

Dokumentasi Penggunaan Perangkat Lunak

APLIKASI PENGELOLAAN DATA SUMUR

BERBASIS WEB

Title: Aplikasi Pengelolaan Data Sumur Berbasis Web

Project: SK5

Document identifier: KAPSISM/20/29/SK5

Document type: Technical working paper

Version: 0.1 **Date:** 19th November 2023

Author: Adnan Rafiyansyah Majid (140810200029)

Inspected: N/A **Approved:** N/A

Submitted to CM:

CM identifier:

Distribution: N/A

Confidentiality: Secret

Keywords: Aplikasi, Sumur, Web, Cache, Antamuka, Data

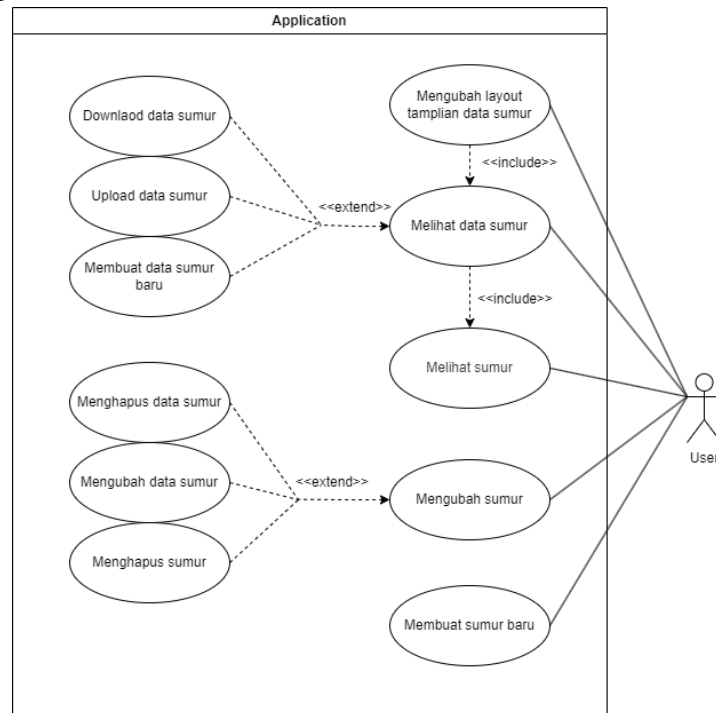
Daftar Isi

1.	Daftar Ilustrasi.....	3
1.1	Use Case Diagram	3
1.2	Data Flow Diagram	3
1.3	Antarmuka Utama	4
2.	Pengantar.....	4
2.1	Tujuan Dokumen	4
2.2	Tujuan Aplikasi	4
3.	Penggunaan Dokumen	4
3.1	Semua Pembaca.....	4
3.2	Pembaca Pengguna Awam	4
3.3	Pembaca Pengguna Berpengalaman.....	4
3.4	Pembaca Pengembang	5
4.	Konsep Operasi	5
4.1	Struktur Data / Unit.....	5
4.1.1	Folder	5
4.1.2	Sumur	5
4.1.3	Informasi.....	5
4.1.4	File Informasi	5
4.2	Antarmuka Aplikasi	5
4.3	Template Informasi Sumur.....	5
4.4	Sinkronisasi dan Penyimpanan Data	6
4.5	Custom Layout	6
4.6	Caching dan Optimization.....	6
4.6.1	Pada Klien	6
4.6.2	Pada Server.....	6
5.	Penggunaan Software	7
5.1	Server (<i>untuk pembaca berpengalaman atau pengembang</i>)	7
5.1.1	Operating Environment	7
5.1.2	Installation	7
5.1.3	Running on development	7
5.1.4	Running on production.....	7
5.2	Client	7
5.2.1	Operating Environment	7
5.2.2	Installation	7

