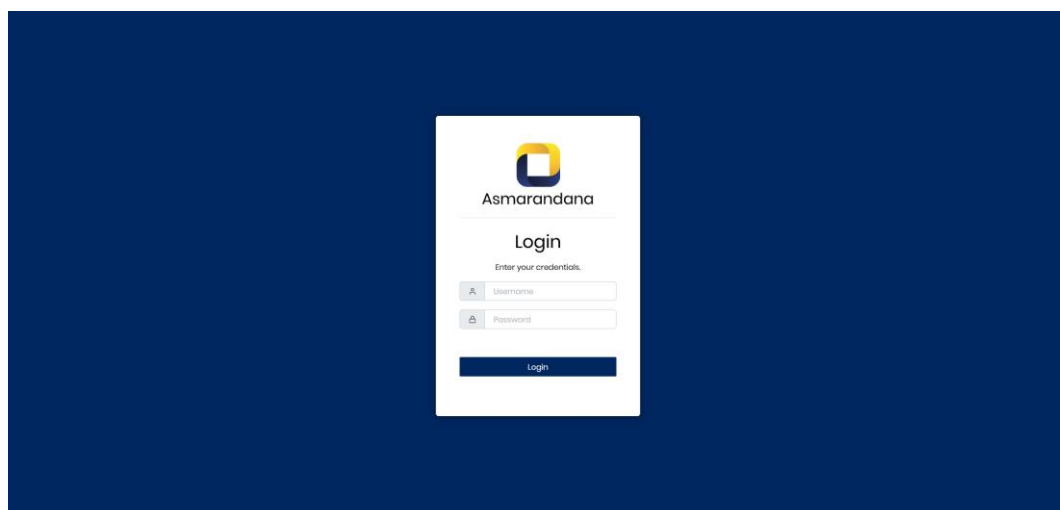
Sistem yang telah selesai akan dipasang pada *server* kantor dan digunakan oleh pegawai kanwil. Adapun *error* yang terjadi selama operasional sistem akan diperbaiki pada tahapan ini dan hasil perbaikannya akan diperbarui pada sistem yang berada pada *server*.

3. HASIL DAN PEMBAHASAN

Hasil penelitian ini adalah aplikasi berbasis web yang dibagi menjadi 2 bagian yaitu tampilan web yang digunakan untuk pengguna dan admin, serta django admin yang hanya dapat diakses oleh admin.

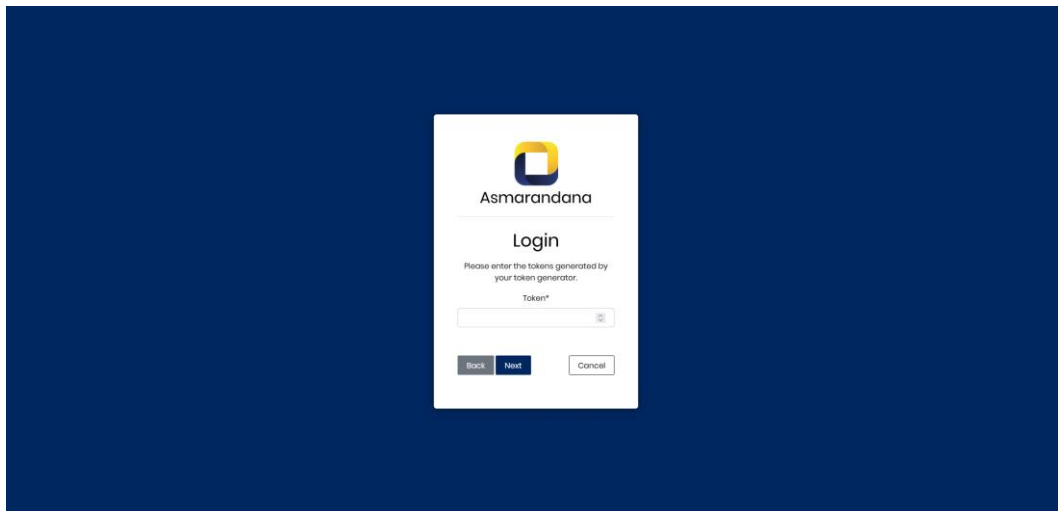
3.1 Hasil tampilan

Tampilan web diawali dengan halaman *login*, pengguna dan admin perlu memasukkan *username* dan *password* untuk dapat masuk. *username* yang digunakan untuk pengguna merupakan 9 digit Nomor Induk Pegawai (NIP) pendek sedangkan untuk admin tidak ada limitasi mengenai *username* yang dipakai. Halaman *login* ditampilkan pada gambar 6.



