

BAB III

ANALISA DAN PERANCANGAN

SISTEM

3.1. Jenis, Sifat, dan Pendekatan Penelitian

Penelitian ini menjelaskan secara lengkap dan terperinci mengenai langkah-langkah yang dilakukan dalam merancang hingga menerapkan sistem *bridging* untuk menghubungkan mesin MRI atau rontgen yang berada di ruang radiologi ke sistem *bridging* yang nantinya berfungsi untuk menyimpan data ke pusat data dan menampilkannya.

3.2. Kebutuhan Fungsional

Kebutuhan fungsional aplikasi ini adalah sebagai jembatan untuk pengembangan selanjutnya atau kedepannya, penelitian ini dirancang untuk membuat sistem *bridging* dari PACS atau alat MRI ke pusat data yang baru menggunakan dicom server.

3.3. Kebutuhan Non Fungsional

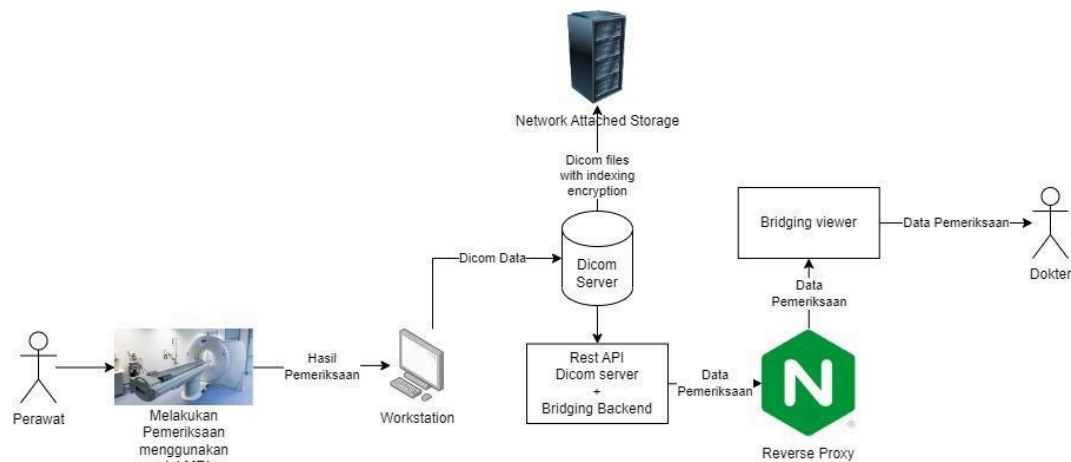
Pada Tabel 3.1 terdapat Kebutuhan Non Fungsional dalam penelitian ini memiliki spesifikasi yang dilakukan pengembangan adalah :

Tabel 3.1
Kebutuhan
Non
Fungsional

Keterangan	Client	Server
CPU	: Intel Pentium 2020M 2 Core	Intel Core i3 2 Core
RAM	: 8GB	4GB
VGA	: AMD Radeon 8500M Series	Tanpa VGA
Sistem Operasi	: Windows 11 Professional	Ubuntu Server

