

FR-TS

Functional Requirement: FR.005 - Manajemen kehadiran Siswa dan Pengajar

1. Pendahuluan

Dokumen ini merinci persyaratan fungsional untuk **FR.005 - Manajemen kehadiran Siswa dan Pengajar** dalam proyek Aplikasi Manajemen Kelas/Yayasan. Fitur ini bertujuan untuk mendigitalisasi dan menyederhanakan proses absensi, memungkinkan **Pengajar** dan **Ketua Kelas** yang ditunjuk untuk mencatat status kehadiran siswa, serta mencatat kehadiran Pengajar sebagai dasar perhitungan honor.

2. Deskripsi Fungsional

Sistem harus menyediakan mekanisme pencatatan kehadiran yang berbeda untuk Siswa dan Pengajar/Staf.

- **Untuk Siswa:** Hanya Pengajar/Ketua Kelas yang memiliki jadwal/hak di kelas tersebut yang dapat mencatat absensi antarmuka daftar siswa per kelas, dengan opsi status (Hadir, Izin, Sakit, Alfa, Mustami'ah)
- **Untuk Pengajar/Staf:**
 - **Pengajar :** Pengajar (Mandiri) Hanya Pengajar yang dapat mencatat kehadiran dirinya sendiri untuk sesi mengajar yang dijadwalkan. Mekanisme **Check-In/Check-Out** yang dibatasi waktu (15 menit sebelum hingga 30 menit setelah jam mulai sesi).

3. User Stories

A. Kehadiran Siswa

- Sebagai **Pengajar**, saya ingin dapat mengakses daftar siswa dan mencatat kehadiran mereka dengan opsi **Hadir, Izin, Sakit, Alfa, Mustami'ah**.
- Sebagai **Ketua Kelas**, saya dapat membantu Pengajar mencatat kehadiran teman-teman sekelas saya.
- Sebagai **Siswa**, saya ingin melihat riwayat kehadiran saya sendiri.

B. Kehadiran Pengajar

- Sebagai **Pengajar**, saya ingin mencatat kehadiran saya sendiri (**Check-In**), dibatasi dalam periode waktu yang relevan dengan sesi mengajar saya, agar jam kerja saya tercatat untuk perhitungan honor.
- Sebagai **Pengajar**, saya ingin melihat riwayat kehadiran saya sendiri, sehingga saya dapat memverifikasi catatan jam mengajar saya.
- Sebagai **admin**, saya ingin melihat laporan ringkasan kehadiran semua Pengajar/Staf

untuk evaluasi kinerja.

4. Tampilan Antarmuka Pengguna (UI)

UI 4.1: Daftar Sesi Mengajar Hari Ini

- **Fungsionalitas Utama:** Menampilkan daftar sesi kelas yang dijadwalkan hari ini. Akses ke menu absensi dimulai dari sini.
- **Peran Pengguna:** Pengajar, Ketua Kelas.

UI 4.2: Catat Kehadiran Siswa

- **Fungsionalitas Utama:** Menampilkan daftar siswa untuk kelas yang dipilih, dengan *toggle* atau *dropdown* untuk status: Hadir, Izin, Sakit, Alfa, mustami'ah.
- **Peran Pengguna:** pengajar, ketua kelas.

UI 4.3: Check-In/Check-Out Pengajar

- **Fungsionalitas Utama:** Tombol *Check-In* yang aktif hanya saat sesi mengajar sedang atau akan dimulai.
- **Peran Pengguna:** Pengajar.

UI 4.4: Riwayat Kehadiran (Siswa/Pengajar)

- **Fungsionalitas Utama:** Menampilkan tabel riwayat kehadiran berdasarkan filter tanggal/bulan.
- **Peran Pengguna:** Pengajar, Siswa.