5.2.3	Running.....	7
5.3	Use Cases.....	7
5.3.1	Memilih Folder atau Sumur	7
5.3.2	Memilih Root Folder.....	7
5.3.3	Melihat Sumur	8
5.3.4	Melihat Informasi Sumur.....	8
5.3.5	Mengupload File Informasi.....	8
5.3.6	Mendownload File Informasi.....	8
5.3.7	Menambah Sumur / Folder	8
5.3.8	Menambah Folder.....	8
5.3.9	Menambah Sumur	8
5.3.10	Mengedit Sumur / Informasi	8
5.3.11	Menghapus Sumur	9
5.3.12	Menghapus Folder	9
5.3.13	Menghapus Informasi	9
5.3.14	Membuat Pane Baru Pada Tampilan Data	9
5.3.15	Mengubah Urutan Tab Pada Pane.....	9
5.3.16	Mengubah Ukuran Pane.....	9
5.3.17	Mensinkronisasi Data Sumur dan Folder.....	9
5.3.18	Mensinkronisasi Data Informasi dan Versi File Informasi.....	9
6.	Permasalahan dan Error	9
7.	Glossary	10

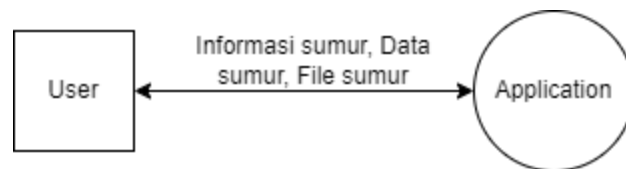
1. Daftar Ilustrasi

1.1 Use Case Diagram

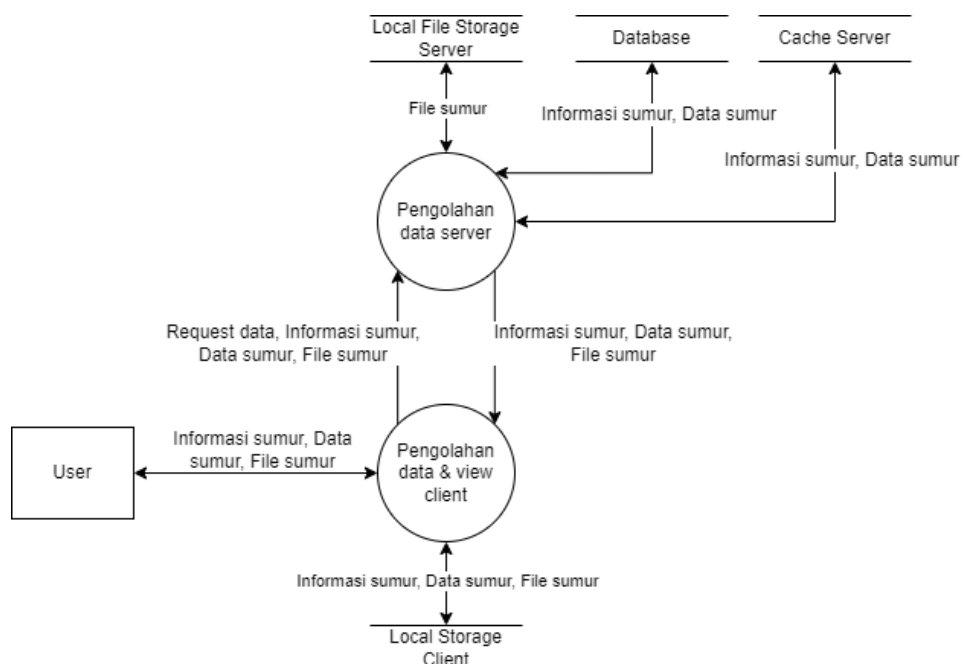


1.2 Data Flow Diagram

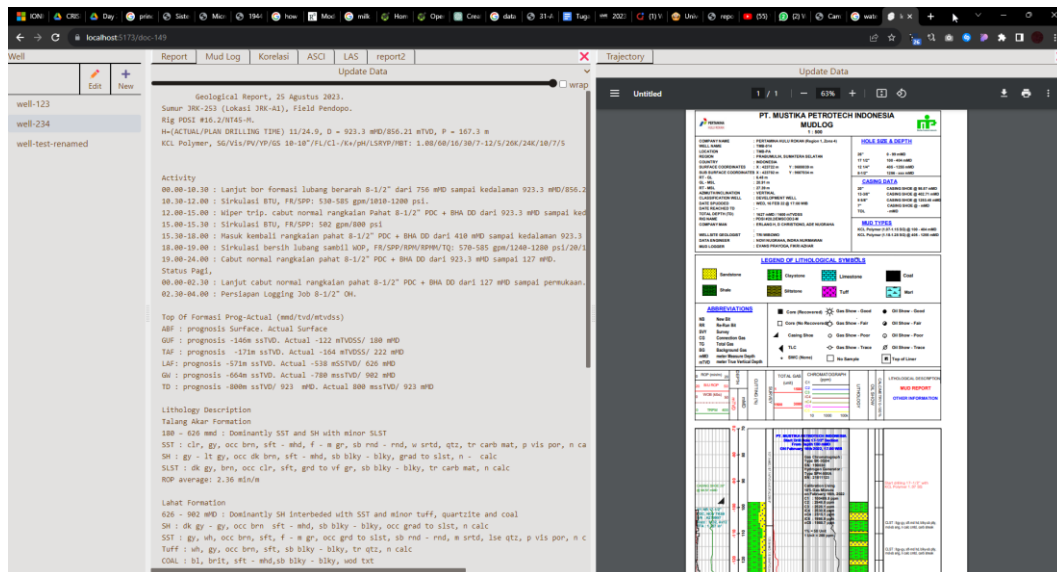
DFD level 0 :



DFD level 1 :



1.3 Antarmuka Utama



2. Pengantar

2.1 Tujuan Dokumen

Dokumen ini terdiri dari beberapa bagian, termasuk pengantar, , daftar ilustrasi, Penggunaan Dokumen, Penggunaan software, Troubleshooting, dan glossary. Setiap bagian dirancang untuk memberikan informasi yang komprehensif dan mendukung pemahaman pengguna terhadap aplikasi pencatatan informasi. Dokumen disusun secara terstruktur untuk memudahkan navigasi. Dengan membaca dan mengikuti panduan ini, diharapkan dapat mengoptimalkan penggunaan Aplikasi sesuai dengan kebutuhan.