3.4. Arsitektur Sistem

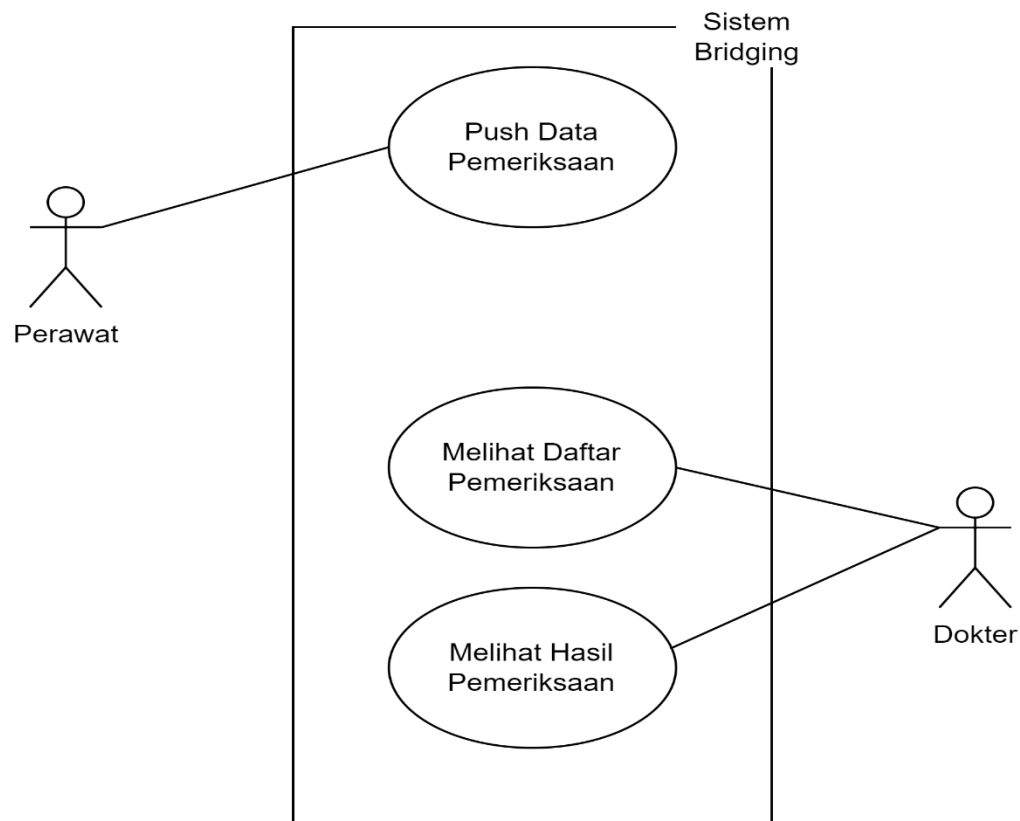
Pada Gambar 3.1 menjelaskan arsitektur dalam penelitian memiliki 2 pengguna yaitu: perawat bertugas untuk membantu pemeriksaan pasien lalu melakukan *push* data hasil pemeriksaan ke *dicom* server untuk disimpan ke penyimpanan pusat, lalu dokter sebagai pengguna yang melakukan diagnosa pasien dari hasil pemeriksaan dengan menggunakan *bridging viewer*. Arsitektur yang dirancang ini memiliki perangkat keras dan perangkat lunak seperti komputer MRI yang memiliki fungsi untuk mendapatkan hasil pemeriksaan dari alat MRI lalu melakukan *push* data ke *dicom* server, *dicom* server sebagai server untuk menerima data hasil pemeriksaan lalu disimpan ke *Network Attached Storage*. *Network Attached Storage* sebagai penyimpanan pusat hasil pemeriksaan yang telah dienkripsi oleh *dicom* server, *Rest API* sebagai antarmuka untuk berkomunikasi dengan *bridging viewer* dengan memanfaatkan protokol *http*, *nginx* sebagai *reverse proxy* untuk mengatasi masalah *CORS* (*Cross Origin Resource Sharing*), dan *bridging viewer* untuk mengambil data pemeriksaan dari pusat dan menampilkannya ke dokter.



Gambar 3.1 Arsitektur Sistem

3.5. Use Case Diagram

Pada Gambar 3.2 merupakan *Use Case Diagram* yang terdapat 2 pengguna dan fungsinya, pengguna yang pertama adalah perawat atau petugas medis dalam ruang radiologi, perawat berperan dengan tugas untuk melakukan *push data* pemeriksaan ke sistem *bridging* yang telah dibuat, hasil pemeriksaan yang masuk tersebut akan diakses oleh dokter dari melihat daftar pemeriksaan berupa nama pasien, rekam medis, dan tanggal pemeriksaan hingga hasil pemeriksaan yang berupa gambar *CT scan*.