5. Aturan Bisnis / Validasi

Elemen Data / Input	Deskripsi Aturan Bisnis / Validasi	Peran Pengguna
Siswa: Status Lengkap	Absensi Siswa harus memiliki pilihan status: Hadir, Izin, Sakit, Mustami'ah .	Pengajar/Ketua Kelas
Siswa: Waktu Absen	Absensi Siswa harus diselesaikan dalam batas waktu (setelah sesi kelas berakhir).	Sistem
Pengajar: Absensi Mandiri	Tombol " Absen Hadir " hanya aktif pada rentang waktu 15 menit sebelum	Sistem

	dan 15 menit sesudah jadwal kelas dimulai.	
Integrasi Gaji	Data kehadiran Pengajar (mengajar dan staf) adalah komponen wajib dalam perhitungan Gaji Pengajar.	Sistem

6. Persyaratan Non-Fungsional (NFRs)

- **Keamanan:** Akses ke pencatatan absensi harus dilindungi oleh otorisasi peran yang ketat.
- **Kinerja:** Proses pencatatan absensi harus sangat cepat (< 1 detik).
- **Usability:** Antarmuka harus sederhana dan berfungsi optimal di perangkat seluler.

7. Ketergantungan

- **FR.001 - (Manajemen Pengguna):** Diperlukan data Pengajar dan Siswa yang aktif.
- **FR.003 (Manajemen Kelas dan Siswa):** Diperlukan untuk daftar Siswa per kelas.
- **FR.004 - (Manajemen Jadwal Pelajaran):** Diperlukan data Jadwal Pelajaran yang aktif, Jam Mulai, dan Durasi untuk memicu sesi presensi dan memvalidasi waktu Check-In.
- **FR.011 (Perhitungan Gaji Pengajar):** Data kehadiran Pengajar adalah input utama.
- **Basis Data Skema Kehadiran:** Harus mendukung dua jenis pencatatan (Siswa vs. Pengajar/Staf).

8. Asumsi

- Jadwal sesi mengajar sudah diinput dengan akurat ke dalam sistem (**FR.004**).

9. Isu Terbuka

- Bagaimana mekanisme verifikasi jika Pengajar mengklaim hadir tetapi absen tidak terekam (masalah teknis)?

10. Pertimbangan Masa Depan

- Penerapan *reminder* otomatis kepada Pengajar yang belum menyelesaikan absensi Siswa.

