

1. ***Use case description* Login**

*Name of the use case* : Login

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan setiap user untuk melakukan login pada sistem informasi.

*Characteristic of activation* : Resepsionis, Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah

*Pre-condition(s)* : User berada di halaman Login dan mengetahui id beserta password

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman login
2. Aktor mengisi username dan password pada form login
3. Aktor mengklik tombol “Login”
4. Sistem memeriksa username dan password di dalam database. Jika terdapat username dan password yang diinputkan ada di dalam database, maka sistem akan dialihkan ke halaman beranda. Jika tidak, sistem akan menampilkan pesan “ID dan password salah."

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : User berhasil login dan masuk ke dalam aplikasi

*Rule(s)* :

1. Aktor harus memiliki akun

*Constraint(s)* :

1. User hanya bisa memiliki 1 akun

*GUI* (optional)

1. ***Use case description* Mengelola Data Pengajuan Surat Masuk**

*Name of the use case* : Mengelola Data Pengajuan Surat Masuk

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan Resepsionis untuk melakukan pengelolaan data pengajuan surat masuk, termasuk menambahkan, mengedit, melihat, dan menghapus. Selain itu, use case ini menyediakan fungsionalitas yang memungkinkan Staf Administrasi Umum untuk melihat detail data pengajuan surat masuk yang dipilih.

*Characteristic of activation* : Resepsionis, Staf Administrasi Umum

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda.
2. Aktor mengklik “Data Pengajuan Surat Masuk” yang ada di sidebar halaman beranda.
3. Sistem menampilkan halaman data pengajuan surat masuk yang terdiri dari tabel.
4. Jika Resepsionis ingin menambahkan data pengajuan surat masuk, maka resepsionis mengklik tombol “Tambah Data”. Kemudian sistem dialihkan ke halaman form tambah pengajuan surat masuk yang terdiri dari nama lengkap, asal instansi, jabatan, tanggal kunjungan, isi ringkasan surat, file surat. Resepsionis mengisi form tersebut dan klik tombol “Tambah”. Data yang sudah diinputkan akan ditambahkan ke dalam database.
5. Jika Resepsionis ingin mengedit data pengajuan surat masuk, maka Resepsionis mengklik ikon “pensil” di kolom aksi. Kemudian sistem dialihkan ke halaman form edit pengajuan surat masuk. Resepsionis mengganti data yang ingin digantikan. Jika sudah selesai, Resepsionis mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
6. Jika Resepsionis atau Staf Administrasi Umum ingin melihat detail data pengajuan surat masuk, maka Resepsionis mengklik ikon “mata” di kolom aksi. Kemudian sistem dialihkan ke halaman detail pengajuan surat masuk.
7. Jika Resepsionis ingin menghapus data pengajuan surat masuk, maka Resepsionis mengklik ikon “tong sampah” di kolom aksi. Sistem akan menampilkan pesan “Apakah anda ingin menghapus data ini di dalam dabatase?” beserta tombol “Ya” dan “Kembali”. Jika Resepsionis ingin menghapus data, maka mengklik tombol “Ya” dan data tersebut akan dihapuskan di dalam database. Jika tidak ingin menghapus data tersebut, maka mengklik tombol “Kembali” dan sistem akan dialihkan kembali ke halaman pengajuan surat masuk.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diupdate di dalam database

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Tidak ada

*GUI* (optional)

1. ***Use case description* Mengelola Data Surat Masuk**

*Name of the use case* : Mengelola Data Surat Masuk

*Brief description* : Use case ini menyediakan fungsionalitas yang memungkinkan Staf Administrasi Umum untuk melakukan pengelolaan data surat masuk, termasuk menambahkan, mengubah, melihat, dan menghapus data surat masuk. Selain itu, use case ini menyediakan fungsionalitas yang memungkinkan Kepala Tata Usaha dan Kepala Sekolah untuk mengubah dan melihat data surat masuk.