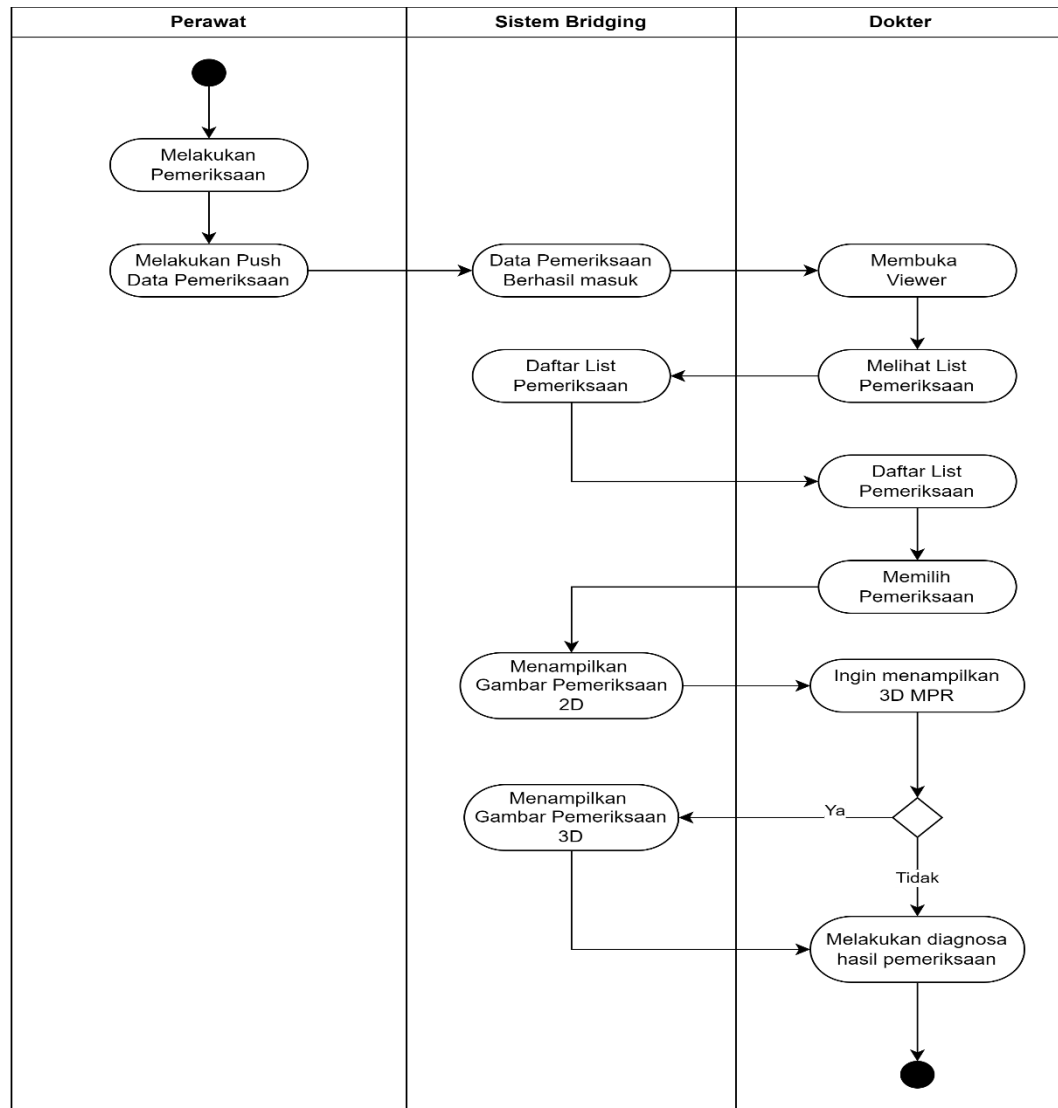


Gambar 3.2 Use Case Diagram

3.6. Activity Diagram

Pada Gambar 3.3 merupakan *activity diagram* atau diagram aktivitas yang dilakukan oleh perawat, sistem bridging, dan dokter. Aktivitas dimulai dengan perawat yang melakukan pemeriksaan, setelah pemeriksaan dilakukan perawat melakukan *push data* pemeriksaan ke sistem bridging, lalu ketika dokter ingin melihat daftar list pemeriksaan, maka dokter akan membuka *viewer*, *viewer* tersebut akan menampilkan riwayat pemeriksaan dari banyaknya pemeriksaan yang telah dilakukan, selanjutnya dokter akan memilih pemeriksaan, ketika telah memilih pemeriksaan, sistem bridging akan menampilkan hasil *rontgen* pemeriksaan berupa 2D, apabila dokter ingin menampilkan 3D maka akan memasuki proses menampilkan 3D MPR yang dilakukan

oleh sistem *bridging*, setelah beberapa proses telah dilakukan yang terakhir adalah dokter melakukan diagnosa pasien.