11. Alur/Flowchart

graph TD

```

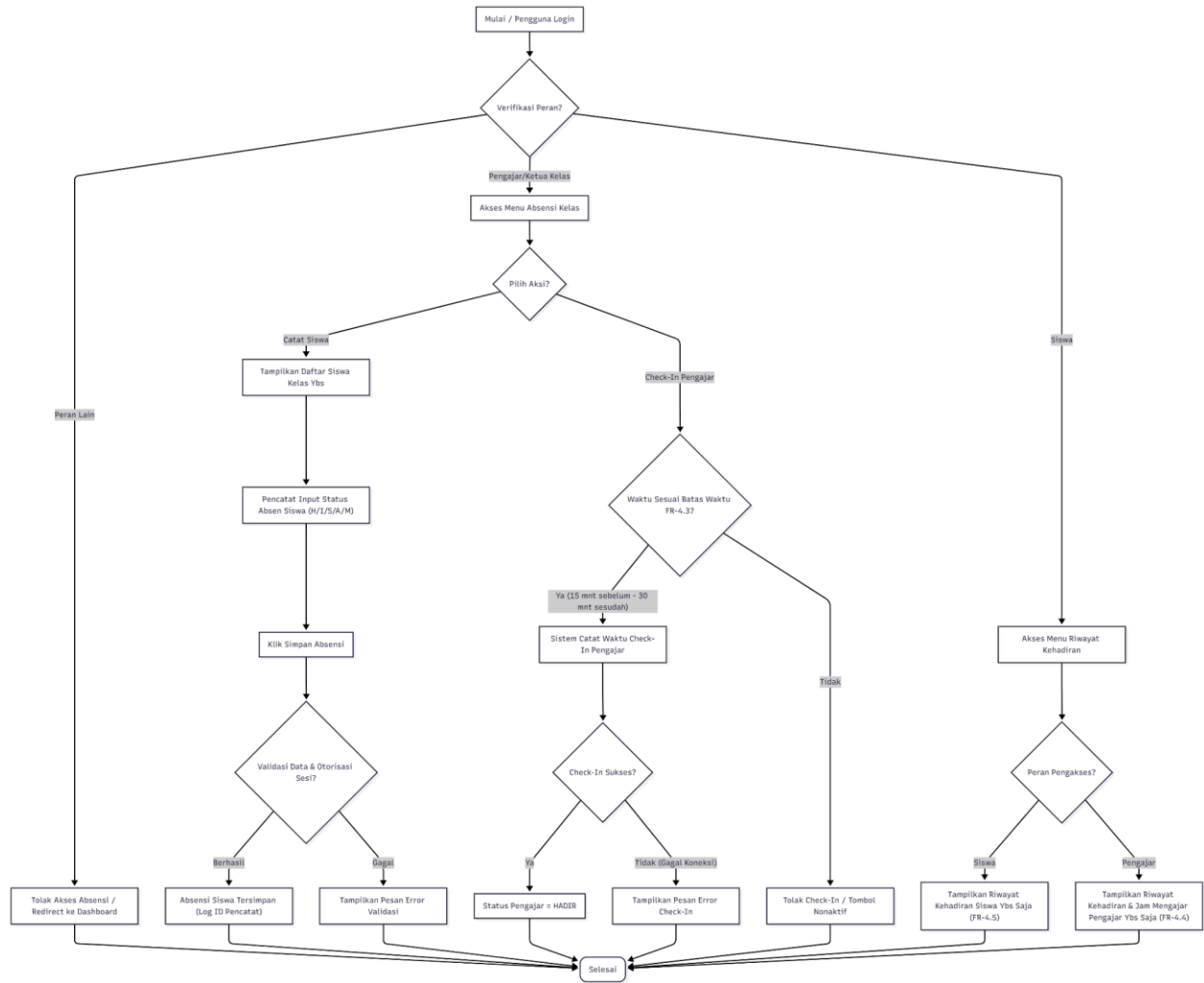
A[Mulai / Pengguna Login] --> B{Verifikasi Peran?};
%% --- JALUR OTORISASI ---
B -- Pengajar/Ketua Kelas --> C[Akses Menu Absensi Kelas];
B -- Siswa --> F[Akses Menu Riwayat Kehadiran];

```

```

B -- Peran Lain --> R[Tolak Akses Absensi / Redirect ke Dashboard];
R --> Z{Selesai};
%% --- JALUR PENCATATAN (Pengajar/Ketua Kelas) ---
C --> D{Pilih Aksi?};
%% Sub-Jalur 1: Catat Kehadiran Siswa (Ketua Kelas dan Pengajar)
D -- Catat Siswa --> G[Tampilkan Daftar Siswa Kelas Ybs];
G --> H[Pencatat Input Status Absen Siswa (H/I/S/A/M)];
H --> I[Klik Simpan Absensi];
I --> J{Validasi Data & Otorisasi Sesi?};
J -- Berhasil --> K[Absensi Siswa Tersimpan (Log ID Pencatat)];
J -- Gagal --> L[Tampilkan Pesan Error Validasi];
K --> Z;
L --> Z;
%% Sub-Jalur 2: Check-In Pengajar (Hanya Pengajar)
D -- Check-In Pengajar --> M{Waktu Sesuai Batas Waktu FR-4.3?};
M -- Ya (15 mnt sebelum - 30 mnt sesudah) --> N[Sistem Catat Waktu Check-In Pengajar];
N --> O{Check-In Sukses?};
O -- Ya --> P[Status Pengajar = HADIR];
P --> Z;
O -- Tidak (Gagal Koneksi) --> Q[Tampilkan Pesan Error Check-In];
Q --> Z;
M -- Tidak --> S[Tolak Check-In / Tombol Nonaktif];
S --> Z;
%% --- JALUR RIWAYAT ---
F --> T{Peran Pengakses?};
T -- Siswa --> U[Tampilkan Riwayat Kehadiran Siswa Ybs Saja (FR-4.5)];
U --> Z;
T -- Pengajar --> V[Tampilkan Riwayat Kehadiran & Jam Mengajar Pengajar Ybs Saja (FR-4.4)];
V --> Z;

