

**PEMBANGUNAN APLIKASI WEB IPOSYANDU BERBASIS
MICROSERVICE DENGAN NEST JS
STUDI KASUS PADA LAYANAN POSYANDU KADER**

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan Seminar Kualifikasi Penelitian
Program Studi Teknik Informatika Universitas Pasundan

Oleh:

Chandra Arcychan Azfar
NRP. 20.304.0151



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
JULI 2024**

**LEMBAR PENGESAHAN
LAPORAN TUGAS AKHIR**

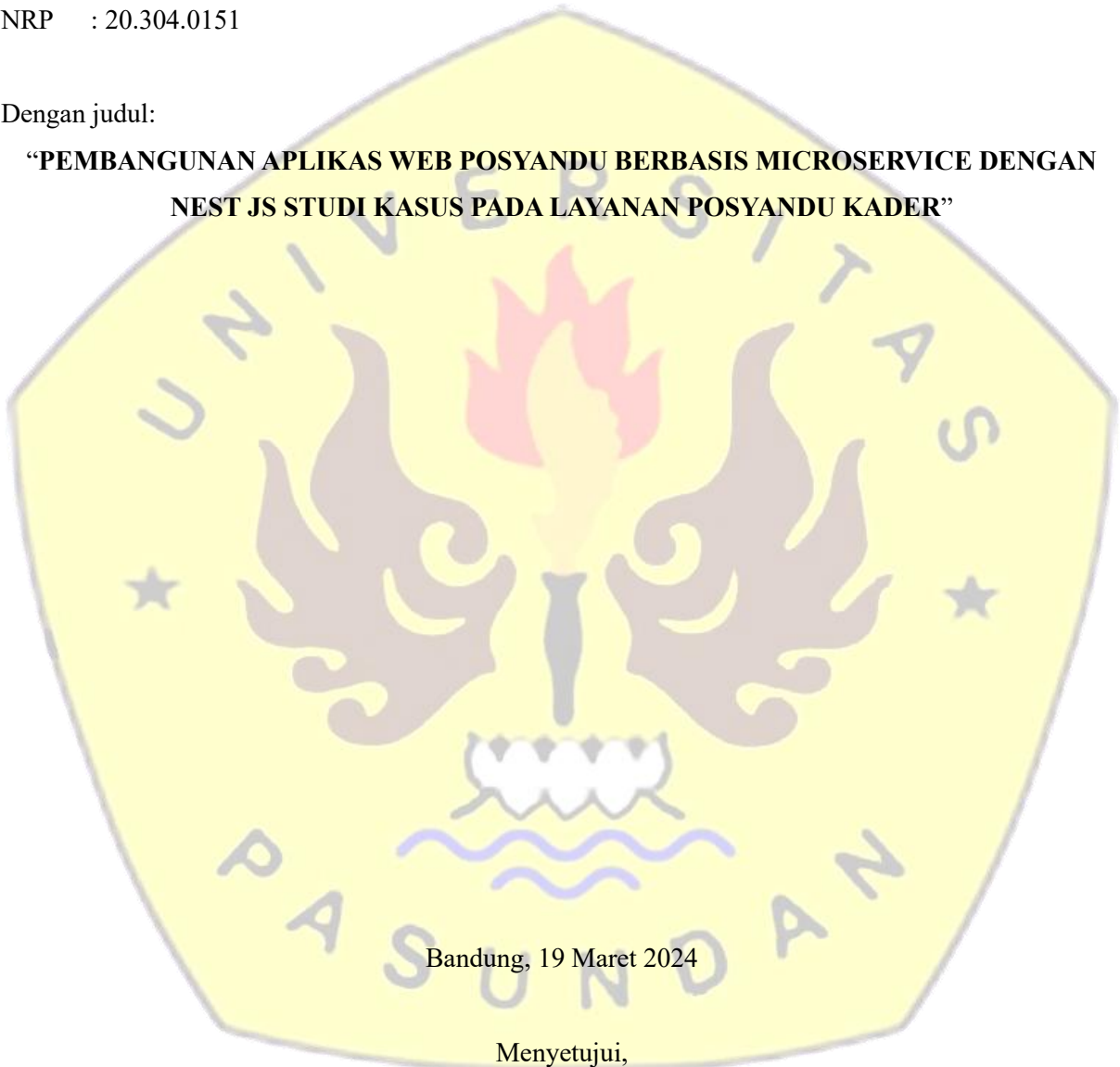
Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang tugas akhir dari:

Nama : Chandra Arcychan Azfar

NRP : 20.304.0151


Dengan judul:

**“PEMBANGUNAN APLIKAS WEB POSYANDU BERBASIS MICROSERVICE DENGAN
NEST JS STUDI KASUS PADA LAYANAN POSYANDU KADER”**



Bandung, 19 Maret 2024

Menyetujui,
Pembimbing Utama


Dr. Ayi Purbasari, ST., MT.