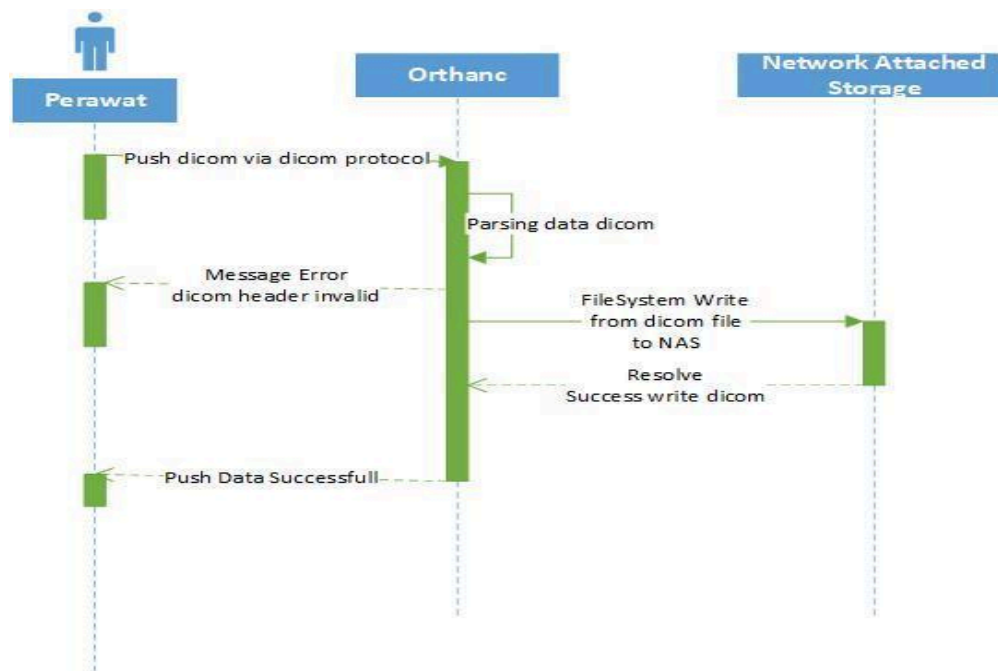
Gambar 3.3 Activity Diagram

3.7. Sequence Diagram

3.7.1. Push Data

Pada Gambar 3.4 menjelaskan diagram atau alur ini berfungsi untuk menyimpan data *dicom* menggunakan protokol *dicom* yang disediakan oleh *dicom* server atau *orthanc* yang *default* konfigurasinya berjalan pada *port* 4242. Petugas radiologi berperan untuk melakukan rontgen pada pasien, setelah dilakukan rontgen dengan alat MRI data *dicom* akan masuk ke *local data* pada komputer MRI lalu petugas tersebut melakukan push data yang harus memiliki label "*Patient ID*" sebagai nomor rekam medis pasien dan "*Study ID*" sebagai ID kasus pada pasien dengan perangkat

lunak yang disediakan oleh vendor alat rontgen atau MRI menggunakan protokol dicom, data yang dikirim tersebut akan diterima *orthanc dicom* server pada *bridging backend* yang hasilnya akan disimpan ke dalam penyimpanan NAS berupa *mapping* data dan direktori, kemudian *header dicom* disimpan sebagai *indexing* setelah push berhasil, *dicom* server tidak akan mengembalikan *callback* kecuali ketika terjadi *error*.