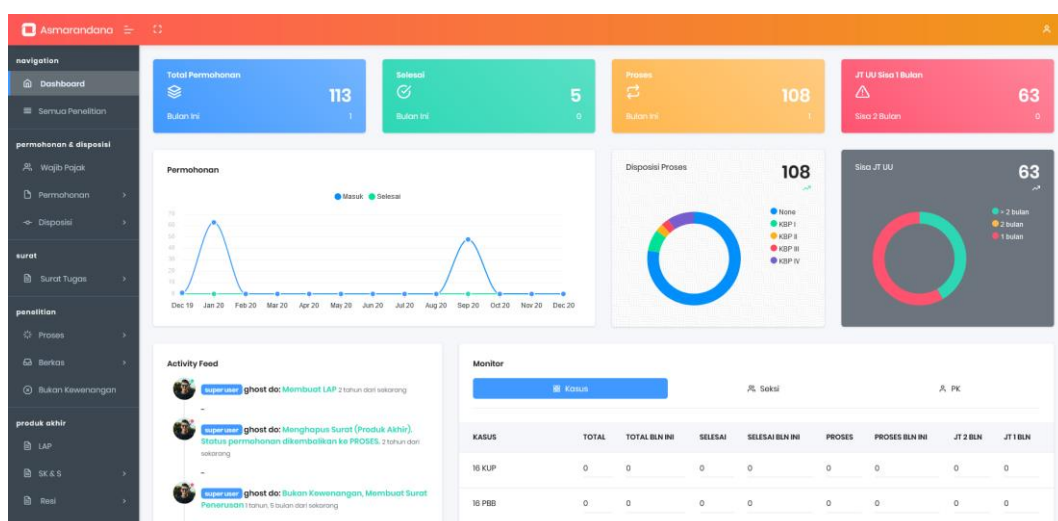
Gambar 6. Halaman *login*

Apabila *username* dan password yang diinputkan benar maka akan dilanjutkan pada halaman *dashboard*. Namun jika TOTP diaktifkan maka pengguna akan diminta untuk memasukkan kode TOTP sebelum lanjut ke halaman *dashboard*. Tampilan untuk memasukkan kode TOTP dapat dilihat pada gambar 7.



Gambar 7. Tampilan permintaan kode TOTP

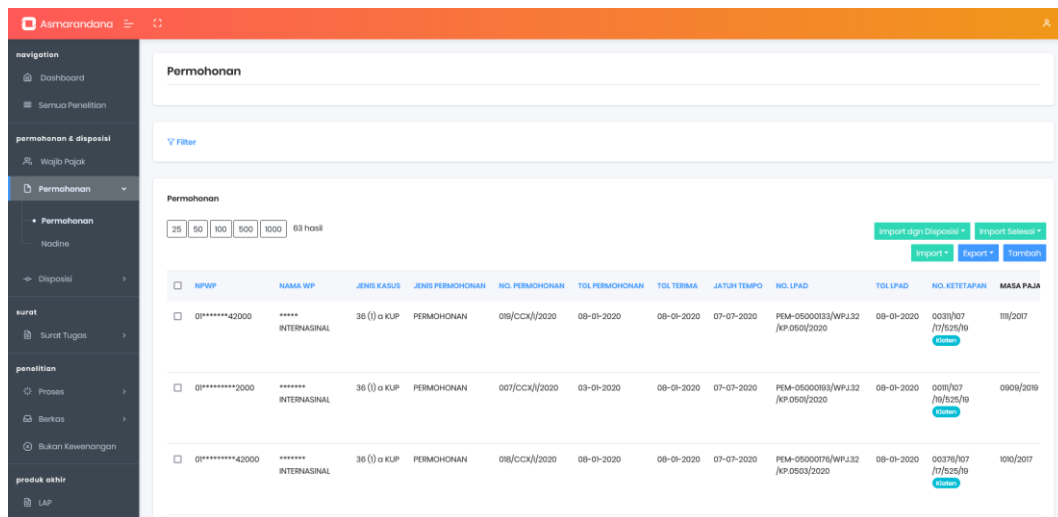
Pada halaman *dashboard* memperlihatkan laporan statistik mengenai proses permohonan. Pengguna dapat melihat grafik jumlah permohonan yang masuk dan jumlah permohonan yang telah berhasil diselesaikan. Selain itu *dashboard* juga menampilkan jumlah permohonan yang diproses serta yang telah berhasil diselesaikan oleh masing-masing seksi. Jumlah permohonan yang sedang diproses serta jumlah permohonan dengan sisa jatuh tempo 2 bulan dan 1 bulan juga ditampilkan pada halaman ini. Isi dari *side panel* dashboard akan berubah berdasarkan hak akses yang diberikan. Tampilan untuk halaman *dashboard* setelah login sebagai admin dapat dilihat pada gambar 8.



Gambar 8. Tampilan *dashboard* setelah login sebagai admin

Data permohonan yang mendekati jatuh tempo dapat dilihat cukup dengan mengeklik jumlah jatuh tempo yang ditampilkan pada *dashboard*. Pengguna akan dibawa pada tampilan data permohonan yang sedang berada dalam jatuh tempo setelah melakukan klik pada jumlah jatuh pada dashboard

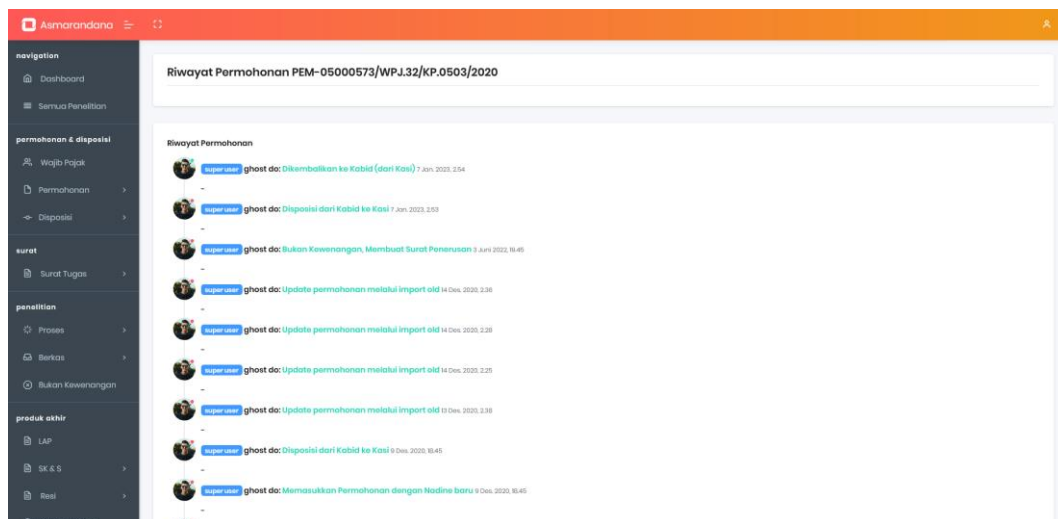
seperti ditampilkan pada gambar 9. Pengguna juga dapat melihat data seluruh data permohonan, permohonan yang telah selesai, dan yang sedang diproses dengan klik jumlah tautan masing-masing.