*Characteristic of activation* : Staf Administrasi Umum, Kepala Tata Usaha dan Kepala Sekolah

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda.
2. Jika Staf Administrasi Umum ingin menambah data surat masuk, maka Staf Administrasi umum klik “Data Surat Masuk” di sidebar halaman beranda. Di bagian kolom aksi, Staf Administrasi Umum mengklik tombol “Tambah Data” dan dialihkan ke halaman form tambah surat masuk. Kemudian Staf Administrasi Umum menginputkan data yang ada di form tambah surat masuk berupa nama surat, tanggal surat, sifat surat, perihal, dan file surat. Setelah menginputkan data, Staf Administrasi Umum mengklik tombol “Tambah”.
3. Jika aktor ingin mengedit data surat masuk, maka aktor klik “Data Surat Masuk” di sidebar halaman beranda. Di bagian kolom aksi, aktor mengklik mengklik ikon “pensil” di kolom aksi. Kemudian sistem dialihkan ke halaman form edit data surat masuk. Aktor mengganti data yang ingin digantikan. Jika sudah selesai, aktor mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
4. Jika aktor ingin melihat detail data surat masuk, maka aktor klik “Data Surat Masuk” di sidebar halaman beranda. Di bagian kolom aksi, Staf Administrasi Umum mengklik mengklik ikon “mata” kemudian sistem dialihkan ke halaman detail data surat masuk.
5. Jika Staf Administrasi Umum ingin menghapus data surat masuk, maka Staf Administrasi Umum klik “Data Surat Masuk” di sidebar halaman beranda. Di bagian kolom aksi, Staf Administrasi Umum mengklik mengklik ikon “tong sampah”. Sistem akan menampilkan pesan “Apakah anda ingin menghapus data ini di dalam dabatase?” beserta tombol “Ya” dan “Kembali”. Jika Staf Administrasi Umum ingin menghapus data, maka mengklik tombol “Ya” dan data tersebut akan dihapuskan di dalam database. Jika tidak ingin menghapus data tersebut, maka mengklik tombol “Kembali” dan sistem akan dialihkan kembali ke halaman data surat masuk.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diupdate di dalam database

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Kepala Tata Usaha dan Kepala Sekolah hanya bisa mengubah file surat

*GUI* (optional)

1. ***Use case description* Mengelola Data Legalisir Ijazah**

*Name of the use case* : Mengelola Data Legalisir Ijazah

*Brief description* : Use case ini menyediakan fungsionalitas yang memungkinkan Resepsionis untuk melakukan pengelolaan data legalisir ijazah, termasuk menambahkan, mengubah, melihat, dan menghapus data legalisir ijazah. Selain itu, use case ini menyediakan fungsionalitas yang memungkinkan Staf Administrasi Umum dan Kepala Sekolah untuk mengubah dan melihat data legalisir ijazah.

*Characteristic of activation* : Resepsionis, Staf Administrasi Umum, Kepala Sekolah

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda.
2. Aktor mengklik “Legalisir Ijazah” yang ada di sidebar halaman beranda.
3. Sistem menampilkan halaman legalisir ijazah.
4. Jika Resepsionis ingin menambahkan data legalisir ijazah, maka resepsionis mengklik tombol “Tambahkan Data”. Kemudian sistem dialihkan ke halaman form tambah legalisir ijazah yang terdiri dari nama lengkap, tempat tanggal lahir, nomor induk, tahun alumni, tanggal kunjungan, dan file surat. Resepsionis mengisi form tersebut dan klik tombol “Tambah”. Data yang sudah diinputkan akan ditambahkan ke dalam database.
5. Jika aktor ingin mengedit data legalisir ijazah, maka aktor mengklik ikon “pensil” di kolom aksi. Kemudian sistem dialihkan ke halaman form edit legalisir ijazah. Aktor mengganti data yang ingin digantikan. Jika sudah selesai, Aktor mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
6. Jika aktor ingin melihat detail data legalisir ijazah, maka aktor mengklik ikon “mata” di kolom aksi. Kemudian sistem dialihkan ke halaman detail legalisir ijazah.
7. Jika Resepsionis ingin menghapus data surat masuk, maka Resepsionis mengklik ikon “tong sampah” di kolom aksi. Sistem akan menampilkan pesan “Apakah anda ingin menghapus data ini di dalam dabatase?” beserta tombol “Ya” dan “Kembali”. Jika Resepsionis ingin menghapus data, maka mengklik tombol “Ya” dan data tersebut akan dihapuskan di dalam database. Jika tidak ingin menghapus data tersebut, maka mengklik tombol “Kembali” dan sistem akan dialihkan kembali ke halaman data pengajuan surat masuk.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diupdate di dalam database

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Staf Administrasi Umum dan Kepala Sekolah hanya bisa melihat dan mengubah file legalisir ijazah.