2.2 Tujuan Aplikasi

3. Penggunaan Dokumen

Berikut adalah beberapa saran untuk menggunakan dokumen ini secara efektif untuk beberapa macam pembaca :

3.1 Semua Pembaca

- Gunakan daftar isi untuk menavigasi dengan cepat ke bagian yang sesuai dengan kebutuhan.
- Gunakan daftar ilustrasi untuk membantu memahami dasar-dasar aplikasi atau prosedur.
- Baca kembali bagian-bagian sebelumnya untuk memahami hal yang bersangkutan.

3.2 Pembaca Pengguna Awam

- Mulailah dengan membaca dari bagian untuk memahami tujuan aplikasi dan manfaat penggunaannya, kemudian baca semua bagian selanjutnya sampai permasalahan.
- Fokus pada bagian konsep operasi untuk memahami dasar-dasar cara kerja aplikasi sebelum memulai penggunaan.
- Gunakan bagian glossary jika ada istilah yang tidak dipahami.

3.3 Pembaca Pengguna Berpengalaman

- Baca dahulu bagian konsep operasi jika belum paham dasar-dasar cara kerja aplikasi.

- Langsung baca bagian penggunaan software untuk mendapatkan informasi praktis tentang penggunaan aplikasi.
- Baca juga glossary karena ada beberapa istilah yang pengertiannya dikhususkan untuk aplikasi.

3.4 Pembaca Pengembang

- Perhatikan bagian permasalahan dan error untuk lebih memahami cara kerja aplikasi dalam berbagai skenario berbeda.

4. Konsep Operasi

4.1 Struktur Data / Unit

Aliran data dapat dilihat pada bagian ilustrasi (1.2). Terdapat dua data utama yang bersifat nested, yaitu folder dan sumur. Ada juga informasi yaitu informasi dan file informasi.