NPPWP	NAMA WP	JENIS KASUS	JENIS PERMOHONAN	NO. PERMOHONAN	TOL PERMOHONAN	TOL TERIMA	JATUH TEMPO	NO. LPAD	TOL LPAD	NO. KETETAPAN	MASA PAJAK
01*****42000	***** INTERNASIONAL	36 (1) a KUP	PERMOHONAN	018/CCX/2020	08-01-2020	08-01-2020	07-07-2020	PEM-05000153/WPJ.32 /KP.0503/2020	08-01-2020	00311/07 /17/025/19	11/2017
01*****2000	***** INTERNASIONAL	36 (1) a KUP	PERMOHONAN	007/CCX/2020	03-01-2020	08-01-2020	07-07-2020	PEM-05000153/WPJ.32 /KP.0503/2020	08-01-2020	00311/07 /17/025/19	09/2019
01*****42000	***** INTERNASIONAL	36 (1) a KUP	PERMOHONAN	018/CCX/2020	08-01-2020	08-01-2020	07-07-2020	PEM-05000176/WPJ.32 /KP.0503/2020	08-01-2020	00376/07 /17/025/19	10/2017

Gambar 9. Tampilan data permohonan dengan sisa jatuh tempo 1 bulan

Pengguna dapat melihat riwayat permohonan yang berisi kegiatan yang telah dilakukan dalam memproses permohonan tersebut. Riwayat permohonan ditampilkan pada gambar 10.



Riwayat Permohonan
ghost da: Dikembalikan ke Kasid (dari Kasid) 7 Jan 2020, 2:54
ghost da: Disposisi dari Kasid ke Kasid 7 Jan 2020, 2:53
ghost da: Bukan Kewenangan, Membuat Surat Penerimaan 9 Jan 2020, 16:45
ghost da: Update permohonan melalui import old 14 Des 2020, 2:36
ghost da: Update permohonan melalui import old 14 Des 2020, 2:36
ghost da: Update permohonan melalui import old 14 Des 2020, 2:25
ghost da: Update permohonan melalui import old 14 Des 2020, 2:36
ghost da: Disposisi dari Kasid ke Kasid 9 Des 2020, 16:45
ghost da: Memasukkan Permohonan dengan Nafine baru 9 Des 2020, 16:45

Gambar 10. Riwayat permohonan

Proses *import* permohonan dapat dilakukan dengan mengeklik tombol *import* pada tampilan permohonan terlihat pada gambar 9. Pengguna akan diarahkan menuju form *import* terlihat pada gambar 11. Pengguna memilih *file* yang akan dilakukan *import* dan memilih format *file* yang sesuai. Setelah klik *submit* maka sistem akan memvalidasi data yang ada pada *file* sebelum diproses.

Import Permohonan

Gunakan halaman ini untuk import permohonan yang telah selesai, dengan Id Nadin dan Tanggal Nadin

This importer will import the following fields: id_nadin, tanggal_nadin, npwp, jenis_kasus, jenis_permohonan, no_permohonan, tanggal_permohonan, no_upad, tanggal_upad, tanggal_terima, tanggal_jatuh_temporiser, no_ketetapan, masa_pajak_masi, masa_pajak_akhir, tahun_pajak, nilai, mata_uang, ndp_kasi, ndp_uk, nomor_lap, tanggal_lap, nomor_surat, tanggal_surat, hal_surat, nomor_sk, tanggal_sk, tanggal_kirik_sk, keputusan_sk, nilai_akhir_sk, kppesl_real_md, kppesl_real_jab, nomor_kerab, jenis_kerab

File to import*

Format*

Submit

Gambar 11. Tampilan form *import* permohonan

Gambar 12 merupakan tampilan tampilan *import preview* jika semua data yang ada pada *file* yang pengguna *upload* lolos validasi. Pengguna dapat melihat aksi apa saja yang akan dilakukan oleh sistem sebelum pengguna mensetujuinya dengan mengeklik *confirm import*. Apabila terjadi *error* pada data yang ada pada file seperti NPWP yang tidak ada pada *database* maka akan muncul *error*.

Import Preview

Below is a preview of data to be imported. If you are satisfied with the results, click 'Confirm import'

Confirm Import

ID_NADINE	TANGGAL_NADINE	NPWP	JENIS_KASUS	JENIS_PERMOMONAN	NO_PERMOMONAN	TANGGAL_PERMOMONAN	NO_UPAD	TANGGAL_UPAD	TANGGAL_TERIMA	TANGGAL_JATUH
New 8888	2020-01-02	457*****2000	38.01.a KUP	PERMOMONAN		2020-09-01	PEM-05007407/WP.32 /KP.0003/2020	2020-09-03	2020-09-03	2021-03-02
New 8888	2020-01-02	457*****2000	38.01.a KUP	PERMOMONAN		2020-09-01	PEM-05007420/WP.32 /KP.0003/2020	2020-09-03	2020-09-03	2021-03-02

Gambar 12. Tampilan *import preview* jika data yang dimasukkan dalam *file* tidak bermasalah

Permohonan dapat diproses secara kolektif dengan cara melakukan checklist pada data yang akan diproses, kemudian mengisi form tahapan proses yang berada di bawah tabel. Tampilan dapat dilihat pada gambar 13. Django admin yang dapat diakses admin dapat dilihat pada gambar 14.