```



Test Scenario: TS.005 - Manajemen Kehadiran Siswa dan Pengajar

1. Pendahuluan

Dokumen ini merinci skenario pengujian untuk Functional Requirement **FR.005 - Manajemen Kehadiran Siswa dan Pengajar**. Pengujian ini memverifikasi fungsionalitas utama yang melibatkan pencatatan absensi Siswa (oleh Pengajar/Ketua Kelas), pencatatan kehadiran mandiri Pengajar (Check-In), dan validasi otorisasi akses data.

2. Prasyarat (Pre-conditions)

Data dan status yang diasumsikan sudah tersedia dan dikonfigurasi di sistem:

- 1. **Pengguna & Kelas:**
 - o Pengajar A (Mengajar Kelas X).
 - o Siswa B (Terdaftar di Kelas X dan ditunjuk sebagai Ketua Kelas X).
 - o Siswa C (Terdaftar di Kelas X).
 - o Siswa D (Terdaftar di Kelas Y, bukan di Kelas X).
 - o Pengguna Non-Otorisasi (misal: Admin, Staf Administrasi, atau Pengajar/Siswa dari kelas lain).
- 2. **Jadwal Aktif (FR.004):**
 - o Tersedia Jadwal aktif untuk Kelas X yang diampu oleh Pengajar A (misal: Senin, 08:00 - 09:30).

3. Skenario Pengujian (Test Cases)

TS-CASE-001: Pencatatan Kehadiran Siswa oleh Pengajar

Kategori	Detail
Tujuan Pengujian	Memastikan Pengajar dapat mencatat absensi Siswa dengan semua status yang tersedia (Hadir, Izin, Sakit, Alfa, Mustami'ah) untuk kelas yang diajarnya.
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Pengajar A. 2. Akses menu Absensi untuk Kelas X, sesi mengajar hari ini (Asumsi waktu saat ini valid untuk

	absensi). 3. Catat status Siswa B sebagai Hadir . 4. Catat status Siswa C sebagai Izin . 5. Klik "Simpan Absensi". 6. Verifikasi data absensi Siswa B dan C di Riwayat Absensi.
Hasil yang Diharapkan (Expected Result)	Sistem menampilkan pesan sukses. Absensi Siswa B terekam "Hadir" dan Siswa C terekam "Izin" untuk sesi Kelas X tersebut, dan Pengajar A tercatat sebagai pencatat.
Status Pengujian	To Do
Catatan/Defek ID	

TS-CASE-002: Pencatatan Kehadiran Siswa oleh Ketua Kelas (Otorisasi Akses)

Kategori	Detail
Tujuan Pengujian	Memastikan Ketua Kelas memiliki hak akses untuk mencatat absensi, namun hanya untuk kelas yang diampunya.
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Siswa B (Ketua Kelas X). 2. Akses menu Absensi untuk Kelas X, sesi mengajar hari ini. 3. Coba akses menu Absensi untuk Kelas Y (yang bukan kelasnya). 4. Catat status Siswa C sebagai Sakit di Kelas X. 5. Klik "Simpan Absensi".
Hasil yang Diharapkan (Expected Result)	Akses ke menu Absensi Kelas X berhasil. Akses ke Absensi Kelas Y ditolak (atau Kelas Y tidak muncul). Absensi Siswa C terekam sebagai "Sakit", dan Siswa B tercatat sebagai pencatat (log Ketua Kelas).