4.1.1 Folder

- Folder digunakan untuk mengelompokkan sumur, contohnya untuk tiap lokasi atau folder.
- Suatu folder harus memiliki nama dan dapat memiliki banyak sumur atau folder lain.
- Folder dapat dibuat atau disimpan dalam folder lain atau root folder.

4.1.2 Sumur

- Suatu sumur harus memiliki nama dan dapat memiliki banyak informasi.
- Sumur dapat dibuat atau disimpan dalam suatu folder atau root.

4.1.3 Informasi

- Informasi bisa disebut juga informasi karena merupakan data atau informasi dari suatu sumur seperti korelasi, mud log, well report, dan lain-lain.
- Suatu informasi harus memiliki nama dan dapat memiliki satu file informasi.
- Tipe file informasi untuk suatu informasi dapat diatur oleh pengguna.
- Informasi dibuat atau disimpan dalam suatu sumur.

4.1.4 File Informasi

- File informasi adalah file dari suatu informasi.
- Tipe file dapat berupa apa saja atau tertentu sesuai pengaturan pada informasi.
- File dapat diupload versi terbaru ke server oleh pengguna atau file pada klien diupdate ke versi terbaru dari server.

4.2 Antarmuka Aplikasi

Aplikasi merupakan SPA dengan Tampilan utama yaitu tampilan list folder dan sumur serta, disebelahnya, terdapat tampilan data. Selain itu ada juga tombol-tombol fungsional seperti “refresh”, “home”, “add”, dan “edit”. Tampilan data dapat berupa blank (jika tidak ada sumur atau folder yang dipilih), edit folder, edit sumur, tambah data, dan tampilan informasi sumur. Dapat dilihat pada bagian ilustrasi (1.3).

4.3 Template Informasi Sumur

Dalam membuat suatu sumur, akan terdapat template informasi yang jika digunakan, maka yang terbuat bukan sumur kosong (tidak memiliki informasi) tetapi sumur berisi beberapa informasi kosong (tidak memiliki file informasi) sesuai template yang digunakan.

4.4 Sinkronisasi dan Penyimpanan Data

- Data-data yang tersimpan pada klien (browser) merupakan mirror dengan yang ada pada server (dengan asumsi seluruh data telah tersinkronisasi dan file informasi telah diupdate).
- Sinkronisasi data folder dan sumur dengan server akan berjakan jika dilakukan “refresh”.
- Data informasi dan versi file sumur terbaru akan tersinkronisasi saat sumur yang bersangkutan diklik oleh pengguna, akan ada pemberitahuan jika file sumur butuh update.
- Sinkronisasi akan secara otomatis menghapus data-data pada klien yang sudah tidak ada atau dihapus (termasuk file informasi), dan mengupdate data-data yang berubah.
- Sinkronisasi file informasi dilakukan dengan cara update yang akan mendownload file informasi terbaru dari server, atau dengan cara upload yang akan mengupload file informasi terbaru ke server, keduanya akan, secara otomatis, menyimpan file terbaru ke penyimpanan file browser jika berhasil. Berikut beberapa informasi tambahan mengenai file informasi :
 - File tersimpan secara utuh di penyimpanan file klien (browser) dan juga server.
 - File versi lama akan terhapus dan diganti oleh versi baru setelah berhasil update atau upload.
 - File yang tersimpan di klien akan hilang jika dilakukan “clear site data”.
 - File dapat didownload dari penyimpanan file browser ke perangkat pengguna.
- Aplikasi menyimpan layout dalam klien (browser)

4.5 Custom Layout

- Layout aplikasi dapat diubah sesuai keinginan pengguna
- Collapse dan expand folder.
- Tabs dan Panes untuk informasi-informasi masing-masing sumur.
- Size untuk setiap pane dalam aplikasi termasuk pane tampilan utama.
- Semua diatas tersimpan pada klien (browser) termasuk sumur apa yang terakhir dibuka.

4.6 Caching dan Optimization

4.6.1 Pada Klien

Semua data yang dibutuhkan klien untuk berjalannya aplikasi secara lancar tersimpan pada klien, aplikasi hanya akan melakukan “fetch” jika disuruh oleh pengguna (lazy loading). Dengan begini, akan meringankan server dan mempersingkat TTI.