Gambar 13. Proses pengisian Nadine pada permohonan secara kolektif

Gambar 14. Tampilan django admin

3.2 Hasil *Black Box Testing*

Hasil pengujian *black box testing* diperlihatkan pada tabel 1.

Tabel 1. Hasil pengujian *black box testing*

No	Kriteria	Pengujian	Input	Hasil yang diharapkan	Hasil
1	Autentikasi	Login dengan <i>username</i> dan <i>password</i> yang benar	Mengisi <i>username</i> dan <i>password</i> lalu klik <i>Login</i>	Menuju ke halaman <i>dashboard</i>	Valid
		Login dengan <i>username</i> dan <i>password</i> yang salah	Mengisi <i>username</i> dan <i>password</i> yang salah lalu klik <i>Login</i>	Menuju ke halaman <i>login</i> dengan pesan <i>error</i> jika <i>username/password</i> salah	Valid
		Login dengan <i>username</i> dan <i>password</i> yang benar dengan	Mengisi <i>username</i> dan <i>password</i> lalu klik <i>Login</i>	Menuju ke halaman untuk mengisi kode TOTP	Valid

		tambahan TOTP			
		TOTP dengan kode yang benar	Mengisi kode TOTP yang benar	Menuju ke halaman <i>dashboard</i>	Valid
		TOTP dengan kode yang salah	Mengisi kode TOTP yang salah	Tetap pada halaman TOTP dengan pesan bahwa kode salah	Valid
		<i>Logout</i> dari <i>dashboard</i>	Klik icon pengguna pada pojok kanan atas lalu klik tombol <i>icon logout</i>	Kembali ke halaman <i>login</i>	Valid
		<i>Logout</i> dari <i>django admin</i>	Klik keluar dari <i>django admin</i>	Kembali ke halaman <i>login</i>	Valid
2	Wajib Pajak	Menambah wajib pajak	Pada halaman data wajib pajak, klik tambah, mengisikan 15 digit NPWP dan nama lalu klik <i>Save</i>	Data NPWP berhasil tersimpan, muncul pesan bahwa data berhasil disimpan.	Valid
		Mengimpor wajib pajak dengan data yang benar	Pada halaman data wajib pajak, klik <i>Import</i> , memilih <i>file</i> , memilih <i>file format</i> yang sesuai, lalu klik <i>Submit</i>	Muncul halaman <i>preview import</i> yang sesuai dengan data yang dimasukkan	Valid
		Mengimpor wajib pajak dengan data yang salah	Pada halaman data wajib pajak, klik <i>import</i> , memilih <i>file</i> , memilih <i>file format</i> yang sesuai, lalu klik <i>Submit</i>	Muncul halaman <i>preview import</i> yang sesuai dengan informasi kesalahan data	Valid
		Konfirmasi dari <i>preview import</i>	Mengeklik tombol <i>Confirm Import</i>	Kembali ke halaman data WP dengan pesan bahwa data yang telah berhasil disimpan	Valid
		Mengubah data Wajib pajak	Mengeklik tombol dengan ikon <i>edit</i> , lalu ubah data, dan klik simpan	Data berhasil diubah, kembali ke halaman data wajib pajak, muncul pesan bahwa data berhasil diubah	Valid
3	Permohonan dan Nadine	Mengisian permohonan melalui form dengan data yang valid	Mengisi permohonan yang valid melalui form, lalu mengisi form yang diperlukan dan klik Simpan	Data terisi, informasi bahwa permohonan telah disimpan, dan menuju ke halaman tabel permohonan	Valid

	Mengisian permohonan melalui form dengan data yang valid dan ingin menambah lagi	Mengisi permohonan yang valid melalui form, lalu mengisi form yang diperlukan dan klik Simpan dan Tambah lagi	Data terisi, informasi bahwa permohonan telah disimpan, dan tetap pada halaman form permohonan yang siap untuk diisi data baru	Valid
	Mengisi form dengan data yang tidak valid	Mengisi permohonan yang tidak valid melalui form, lalu mengisi <i>form</i> yang diperlukan dan klik Simpan	Muncul pesan <i>error</i> bahwa data tidak valid.	Valid
	Mengisi form dengan data yang tidak lengkap	Mengisi permohonan yang tidak lengkap melalui form, lalu mengisi form yang diperlukan dan klik Simpan	Muncul pesan bahwa ada bagian form yang harus diisi.	Valid
	Melakukan <i>import</i> permohonan dalam proses	Melakukan <i>import</i> permohonan, memilih <i>file</i> , memilih <i>file format</i> yang sesuai dengan, klik submit	Muncul halaman <i>preview import</i> yang sesuai dengan data yang dimasukkan. Muncul tombol <i>confirm import</i> .	Valid
	Melakukan <i>import</i> permohonan dalam proses dengan data yang tidak valid	Melakukan <i>import</i> permohonan, memilih <i>file</i> dengan data yang tidak <i>valid</i> , memilih <i>file format</i> yang sesuai, klik submit	Muncul halaman <i>preview import</i> yang sesuai dengan informasi kesalahan data.	Valid
	Melakukan <i>import</i> permohonan yang selesai dengan data yang valid	Melakukan <i>import</i> permohonan, memilih <i>file</i> , memilih <i>file format</i> yang sesuai, klik submit	Muncul halaman <i>preview import</i> yang sesuai dengan data yang dimasukkan. Muncul tombol <i>Confirm Import</i> .	Valid
	Melakukan <i>import</i> permohonan yang selesai dengan data yang tidak valid	Melakukan <i>import</i> permohonan, memilih <i>file</i> dengan data yang tidak valid, memilih <i>file format</i> yang sesuai, klik submit	Muncul halaman <i>preview import</i> yang sesuai dengan informasi kesalahan data.	Valid