*GUI* (optional)

1. ***Use case description* Mengelola Data Pengajuan Surat Keluar**

*Name of the use case* : Mengelola Data Pengajuan Surat Keluar

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan Staf Administrasi Umum untuk melakukan pengelolaan data pengajuan surat keluar, termasuk menambahkan, mengedit, melihat, menghapus data pengajuan surat keluar. Selain itu, use case ini menyediakan fungsionalitas yang memungkinkan Kepala Tata Usaha dan Kepala Sekolah untuk mengubah dan melihat data pengajuan surat keluar.

*Characteristic of activation* : Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah.

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda.
2. Aktor mengklik “Data Pengajuan Surat Keluar” yang ada di sidebar halaman beranda.
3. Sistem menampilkan halaman data pengajuan surat keluar yang terdiri dari tabel.
4. Jika Staf Administrasi Umum ingin menambahkan data pengajuan surat keluar, maka Staf Administrasi Umum mengklik tombol “Tambah Data”. Kemudian sistem dialihkan ke halaman form tambah pengajuan surat keluar yang terdiri dari nama pengetik, NIP, konsep surat, file surat. Staf Administrasi Umum mengisi form tersebut dan klik tombol “Tambah”. Data yang sudah diinputkan akan ditambahkan ke dalam database.
5. Jika aktor ingin mengedit data pengajuan surat keluar, maka aktor mengklik ikon “pensil” di kolom aksi. Kemudian sistem dialihkan ke halaman edit pengajuan surat keluar. Aktor mengganti data yang ingin digantikan. Jika sudah selesai, aktor mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
6. Jika aktor ingin melihat detail data pengajuan surat keluar, maka aktor mengklik ikon “mata” di kolom aksi. Kemudian sistem dialihkan ke halaman detail pengajuan surat keluar.
7. Jika Staf Administrasi Umum ingin menghapus data pengajuan surat keluar, maka Staf Administrasi Umum mengklik ikon “tong sampah” di kolom aksi. Sistem akan menampilkan pesan “Apakah anda ingin menghapus data ini di dalam dabatase?” beserta tombol “Ya” dan “Kembali”. Jika Staf Administrasi Umum ingin menghapus data, maka mengklik tombol “Ya” dan data tersebut akan dihapuskan di dalam database. Jika tidak ingin menghapus data tersebut, maka mengklik tombol “Kembali” dan sistem akan dialihkan Kembali ke halaman data pengajuan surat keluar.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diupdate di dalam database

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Kepala Tata Usaha dan Kepala Sekolah hanya bisa melihat dan mengubah file pengajuan surat keluar.

*GUI* (optional)

1. ***Use case description* Mengelola Data Surat Keluar**

*Name of the use case* : Mengelola Data Surat Keluar

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan Staf Administrasi Umum untuk melakukan pengelolaan data pengajuan surat keluar, termasuk menambahkan, mengedit, melihat, menghapus data pengajuan surat keluar.