4.6.2 Pada Server

Semua data kecuali file informasi menggunakan strategi write-through caching dengan cache flushing setiap 1 jam untuk menjaga konsistensi data. Hal ini dilakukan untuk mempercepat kinerja overall karena akan jauh lebih banyak operasi read daripada write serta menghindari processing time yang terjadi karena struktur data pada database server (relational) dengan yang diperlukan klien (JSON) sedangkan keduanya adalah mirror (Baca bagian 4.4) dengan data cache pada memory yang berupa Javascript object.

5. Penggunaan Software

5.1 Server (*untuk pembaca berpengalaman atau pengembang*)

5.1.1 Operating Environment

- Platform & version: Node.js 18.16.1
- Database & version: MariaDB 10.5.21
- Storage: 512GB, file aplikasi <1GB
- OS: windows 11
- Memory/RAM: 16GB
- CPU: 8 core
- Network type: internal facing (VPN)

5.1.2 Installation

- Install the correct database and Node.js version
- Run `"npm install"` or your node package manager's equivalent to install the required node modules.
- Make new file `".env"` based on already existing file `"env"` and configure it according to your environment.

5.1.3 Running on development

- Run `"npm run dev"` or your node package manager's equivalent. This is to run the application server on development mode.

5.1.4 Running on production (unavailable)

5.2 Client

5.2.1 Operating Environment

- Platform & version: Google Chrome 117.0+
- Storage: 512GB
- Memory/RAM: 4GB
- CPU: 2 core
- Network type: internal facing (VPN)

5.2.2 Installation (unavailable)

5.2.3 Running

- Run the application's host url on browser.

5.3 Use Cases

5.3.1 Memilih Folder atau Sumur

- Menekan folder atau sumur pada list.
- Folder atau sumur terpilih untuk edit atau add

5.3.2 Memilih Root Folder

- Menekan tombol `"home"`.
- Root folder terpilih untuk edit atau add

5.3.3 Melihat Sumur

- Menekan pada list sumur yang akan di lihat.
- Klien mengambil informasi terbaru dari server dan file-filenya dari penyimpanan klien.
- Tampilan data informasi.

5.3.4 Melihat Informasi Sumur

- Menekan tab informasi yang akan dilihat.
- Tampilan informasi jika didukung dan accordion untuk upload/download data yang berwarna kuning jika data pada klien tidak ada atau bukan merupakan yang terbaru.

5.3.5 Mengupload File Informasi

- Menekan tombol pilih file.
- Memilih file untuk diupload.
- Tampilan tombol upload dan peringatan jika tipe file salah.
- Upload informasi.
- Tampilan Progress upload.
- Tampilan upload gagal / berhasil.
- File disimpan di klien jika berhasil.
- Tampilan update data menjadi tidak kuning dan muncul tampilan informasi.

5.3.6 Mendownload File Informasi

- Menekan tombol update.
- Tampilan progress download.
- File disimpan di klien jika berhasil.
- Tampilan update data menjadi tidak kuning dan muncul tombol “download as file” dan “open in new tab” jika data terbaru berhasil terdownload.

5.3.7 Menambah Sumur / Folder

- Menekan folder tempat data baru akan ditambahkan atau body kosong untuk root.
- Menekan tombol tambah.
- Tampilan untuk menambahkan sumur dan folder.

5.3.8 Menambah Folder

- Mengisi nama folder.
- Konfirmasi menambahkan folder.
- Folder baru ditampilkan di list

5.3.9 Menambah Sumur

- Mengisi nama sumur.
- Memilih preset atau template sumur.
- Konfirmasi menambahkan sumur.
- Sumur baru ditampilkan di list

5.3.10 Mengedit Sumur / Informasi

- Menekan tombol edit informasi.
- Tampilan tombol untuk menghapus sumur.

- Tampilan untuk memilih data yang ingin diedit, dihapus, atau menambahkan data baru.
- Mengisi nama dan tipe file informasi.
- Konfirmasi edit informasi.