		Memilih permohonan yang akan dimasukkan dengan ID Nadine baru	Pada <i>sidebar</i> klik permohonan, lalu nadine, pilih permohonan yang akan dimasukkan pada ID Nadine yang baru, kemudian mengisi ID Nadine dan tanggal Nadine, Klik Set Nadine	Muncul pesan bahwa Permohonan sudah memiliki Nadine	Valid
		Memilih permohonan yang akan dimasukkan dengan ID Nadine yang sudah ada	Pada <i>sidebar</i> klik permohonan, lalu nadine, Klik tab Ubah, pilih permohonan yang akan dimasukkan pada ID Nadine yang baru, kemudian mengisi ID Nadine dan tanggal Nadine, Klik Set Nadine	Muncul pesan bahwa Permohonan sudah memiliki Nadine	Valid
		Mengubah data ID Nadine	Pada <i>sidebar</i> klik permohonan, lalu nadine, Klik tab List ID Nadine, klik ikon ubah pada ID Nadine yang akan diubah, lakukan perubahan data, klik simpan semua	Pengguna diarahkan ke halaman <i>List ID Nadine</i> . Muncul pesan bahwa Nadine telah berhasil diubah.	Valid
		Menghapus ID Nadine	Pada sidebar klik permohonan, lalu nadine, Klik <i>tab List ID Nadine</i> , pilih ID Nadine yang akan dihapus, <i>check</i> persetujuan penghapusan, klik hapus	Muncul pesan Nadine berhasil dihapus.	Valid
4	Disposisi	Disposisi Kabid ke Kasi	Memilih permohonan yang akan didisposisi, memilih Kepala Seksi, menuliskan catatan jika diperlukan, klik disposisikan.	Muncul pesan bahwa permohonan telah berhasil didisposisi.	Valid
		Disposisi Kasi ke Penelaah keberatan	Memilih permohonan yang akan didisposisi, memilih Penelaah Keberatan, menuliskan catatan jika diperlukan, klik Disposisikan.	Muncul pesan bahwa permohonan telah berhasil didisposisi. Data permohonan lanjut ke permohonan pada pembuatan Surat Tugas	Valid

		Kembalikan Kasi ke Kabid	Memilih permohonan yang akan didisposisi, memilih Penelaah Keberatan, menuliskan catatan jika diperlukan, klik Kembalikan ke Kabid.	Muncul pesan bahwa permohonan berhasil dikembalikan ke Kabid.	Valid
5	Surat Tugas	Membuat Surat Tugas	Memilih permohonan, mengisi nomor, tanggal, mengisi catatan jika diperlukan. Klik Simpan.	Muncul pesan bahwa Surat Tugas berhasil dibuat.	Valid
		Mengirim Surat Tugas	Memilih Surat Tugas yang ingin dikirim, mengisi catatan jika diperlukan, klik Kirim	Muncul pesan bahwa Surat Tugas berhasil dikirim.	Valid
		Mengganti Surat Tugas	Memilih Surat Tugas yang ingin diganti, mengisi Nomor ST Pengganti, Tanggal, Kasi, Penelaah keberatan, dan mengisi Catatan jika perlu. Klik Simpan	Muncul pesan bahwa Surat Tugas berhasil diganti.	Valid
		Terima Surat Tugas	Memilih Surat Tugas, mengisi catatan jika diperlukan, klik Terima	Muncul pesan bahwa Surat Tugas telah diterima.	Valid
		Mengembalikan Surat Tugas	Memilih Surat Tugas, mengisi catatan jika diperlukan, klik Kembalikan	Muncul pesan bahwa Surat Tugas telah dikembalikan.	Valid
6	Proses Penelitian	Melihat Penelitian	Klik Proses pada <i>sidebar</i> , lalu klik Lihat Penelitian	Muncul data penelitian yang perlu diproses	Valid
		Memproses penelitian untuk permohonan yang belum penelitian	Memilih permohonan yang akan diproses, lakukan proses pada form dibawah, dan klik Simpan	Muncul pesan bahwa data telah berhasil disimpan	Valid
		Mengubah isi penelitian	Pilih <i>Tab</i> Ubah, pilih penelitian yang ingin ubah, isi form dibawah dengan data baru lalu simpan	Muncul pesan bahwa data telah berhasil disimpan	Valid
		Menghapus proses penelitian	Pilih <i>Tab</i> hapus, pilih penelitian yang ingin dihapus, check bahwa anda yakin ingin menghapus lalu klik hapus.	Muncul pesan bahwa data telah berhasil dihapus	Valid