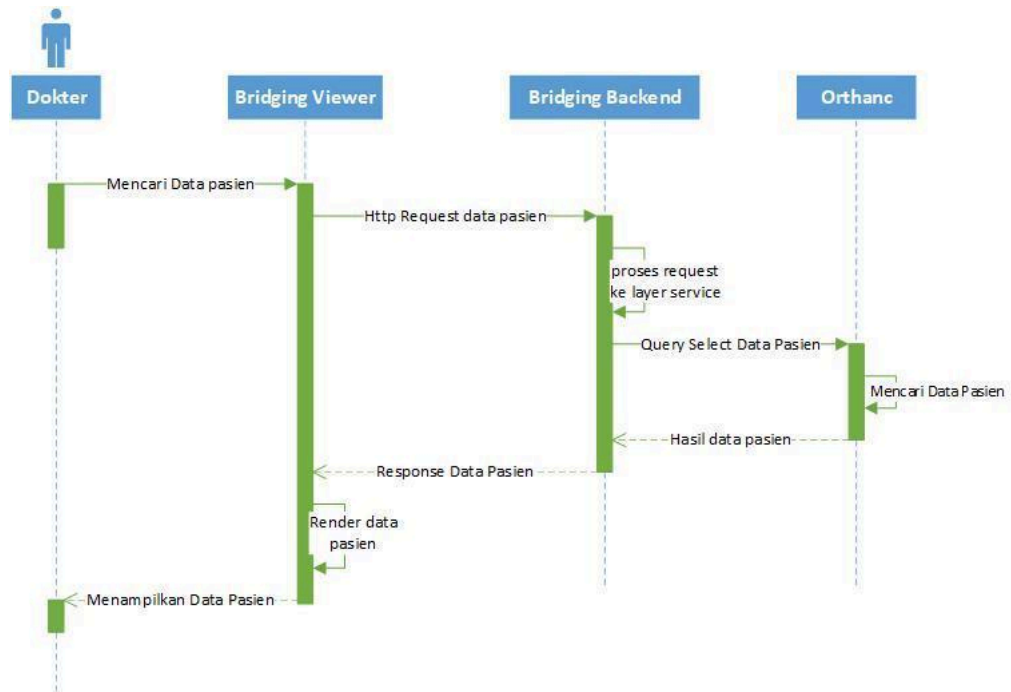


Gambar 3.4 Diagram Push Data

3.7.2. Melihat Data

Pada Gambar 3.5 menjelaskan diagram atau alur melihat data ini berfungsi untuk mengambil data dari *bridging viewer* atau antarmuka aplikasi *web* yang berfungsi untuk menampilkan data dari *dicom* server.

Diagram ini diperankan oleh *user* atau pengguna aplikasi yaitu dokter yang memiliki akses dengan menggunakan nomor rekam medis, *bridging viewer* mengambil data menggunakan protokol *http* atau dapat disebut komunikasi menggunakan konsep REST API (*Representational State Transfer Application Programming Interface*) yang dilakukan *http request* ke *bridging backend*, kemudian *http request* tersebut diproses dan dilakukan *get* data ke *dicom* server dan diteruskan oleh *bridging backend* lalu mengembalikan *http response* yang akan dilakukan *render component* oleh *bridging viewer* dan dapat dilihat gambar *rontgen* tersebut oleh dokter.