*Characteristic of activation* : Staf Administrasi Umum

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda.
2. Jika aktor ingin menambahkan data surat keluar, maka aktor mengklik tombol “Tambah Data”. Kemudian sistem dialihkan ke halaman form tambah surat keluar yang terdiri dari nomor surat, tanggal surat, sifat surat, perihal, dan file surat. Aktor mengisi form tersebut dan klik tombol “Tambah”. Data yang sudah diinputkan akan ditambahkan ke dalam database.
3. Jika Aktor ingin mengedit data surat keluar, maka aktor klik “Data Surat Keluar” di sidebar halaman beranda. Di bagian kolom aksi, aktor mengklik mengklik ikon “pensil” di kolom aksi. Kemudian sistem dialihkan ke halaman edit data surat keluar. Aktor mengganti data yang ingin digantikan. Jika sudah selesai, aktor mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
4. Jika aktor ingin melihat detail data surat keluar, maka aktor klik “Data Surat Keluar” di sidebar halaman beranda. Di bagian kolom aksi, aktor mengklik ikon “mata” Kemudian sistem dialihkan ke halaman detail data surat keluar.
5. Jika aktor ingin menghapus data pengajuan surat keluar, maka aktor mengklik ikon “tong sampah” di kolom aksi. Sistem akan menampilkan pesan “Apakah anda ingin menghapus data ini di dalam dabatase?” beserta tombol “Ya” dan “Kembali”. Jika Staf Administrasi Umum ingin menghapus data, maka mengklik tombol “Ya” dan data tersebut akan dihapuskan di dalam database. Jika tidak ingin menghapus data tersebut, maka mengklik tombol “Kembali” dan sistem akan dialihkan Kembali ke halaman data pengajuan surat keluar.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diupdate di dalam database

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Tidak ada

*GUI* (optional)

1. ***Use case description* Menyerahkan Data Surat**

*Name of the use case* : Menyerahkan Data Surat

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan Resepsionis, Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah menyerahkan data surat masuk, surat keluar, atau legalisir ijazah ke aktor lain.

*Characteristic of activation* : Resepsionis, Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah

*Pre-condition(s)* : User berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Aktor berada di halaman beranda
2. Aktor mengklik menu data surat yang ingin diserahkan pada sidebar pada halaman beranda.
3. Sistem menampilkan halaman data surat yang dipilih
4. Dibagian kolom aksi, aktor mengklik ikon “panah” dan diarahkan ke halaman penyerahan data surat.
5. Di halaman penyerahan data surat, aktor memilih tujuan penyerahan dan memberikan catatan.
6. Aktor mengklik tombol “Serahkan” dan akan muncul pesan “Apakah anda ingin menyerahkan data ini untuk dikelola lebih lanjut?”. Jika ingin menyerahkan, maka aktor mengklik tombol “Ya”. Jika tidak, maka aktor mengklik tombol “Kembali”. Lalu aktor mengklik tombol “Ya”. Data surat berhasil diserahkan.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : Data sudah berhasil diserahkan

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Tidak ada

*GUI* (optional)

1. ***Use case description* Melihat Hasil Akhir Surat Keluar dan Legalisir Ijazah**

*Name of the use case* : Menerima Hasil Akhir Surat Keluar dan Legalisir Ijazah

*Brief description* : Usecase ini menyediakan fungsionalitas yang memungkinkan Resepsionis untuk melihat dan mengunduh hasil akhir dari surat keluar dan legalisir ijazah.

*Characteristic of activation* : Resepsionis

*Pre-condition(s)* : Aktor berada di halaman beranda.

*Flow of Event*

¤ *Basic Flow*

1. Resepsionis berada di halaman beranda
2. Resepsionis mengklik “Hasil” di sidebar pada halaman beranda.
3. Sistem menampilkan menu “Surat Keluar” dan “Legalisir Ijazah”
4. Jika Resepsionis ingin mendownload hasil surat keluar, maka Resepsionis mengklik menu “Surat Keluar” dan sistem akan dialihkan ke halaman hasil surat keluar. Resepsionis mengklik ikon “mata” pada kolom aksi dan dialihkan ke halaman hasil surat keluar. Resepsionis mengklik file yang ada, dan akan muncul pop up untuk mendownload. Resepsionis klik download, dan file siap untuk diprint.
5. Jika Resepsionis ingin mendownload hasil legalisir ijazah, maka Resepsionis mengklik menu “Legalisir Ijazah” dan sistem akan dialihkan ke halaman hasil legalisir ijazah. Resepsionis mengklik ikon “mata” pada kolom aksi dan dialihkan ke halaman hasil legalisir ijazah. Resepsionis mengklik file yang ada, dan akan muncul *pop-up* untuk mendownload. Resepsionis klik download, dan file siap untuk diprint.