7	Bukan Kewenangan	Membuat surat penerusan jika bukan kewenangannya	Pilih permohonan yang ingin diproses, isi form yang ada dibawah, isi catatan jika diperlukan, klik Simpan	Muncul pesan bahwa data telah berhasil disimpan	Valid
8	LAP	Membuat LAP	Pilih data yang ingin diproses, isi nomor LAP, tanggal LAP, dan isi catatan jika diperlukan, lalu klik Simpan	Muncul pesan bahwa LAP berhasil disimpan.	Valid
		Mengisi nomor LAP pada permohonan dengan nomor LAP yang telah ada	Pada <i>Tab</i> Ubah, pilih data yang ingin diproses, isi nomor LAP yang telah ada sebelumnya lalu klik Simpan	Muncul pesan bahwa Nomor LAP berhasil disimpan.	Valid
		Mengubah data LAP	Pada <i>Tab</i> Daftar <i>List</i> , pada data yang ingin diproses klik ikon ubah, lakukan perubahan dan isi catatan jika diperlukan lalu klik Simpan Semua	Muncul pesan bahwa LAP berhasil diubah.	Valid
		Menghapus LAP	Pada <i>Tab</i> Daftar <i>List</i> , pada LAP yang ingin diproses, centang persetujuan penghapusan, lalu klik hapus.	Muncul pesan bahwa LAP berhasil dihapus	Valid
9	Surat Keputusan (SK)	Mengisi SK	Pilih data yang ingin diproses, isi form, klik Simpan	Muncul pesan bahwa Surat Keputusan berhasil disimpan.	Valid
		Mengubah SK	Klik <i>Tab List</i> Surat Keputusan, pada data yang ingin diubah klik ikon ubah yang ada pada paling kanan, lalu ubah data dan klik Simpan	Muncul pesan bahwa Surat Keputusan berhasil diubah	Valid
		Menghapus SK	Klik <i>Tab List</i> Surat Keputusan, pada data yang ingin dihapus klik ikon hapus yang ada pada paling kanan, lalu klik Hapus	Muncul pesan bahwa Surat Keputusan berhasil dihapus	Valid
10	Surat	Mengisi Surat	Pilih data yang ingin diproses, isi form, klik Simpan	Muncul pesan bahwa Surat berhasil disimpan.	Valid

		Mengubah Surat	Klik <i>Tab List</i> Surat, pada data yang ingin diubah klik ikon ubah yang ada pada paling kanan, lalu ubah data dan klik Simpan	Muncul pesan bahwa Surat berhasil diubah	Valid
		Menghapus Surat	Klik <i>Tab List</i> Surat, pada data yang ingin dihapus klik ikon hapus yang ada pada paling kanan, lalu klik Hapus	Muncul pesan bahwa Surat berhasil dihapus	Valid
11	Resi WP	Mengisi Resi WP	Memilih permohonan yang ingin diproses, mengisi nomor resi dan tanggal resi, lalu tambahkan catatan jika diperlukan.	Muncul pesan bahwa nomor resi WP telah disimpan	Valid
		Mengisi Resi WP dengan resi yang sudah ada	Menuju ke <i>Tab Ubah</i> , lalu pilih data yang ingin diisi resi WP dengan resi yang ada, ketik nomor resi WP, pilih dari nomor resi WP yang ada, klik Simpan	Muncul pesan bahwa nomor resi WP telah disimpan/diubah	Valid
		Mengubah Resi WP	Menuju ke <i>Tab List</i> Resi WP, lalu pada data yang ingin diubah klik ikon ubah, ubah data yang diinginkan, klik Simpan Semua	Muncul pesan bahwa resi WP telah diubah	Valid
		Menghapus Resi WP	Pilih data resi WP yang ingin dihapus, centang pada persetujuan penghapusan, lalu klik Hapus	Muncul pesan bahwa resi WP telah dihapus	Valid
12	Resi KPP	Mengisi Resi KPP	Memilih permohonan yang ingin diproses, mengisi nomor resi dan tanggal resi, lalu tambahkan catatan jika diperlukan.	Muncul pesan bahwa nomor resi KPP telah disimpan	Valid
		Mengisi Resi KPP dengan resi yang sudah ada	Menuju ke <i>Tab Ubah</i> , lalu pilih data yang ingin diisi resi KPP dengan resi yang ada, ketik nomor resi KPP, pilih dari nomor resi KPP yang ada, klik Simpan	Muncul pesan bahwa nomor resi KPP telah disimpan/diubah	Valid

		Mengubah Resi KPP	Menuju ke <i>Tab List</i> Resi KPP, lalu pada data yang ingin diubah klik ikon ubah, ubah data yang diinginkan, klik Simpan Semua	Muncul pesan bahwa resi KPP telah diubah	Valid
		Menghapus Resi KPP	Pilih data resi KPP yang ingin dihapus, centang pada persetujuan penghapusan, lalu klik Hapus	Muncul pesan bahwa resi KPP telah dihapus	Valid
13	Kardus Berkas	Mengisi data Kardus Berkas	Memilih permohonan yang ingin diproses, mengisi nomor kardus dan jenis kardus, lalu tambahkan catatan jika diperlukan.	Muncul pesan bahwa kardus berkas telah disimpan	Valid
		Mengisi dengan nomor kardus yang sudah ada	Menuju ke <i>Tab Ubah</i> , lalu pilih data yang ingin diisi nomor kardus dengan nomor kardus yang sudah ada, ketik nomor kardus, pilih dari nomor kardus yang ada, klik Simpan	Muncul pesan bahwa nomor kardus berkas telah disimpan/diubah	Valid
		Mengubah data Kardus Berkas	Menuju ke <i>Tab List</i> Kardus Berkas, lalu pada data yang ingin diubah klik ikon ubah, ubah data yang diinginkan, klik Simpan Semua	Muncul pesan bahwa data Kardus Berkas telah diubah	Valid
		Menghapus Kardus Berkas	Pilih data Kardus Berkas yang ingin dihapus, centang pada persetujuan penghapusan, lalu klik Hapus	Muncul pesan bahwa Kardus Berkas telah dihapus	Valid
14	Hak akses	<i>Side panel</i> yang berubah mengikuti status/ <i>role</i> pengguna	Pengguna melakukan <i>login</i> dan menuju ke <i>dashboard</i> .	Item side panel mengikuti dengan <i>role</i> pengguna	Valid
		Akses django admin	Pengguna dengan akses <i>superuser</i> atau staff mengakses admin melalui URL atau <i>sidebar</i>	Menuju halaman django admin	Valid