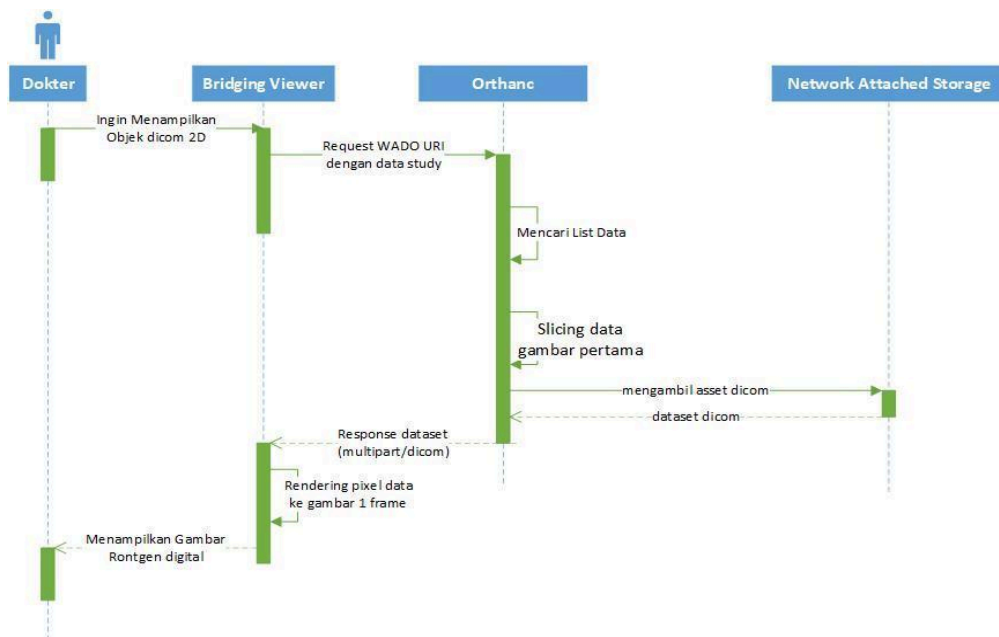
Gambar 3.5 Diagram Melihat Data

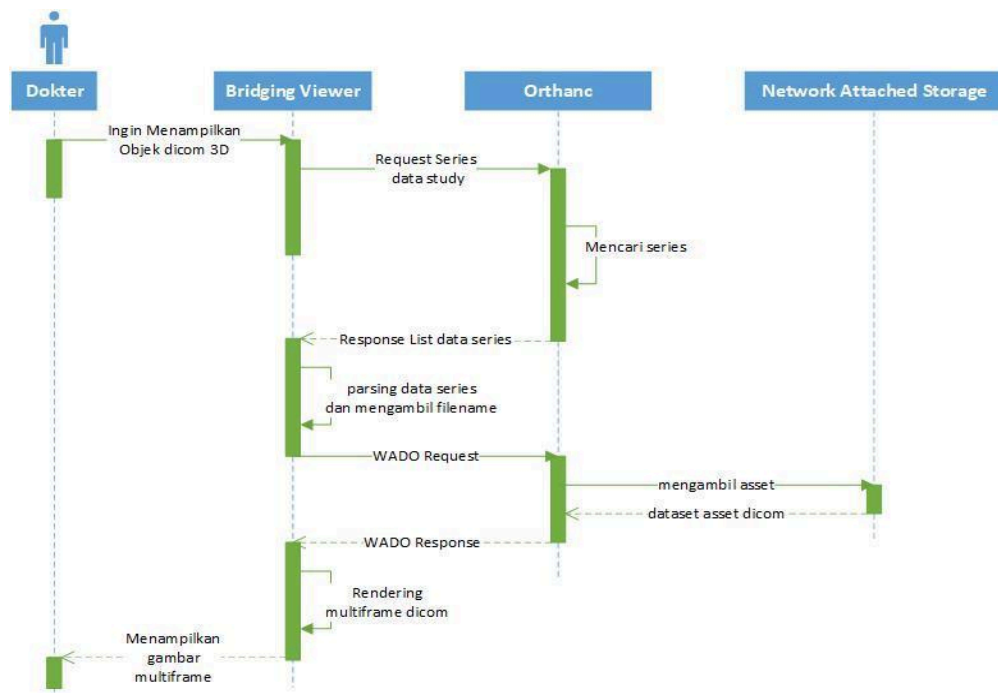
3.7.3. Prosesing Gambar 2 Dimensi dan 3 Dimensi

Pada Gambar 3.6 menjelaskan diagram atau alur pemrosesan gambar 2 dimensi dan pada Gambar 3.7 menjelaskan diagram atau alur pemrosesan 3 dimensi, gambar tersebut merupakan alur dari proses pengambilan data pemeriksaan berbentuk *dicom* yang ditampilkan secara 2 dimensi menggunakan WADO URI (*Web Access Dicom Object Uniform Resource Identifier*) dan 3 dimensi menggunakan WADO (*Web Access Dicom*) *Request* yang dilakukan *request* hingga seluruh potongan gambar dipenuhi agar dapat melanjutkan proses penggabungan objek 3 dimensi.

Diagram dimulai ketika dokter sudah memilih *study dicom* proses dilanjutkan oleh *bridging viewer* yang melakukan WADO (*Web Access Dicom Object*) menggunakan protokol *http*, proses diterima oleh *bridging backend* lalu diproses untuk mencari data studi yang terkait ketika data ditemukan maka akan merespon file *dicom* atau *pixel data dicom* dengan *response* "application/dicom", *response* tersebut akan diolah oleh *bridging viewer* dan dilakukan *rendering component* agar dokter dapat melihat data hasil rontgen yang telah disimpan.