¤ *Alternative Flow*

1. Tidak ada

¤ *Exception Flow*

1. Tidak ada

*Post-condition(s)* : File surat sudah berhasil di download

*Rule(s)* :

1. Tidak ada

*Constraint(s)* :

1. Tidak ada

*GUI* (optional)

1. ***Use case description* Melihat Status Surat**

*Name of the use case :* Melihat Status Surat

*Brief description :* Usecase ini menyediakan fungsionalitas kepada Resepsionis, Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah untuk melihat status surat yang dipilih. Status surat digunakan untuk melihat posisi surat yang sedang dikelola.

*Characteristic of activation :* Resepsionis, Staf Administrasi Umum, Kepala Tata Usaha, dan Kepala Sekolah

*Pre-condition(s) :*Aktor berada di halaman “Beranda”

*Flow of Event*

*¤ Basic Flow*

1. Aktor berada di halaman beranda
2. Aktor mengklik bagian “Status Surat” pada sidebar di halaman beranda
3. Setelah aktor mengklik bagian tersebut, sistem menampilkan menu “Surat Masuk”, “Surat Keluar”, dan “Legalisir Ijazah”
4. Aktor mengklik menu yang ingin dilihat status suratnya.
5. Sistem menampilkan halaman status surat.
6. Aktor mengklik ikon “Mata” yang ada di kolom aksi.
7. Sistem menampilkan halaman informasi surat yang berisikan informasi mengenai posisi surat.

*¤ Alternative Flow*

1. Tidak Ada.

*¤ Exception Flow*

1. Tidak Ada

*Post-condition(s) :*

1. Aktor mengetahui informasi mengenai posisi surat

*Rule(s) :*

1. Tidak Ada

*Constraint(s)* :

1. Tidak Ada

*GUI (optional)* :

1. ***Use case description* Mengelola Akun Pengguna**

*Name of the use case :* Mengelola Akun Pengguna

*Brief description :* Usecase ini menyediakan fungsionalitas kepada Resepsionis untuk mengelola akun yang ada didalam sistem, termasuk menambahkan, mengubah, dan menghapus..

*Characteristic of activation :* Resepsionis

*Pre-condition(s) :*

1. Aktor memiliki akun
2. Aktor berada di halaman “Manajemen Pengguna”

*Flow of Event*

*¤ Basic Flow*

1. Aktor berada di halaman beranda
2. Aktor mengklik bagian “Manajemen Pengguna” pada sidebar di halaman beranda
3. Setelah aktor mengklik bagian tersebut, sistem menampilkan data-data pengguna. Data-data ini disimpan dan ditampilkan pada halaman “Manajemen Pengguna” dalam bentuk tabel yang berisi data username, nama, unit kerja dan aksi. Selain itu, terdapat tombol “tambah user” untuk menambahkan user.
4. Jika aktor ingin menambahkan pengguna, maka aktor mengklik tombol “tambah user” maka aktor dialihkan ke halaman form tambah pengguna. Di dalam form tersebut terdiri dari input username, nama user,  password, unit kerja. Setelah mengisi form, aktor mengklik “Tambah”. Setelah itu data sudah berhasil ditambahkan di dalam database.
5. Jika aktor mengklik ikon “pensil”, maka sistem akan mengalihkan ke halaman form edit data user.  Aktor mengganti data yang ingin digantikan. Jika sudah selesai, aktor mengklik tombol “Simpan”. Data yang sudah diedit akan diupdate ke dalam database.
6. Jika aktor mengklik ikon “sampah”, maka sistem akan menampilkan pesan “Apakah anda yakin ingin menghapus data ini di dalam database?”, dan dua buah tombol yang terdiri dari “Ya” dan “Kembali”. Jika aktor mengklik tombol “Ya”, maka user akan dihapuskan di dalam database. Jika aktor mengklik tombol “Kembali”, maka aktor akan dialihkan kembali ke halaman Manajemen Pengguna.

*¤ Alternative Flow*

1. Tidak Ada.

*¤ Exception Flow*

1. Tidak Ada

*Post-condition(s) :*

1. Data akun pengguna sudah berhasil ditambahkan dan diperbarui di dalam database

*Rule(s) :*

1. Tidak Ada

*Constraint(s)* :

1. Tidak Ada

*GUI (optional)* :