Berdasarkan hasil pengujian dengan metode *black box* sistem dapat berfungsi sesuai dengan hasil yang diharapkan. Meskipun begitu tidak kemungkinan dapat terjadinya kesalahan pada aplikasi ini di masa mendatang.

3.3 Hasil Uji *Usability*

Pengujian *usability* menggunakan metode SUS yang dilakukan dengan kuesioner. Sejumlah 9 responden berpartisipasi dalam kuesioner untuk menentukan tingkat kegunaan sistem ini. SUS berisi 10 pertanyaan yang dapat dilihat pada tabel 2.

Tabel 2. Pertanyaan Kuesioner

Pertanyaan Kuesioner	
P1	Saya pikir saya akan sering menggunakan aplikasi/fitur ini.
P2	Saya merasa aplikasi/fitur ini terlalu rumit padahal dapat dibuat lebih sederhana.
P3	Saya rasa aplikasi/fitur ini mudah untuk digunakan.
P4	Saya pikir saya membutuhkan bantuan dari orang teknis untuk dapat menggunakan aplikasi/fitur ini.
P5	Saya menemukan bahwa terdapat berbagai macam aplikasi/fitur yang terintegrasi dengan baik dalam sistem.
P6	Saya rasa banyak hal yang tidak konsisten terdapat pada aplikasi/fitur ini.
P7	Saya rasa mayoritas pengguna akan dapat mempelajari aplikasi/fitur ini dengan cepat.
P8	Saya menemukan bahwa aplikasi/fitur ini sangat tidak praktis ketika digunakan.
P9	Saya sangat yakin dapat menggunakan aplikasi/fitur ini.
P10	Saya harus belajar banyak hal terlebih dahulu sebelum saya dapat menggunakan aplikasi/fitur ini.

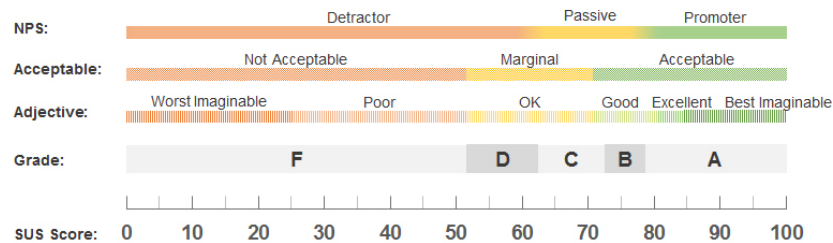
Dari pertanyaan tersebut SUS diberikan 5 pilihan jawaban dengan nilainya yaitu, sangat tidak setuju dengan nilai 1, tidak setuju dengan nilai 2, netral dengan nilai 3, setuju dengan nilai 4, sangat setuju dengan nilai 5. Kemudian untuk pertanyaan ganjil nilai kontribusinya yaitu nilai dikurangi 1, sedangkan untuk pertanyaan genap nilai dikurangi 5. Hasil akhir dari nilai masing-masing pertanyaan yaitu antara 0 hingga 4. Jumlahkan nilai dari 10 pertanyaan tersebut dan dikalikan 2,5 maka didapatkan nilai SUS dari masing-masing responden. Setelah itu rata-ratakan nilai SUS semua responden. Hasil nilai SUS dapat dilihat pada tabel 3.

Tabel 3. Hasil Uji *Usability*

Responden	Kalkulasi										Hasil	Nilai SUS
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10		
1	2	3	4	3	2	3	4	2	3	3	29	72,5
2	3	2	3	2	3	2	3	3	3	2	26	65
3	3	2	3	2	2	2	3	2	3	2	24	60
4	3	2	3	2	3	2	3	2	3	2	25	62,5
5	3	1	2	3	2	2	3	2	4	3	25	62,5
6	3	2	3	2	3	2	3	2	3	2	25	62,5

7	4	2	3	3	3	2	4	3	4	2	30	75
8	3	1	3	2	2	2	3	2	3	1	22	55
9	4	4	4	3	4	3	4	4	4	3	37	92,5
											Total SUS	607,5
											Rata-rata	67,5

Berdasarkan tabel 3, hasil rata-rata dari hasil uji kegunaan sistem adalah 67,5. Maka dapat disimpulkan bahwa sistem mendapatkan posisi penerimaan yang marginal. Pendekatan penerimaan pengguna berdasarkan skor SUS dapat dilihat pada gambar 15.



Gambar 15. Pendekatan penerimaan pengguna berdasarkan skor SUS

4. PENUTUP

4.1 Kesimpulan

Sistem informasi telah dikembangkan dan telah digunakan oleh kanwil dapat membantu dalam mengelola permohonan, memantau proses perkembangan permohonan, dan mengetahui jatuh tempo permohonan secara cepat dan akurat. Setelah semua tahap pengembangan dan pengujian, dapat disimpulkan bahwa penelitian ini menghasilkan sistem informasi berbasis web yang dapat menampilkan informasi proses permohonan dan pemantauan proses permohonan dengan skor SUS 67,5 dan mendapatkan posisi penerimaan yang marginal. Adapun fitur lainnya yang terdapat pada aplikasi ini seperti autentikasi 2 faktor dengan TOTP, import permohonan dengan file xls atau xlsx, dan pemrosesan permohonan secara kolektif pada tahapan tertentu.