Gambar 3.6 Diagram Prosesing 2 Dimensi



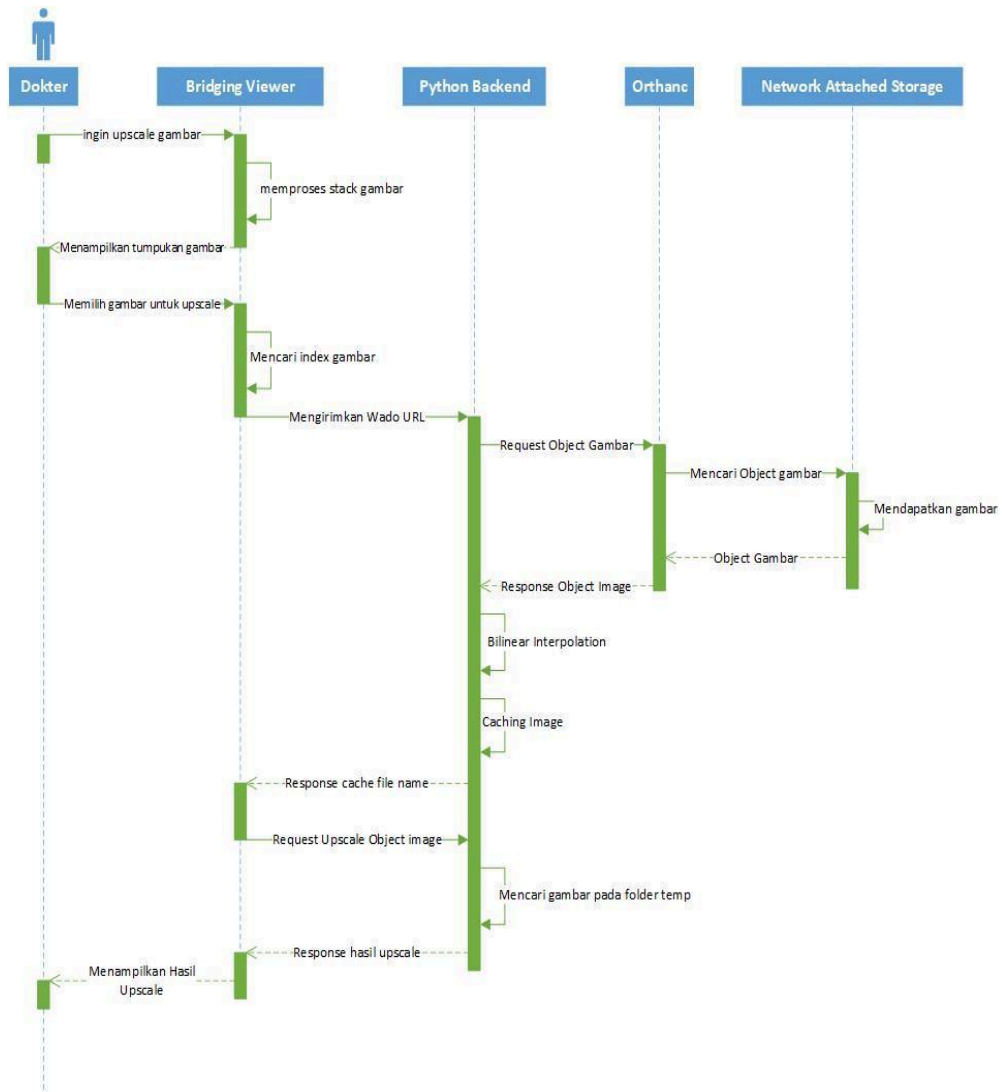


Gambar 3.7 Diagram prosesing gambar 3 dimensi

3.7.4. Bilinear Interpolation

Bilinear Interpolation merupakan metode mengisi poin dengan rata-rata 4 *pixel* terdekat. Dalam metode ini kita dapat melakukan interpolasi linear 2 arah yaitu arah vertikal dan arah horizontal. Kita membutuhkan fungsi untuk melakukan kalkulasi pada suatu titik *grid* atau titik tengah suatu tabel data *pixel* (Prajapati et al., 2012).

Dalam Gambar 3.8 *Bilinear Interpolation* dimulai Ketika dokter ingin melakukan upscale agar gambar hasil pemeriksaan dapat diperbesar dengan ukuran yang cukup, dokter memulai dengan menekan sebuah tombol upscale, lalu *Bridging Viewer* melakukan *http request* yang berisi *WADO URI dicom*, data request tersebut akan dilakukan *request ulang* oleh python ke *orthanc dicom server*, lalu *orthanc dicom server* akan mengembalikan objek gambar, objek gambar yang telah dikembalikan oleh *orthanc dicom server* akan dilakukan proses *Bilinear Interpolation* oleh *backend python*, kemudian gambar tersebut dilakukan *caching* atau disimpan sementara di dalam folder "temp" yang ada pada direktori *backend python*, kemudian *backend python* mengembalikan *http response* ke *bridging viewer* berupa nama file hasil *caching*, dan terakhir *bridging viewer* menampilkan hasil dari *upsampling* tersebut dengan cara memanggil objek gambar tersebut dan ditampilkan ke halaman aplikasi.



Gambar 3.8 Sequence Bilinear Interpolation

3.8. Perancangan Interface

Perancangan *interface* merupakan tahapan dalam membuat struktur sebuah tampilan aplikasi dalam penelitian ini. Adapun tahapan perancangan adalah sebagai berikut :

3.8.1. Interface Halaman Pemeriksaan

Pada Gambar 3.9 terdapat rancangan *interface* halaman pemeriksaan yang merupakan halaman pertama yang akan dibuka oleh dokter halaman ini berisi beberapa komponen seperti komponen pertama adalah *input* nama pasien untuk pencarian nama pasien, lalu komponen kedua adalah *input* tanggal pemeriksaan untuk pencarian tanggal pemeriksaan, dan komponen ketiga atau terakhir adalah tabel yang berisi kolom nomor untuk urutan data. Kolom nama yaitu berisi daftar nama pasien yang telah dimasukkan hasil pemeriksaannya, kolom rekam medis yang berisi nomor rekam medis yang telah disediakan oleh rumah sakit, kolom tanggal pemeriksaan berisi tanggal pasien melakukan pemeriksaan, dan yang terakhir adalah kolom deskripsi yang berisi tentang deskripsi yang telah dimasukkan oleh perawat.