Status Pengujian	To Do
Catatan/Defek ID	

TS-CASE-003: Check-In Mandiri Pengajar (Skenario Sukses - Dalam Batas Waktu)

Kategori	Detail
Tujuan Pengujian	Memastikan Pengajar dapat mencatat Check-In untuk sesinya, asalkan berada dalam rentang waktu yang valid (15 menit sebelum - 30 menit setelah jam mulai).
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Pengajar A. 2. Akses menu Kehadiran Pengajar (Check-In). 3. Catat Check-In untuk sesi Kelas X (Senin, 08:00) pada waktu 08:15 (15 menit setelah jam mulai). 4. Verifikasi status kehadiran Pengajar A di Riwayat Kehadiran.
Hasil yang Diharapkan (Expected Result)	Check-In berhasil dicatat. Sistem merekam waktu Check-In dan menandai status kehadiran Pengajar A sebagai "Hadir" untuk sesi tersebut.
Status Pengujian	To Do
Catatan/Defek ID	

TS-CASE-004: Check-In Mandiri Pengajar (Skenario Gagal - Terlambat)

Kategori	Detail
Tujuan Pengujian	Memastikan sistem menolak Check-In jika dilakukan melebihi batas waktu (lebih dari

	30 menit setelah jam mulai sesi).
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Pengajar A. 2. Akses menu Kehadiran Pengajar (Check-In). 3. Coba Catat Check-In untuk sesi Kelas X (Senin, 08:00) pada waktu 08:31 (31 menit setelah jam mulai). 4. Verifikasi pesan sistem.
Hasil yang Diharapkan (Expected Result)	Sistem menolak Check-In dan menampilkan pesan kesalahan: "Check-In gagal. Waktu yang dipilih telah melampaui batas waktu yang diizinkan (08:30)." Status kehadiran Pengajar A untuk sesi ini ditandai sebagai "Alfa".
Status Pengujian	To Do
Catatan/Defek ID	

TS-CASE-005: Verifikasi Riwayat Kehadiran Siswa (Pembatasan Data)

Kategori	Detail
Tujuan Pengujian	Memastikan Siswa hanya dapat melihat riwayat kehadiran mereka sendiri dan tidak dapat melihat data kehadiran Siswa lain.
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Siswa C. 2. Akses UI Riwayat Kehadiran Siswa. 3. Cek riwayat kehadiran yang ditampilkan. 4. Coba cari/akses data kehadiran Siswa B atau Siswa D (Kelas Y).
Hasil yang Diharapkan (Expected Result)	Siswa C berhasil melihat riwayat kehadirannya sendiri (termasuk status dari TS-CASE-001/002). Upaya untuk melihat data kehadiran Siswa B atau Siswa D ditolak/data tidak muncul.

Status Pengujian	To Do
Catatan/Defek ID	

TS-CASE-006: Verifikasi Otorisasi Akses (Pengguna Non-Otorisasi)

Kategori	Detail
Tujuan Pengujian	Memastikan Pengguna selain Pengajar atau Ketua Kelas tidak dapat mengakses antarmuka pencatatan absensi siswa/pengajar.
Langkah-langkah Pengujian (Test Steps)	1. Login sebagai Pengguna Non-Otorisasi (misal: Admin). 2. Coba akses UI Absensi Siswa. 3. Coba akses UI Check-In Pengajar. 4. Coba temukan opsi/menu "Catat Absensi" di navigasi utama.
Hasil yang Diharapkan (Expected Result)	Pengguna Non-Otorisasi tidak dapat mengakses antarmuka pencatatan absensi. Jika menu terlihat, akses harus ditolak dengan pesan otorisasi.
Status Pengujian	To Do
Catatan/Defek ID	

4. Pasca-kondisi (Post-conditions)

- Log absensi Siswa B dan Siswa C untuk Kelas X pada sesi pengujian telah tersimpan dan diverifikasi.
- Log Check-In Pengajar A untuk sesi mengajar tersebut telah tersimpan (jika Check-In berhasil) atau ditandai sebagai Alfa (jika Check-In gagal).
- Hak akses telah diverifikasi sesuai peran pengguna (Pengajar, Ketua Kelas, Siswa, Admin).

7. Hasil Ringkasan Pengujian

Jumlah Skenario Diuji	-
Jumlah Skenario Lulus	-
Jumlah Skenario Gagal	-
Jumlah Skenario Diblokir	-
Kesimpulan Umum	-