4.2 Saran

Dalam penelitian berikutnya diharapkan dapat mengembangkan aplikasi dengan perencanaan yang lebih matang dan desain dengan *user experience* yang lebih baik sehingga menghasilkan kode yang lebih tertata serta dapat meningkatkan skor SUS.

PERSANTUNAN

Terima kasih kepada seluruh pegawai Kanwil DJP Jawa Tengah II yang terlibat yang telah meluangkan waktu dan tenaga demi penulisan penelitian ini.

DAFTAR PUSTAKA

Adel, A., & Abdullah, B. (2015). A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. *IJCSI International Journal of Computer Science Issues*, 12(1), 106–111.
https://www.academia.edu/10793943/A_Comparison_Between_Three_SDLC_Models_Waterf

- Al-masree, H. K. (2015). Extracting Entity Relationship Diagram (ERD) From Relational Database Schema. *International Journal of Database Theory and Application*, 8(3), 15–26. <https://doi.org/10.14257/IJDTA.2015.8.3.02>
- Amalia, A., Wanda, S., Hamidah, P., & Kristanto, T. (2021). Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web. *Building of Informatics, Technology and Science (BITS)*, 3(3), 269–274. <https://doi.org/10.47065/bits.v3i3.1062>
- Ariawan, M. R., & Ariyanto, G. (2018). SISMTTEM INFORMASI WARUNG KULINER. *Emitor: Jurnal Teknik Elektro*, 18(1), 36–41. <https://doi.org/10.23917/emitor.v18i01.6238>
- Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Memahami Penggunaan UML (Unified Modelling Language)*, 6(1), 1–15. <http://e-journals.unmul.ac.id/index.php/JIM/article/download/16/pdfn-feb-2011.pdf>
- Khafid, B., & Putri, D. A. P. (2020). Pesma Apps as Android-based Integrated Applications for Mahasantri Pesma KH Mas Mansur UMS. *Khazanah Informatika: Jurnal Ilmu Komputer Dan Informatika*, 6(2), 95–102. <https://doi.org/10.23917/khif.v6i2.10494>
- Kulanda Duisebekova, Roman Khabirov, A. Z. (2021). DJANGO AS SECURE WEB-FRAMEWORK IN PRACTICE. *The Bulletin of KazATC*, 1(116), 275–281. <https://doi.org/10.52167/1609-1817-2021-116-1-275-281>
- Lewis, J. R. (2018). The System Usability Scale : Past , Present , and Future The System Usability Scale : Past , Present , and Future. *International Journal of Human–Computer Interaction*, 34(7), 557–590. <https://doi.org/10.1080/10447318.2018.1455307>
- Melliana, A. I., & Nurgiyatna, N. (2021). Sistem Informasi Arsip Surat Pada SMA Negeri 2 Sukoharjo Menggunakan Framework Codeigniter. *Jurnal Pendidikan Dan Teknologi Indonesia*, 1(4), 141–149. <https://doi.org/10.52436/1.jpti.29>
- Menteri Keuangan Republik Indonesia. (2020). *Peraturan Menteri Keuangan Nomor 184/PMK.01/2020 Tentang Perubahan Atas Peraturan Menteri Keuangan Nomor 210/PMK.01/2017 Tentang Organisasi Dan Tata Kerja Instansi Vertikal Direktorat Jenderal Pajak* (pp. 1–134). <https://jdih.kemenkeu.go.id/en/dokumen/peraturan/06998df3-a518-4605-645b-08d89663476f>
- Pérez, B. M. C., Ramos, J. R., Mora, J. J. H., Arenas, M. C., & Hernández, M. J. S. (2021). Experience of using the Django ORM for Migration, Redesign and Implementation of a Database of MySQL Data to Postgre SQL. *International Journal of Science and Research (IJSR)*, 10(11), 96–98. <https://doi.org/10.21275/SR211030134428>
- Permana, I., Hardjianto, M., & Ahmad Baihaqi, K. (2020). Securing the Website Login System with the SHA256 Generating Method and Time-based One-time Password (TOTP). *Systematics*, 2(2), 65–71. <https://doi.org/10.35706/sys.v2i2.3756>
- Puspaningrum, A., & Sudarmilah, E. (2020). SISTEM INFORMASI MANAJEMEN PEMINJAMAN (STUDI KASUS : PENGELOLAAN ASET DAN TATA RUANG TAMAN BUDAYA JAWA TENGAH). *Technologia: Jurnal Ilmiah*, 11(1), 37–45. <https://doi.org/10.31602/TJI.V11i1.2699>
- Rubio, D. (2017). Django admin Management. In *Beginning Django: Web Application Development and Deployment with Python* (pp. 495–547). Apress. https://doi.org/10.1007/978-1-4842-2787-9_11
- Susanto, A., Septiani, N. W. P., & Lestari, M. (2021). Perancangan Sistem Informasi Pengelolaan Surat Masuk Surat Keluar dan SPPD Dikelurahan Jatijajar. *Jurnal Riset Dan Aplikasi Mahasiswa Informatika (JRAMI)*, 2(01), 8–14. <https://doi.org/10.30998/jrami.v2i01.590>