Daftar Studi

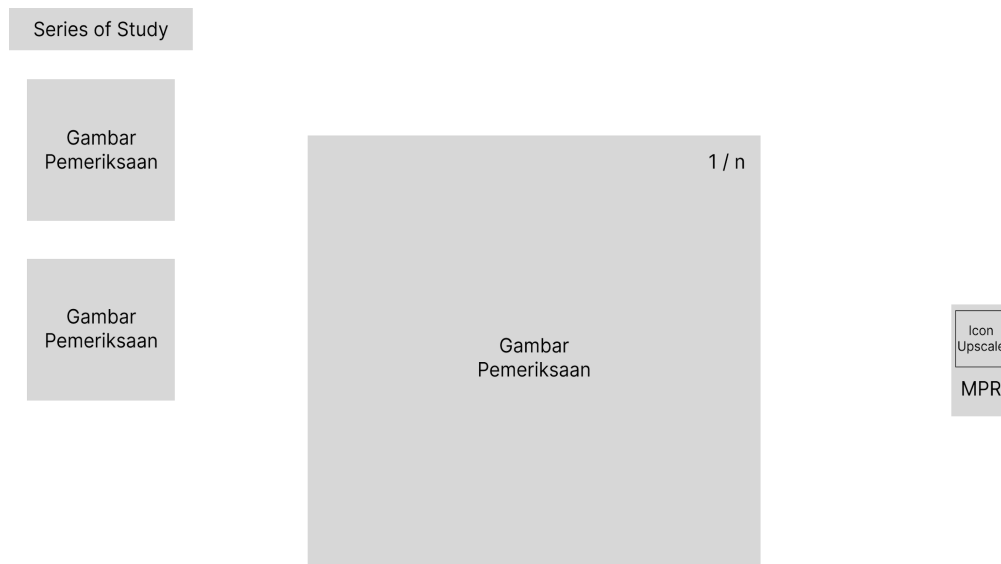
No	Nama Pasien	Rekam Medis	Tanggal Pemeriksaan	Deksripsi
██	██████	██████	██████████	██
██	██████	██████	██████████	██
██	██████	██████	██████████	██

Gambar 3.9 Perancangan Interface Halaman Pemeriksaan

3.8.2. Interface Halaman 2D

Pada Gambar 3.10 terdapat rancangan *interface* halaman 2D yang merupakan halaman ketika baris pemeriksaan. Pada halaman pemeriksaan dilakukan klik sehingga aplikasi akan mengalihkan tampilan pemeriksaan tersebut pada halaman 2D. Halaman 2D berisi beberapa komponen seperti *Series of Study* yang berisi daftar *thumbnail* gambar dari beberapa seri pemeriksaan pada sebuah studi, komponen kedua adalah gambar pemeriksaan, gambar ini muncul ketika *thumbnail* dari *series of study* dilakukan klik maka tampilan gambar pemeriksaan akan muncul dilengkapi dengan urutan gambar yang terdapat pada bagian kanan atas, dan komponen terakhir adalah tombol proses yang ada di samping kanan yang berisi *icon upscale* untuk proses *upscaling* menggunakan *bilinear interpolation* dan di bawahnya tombol MPR untuk mengalihkan ke halaman 3D MPR.

Gambar 3.10 Perancangan Interface Halaman 2D



3.8.3. Interface Halaman 3D MPR

Pada Gambar 3.11 terdapat rancangan *interface* halaman 3D MPR yang merupakan halaman untuk melakukan pemrosesan gambar *dicom* menjadi gambar 3 dimensi yang dipotong menjadi 3 bagian yaitu bagian pertama adalah potongan *axial*. Potongan *axial* sendiri merupakan bentuk potongan dari pemeriksaan yang terlihat dari wujud vertikal atau dilihat dari atas, potongan yang kedua adalah *sagittal* potongan ini merupakan bentuk wujud organ manusia yang dilihat dari samping, dan potongan yang terakhir adalah *coronal*. Potongan *coronal* sendiri merupakan potongan pemeriksaan yang menampilkan hasil pemeriksaan yang dilihat dari depan organ seperti depan perut, dan depan wajah.



Gambar 3.11 Perancangan Interface 3D MPR