5.3.11 Menghapus Sumur

- Menekan tombol hapus sumur.
- Sumur beserta file-file dan data-datanya terhapus dari server,
- Lalu terhapus juga pada klien jika berhasil.
- Sumur terhapus dari tampilan list.

5.3.12 Menghapus Folder

- Menekan tombol edit folder.
- Tampilan tombol untuk menghapus folder.
- Menekan tombol hapus folder.
- Folder beserta sumur-sumurnya beserta file-file dan data-data masing-masing sumur terhapus dari server,
- Lalu terhapus juga pada klien jika berhasil.
- Folder beserta isinya terhapus dari tampilan list.

5.3.13 Menghapus Informasi

- Menekan tombol hapus informasi.
- Informasi beserta filenya terhapus dari server,
- Lalu terhapus juga pada klien jika berhasil.
- Informasi terhapus dari tampilan list.

5.3.14 Membuat Pane Baru Pada Tampilan Data

- Drag and drop tab informasi pada ujung kanan tampilan aplikasi.
- Tab berpindah ke panel baru.

5.3.15 Mengubah Urutan Tab Pada Pane

- Drag and drop tab informasi pada list tab pada panel yang sama atau berbeda.
- Tab berpindah ke panel yang terpilih dan urutan yang sesuai.

5.3.16 Mengubah Ukuran Pane

- Drag gutter sesuai ukuran yang diinginkan.
- Ukuran panel berubah sesuai gutter-nya.

5.3.17 Mensinkronisasi Data Sumur dan Folder

- Menekan tombol “refresh”.

5.3.18 Mensinkronisasi Data Informasi dan Versi File Informasi

- Menekan salah satu sumur.

6. Permasalahan dan Error

(unavailable)

7. Glossary

- Root Folder: Direktori utama atau tingkat tertinggi dalam struktur penyimpanan data.
- Klien: Perangkat atau aplikasi yang terhubung ke server untuk mengakses layanan atau sumber daya.
- Server: Komputer atau sistem yang menyediakan layanan, informasi, atau sumber daya kepada klien melalui jaringan.
- Refresh: Tombol untuk melakukan sinkronisasi pada data sumur dan folder atau proses sinkronisasinya sendiri.
- Nested Object: Objek yang ditempatkan sebagai nilai dari properti objek lainnya, membentuk struktur bertingkat di dalam objek utama.
- Mirror: Salinan identik dari suatu sumber data, sering digunakan untuk tujuan cadangan atau keamanan.
- Sinkronisasi: Proses menyelaraskan data atau status antara dua atau lebih entitas agar konsisten.
- Clear Site Data: Menghapus data penyimpanan lokal pada browser, termasuk cache, cookie, dan data situs, dapat dilakukan dengan tool “inspect element” pada browser.
- Fetch: Proses mengambil data atau sumber daya dari server atau sumber eksternal.
- TTI (Time to Interactive): Metrik yang mengukur waktu yang diperlukan suatu halaman web untuk menjadi sepenuhnya interaktif.
- Relational Database: Jenis basis data yang menggunakan model relasional untuk menyimpan dan mengelola data dalam bentuk tabel terkait.
- JSON (JavaScript Object Notation): Format pertukaran data ringan yang umum digunakan dalam pertukaran data antar aplikasi.
- Write-Through Caching: Strategi caching di mana setiap kali data ditulis ke penyimpanan, juga disalin ke cache.
- Cache Flushing: Menghapus semua atau sebagian data dari cache, memaksa aplikasi untuk mendapatkan data terbaru dari sumber asli.
- Accordion: Desain antarmuka pengguna yang memungkinkan pengguna untuk menyembunyikan atau menampilkan sebagian dari konten, mirip dengan cara aksi accordion pada alat musik.
- Single Page Application (SPA): Aplikasi web yang beroperasi di satu halaman tanpa memuat ulang halaman baru selama interaksi pengguna.
- Lazy Loading: Pengambilan dan pemuatan modul atau komponen secara dinamis hanya saat diperlukan, untuk mengoptimalkan kinerja aplikasi.