ABSTRAK

Posyandu, sebagai bagian dari Upaya Kesehatan Bersumber Daya Masyarakat, memainkan peran vital dalam penyediaan layanan kesehatan dasar untuk masyarakat. Namun, pengelolaan data yang masih manual seringkali menghasilkan ketidakakuratan dan kesulitan akses bagi kader kesehatan. Untuk mengatasi hal ini, iPosyandu dikembangkan sebagai aplikasi yang berpotensi mempermudah pengelolaan data. Namun, saat ini, iPosyandu masih menggunakan model monolith yang dapat menyulitkan manajemen kompleksitas aplikasi. Oleh karena itu, pendekatan mikroservices diusulkan sebagai solusi alternatif untuk meningkatkan manajemen dan skalabilitas aplikasi iPosyandu.

Penelitian ini bertujuan untuk membangun arsitektur Microservices pada website iPosyandu dan mengimplementasikan framework Nest.js. Langkah-langkahnya meliputi pengumpulan informasi melalui observasi, wawancara, dan studi literatur sebagai tahap awal. Selanjutnya, pembangunan aplikasi dilakukan dengan menggunakan metodologi Scrum untuk memastikan pengembangan yang terstruktur dan terukur.

Setelah melalui tahapan tersebut, hasil yang diperoleh termasuk pengembangan API dan arsitektur microservice yang diimplementasikan pada website iPosyandu. Dengan pendekatan ini, diharapkan adanya perubahan signifikan dalam manajemen dan pengelolaan situs iPosyandu, memungkinkan peningkatan fungsionalitas secara independen pada setiap layanan, serta mempermudah perluasan dan pembaruan situs di masa mendatang tanpa mengganggu keseluruhan sistem.

Kata kunci : Posyandu, iPosyandu, Monolith, Microservice, Scrum, Api

ABSTRACT

Posyandu, as part of Community Resource Health Efforts, plays a vital role in providing basic health services to the community. However, manual data management often results in inaccuracies and access difficulties for health cadres. To overcome this, iPosyandu was developed as an application that has the potential to make data management easier. However, currently, iPosyandu still uses a monolith model which can make it difficult to manage application complexity. Therefore, a microservices approach is proposed as an alternative solution to improve the management and scalability of the iPosyandu application.

This research aims to build a Microservices architecture on the iPosyandu website and implement the Nest.js framework. The steps include collecting information through observation, interviews, and literature study as an initial stage. Furthermore, application development is carried out using the Scrum methodology to ensure structured and measurable development.

After going through these stages, the results obtained included API development and microservice architecture which was implemented on the iPosyandu website. With this approach, it is hoped that there will be significant changes in the management and management of the iPosyandu site, allowing for independent improvements in functionality for each service, as well as making it easier to expand and update the site in the future without disrupting the entire system.

Keywords: Posyandu, iPosyandu, Monolith, Microservice, Scrum, Api

DAFTAR ISI

LEMBAR PENGESAHAN.....	ii
LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR.....	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI.....	vii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xiii
DAFTAR ISTILAH.....	xvi
DAFTAR SIMBOL	xvii
DAFTAR LAMPIRAN	xix
BAB 1 PENDAHULUAN.....	1-1
1.1 Latar Belakang.....	1-1
1.2 Identifikasi Masalah	1-2
1.3 Tujuan Tugas Akhir	1-2
1.4 Lingkup Tugas Akhir.....	1-2
1.5 Metodologi Tugas Akhir.....	1-2
1.6 Sistematika Penulisan Tugas Akhir	1-4
BAB 2 LANDASAN TEORI DAN PENELITIAN TERDAHULU.....	2-1
2.1 Teori Pendukung.....	2-1
2.1.1 iPosyandu	2-1
2.1.2 Web	2-1
2.1.3 Backend.....	2-2
2.1.4 Kader.....	2-2
2.1.5 Microservice.....	2-3
2.1.6 Nest Js	2-4
2.1.7 Postgres SQL.....	2-5
2.1.8 RabbitMQ.....	2-5
2.1.9 Docker.....	2-5
2.1.10 Scrum	2-6
2.2 Penelitian Terdahulu	2-6
BAB 3 SKEMA PENELITIAN	3-1
3.1 Alur Penyelesaian Tugas Akhir	3-1
3.2 Perumusan Masalah.....	3-2
3.2.1 Analisis Sebab Akibat	3-2

3.2.2	Analisis Solusi Masalah	3-3
3.3	Kerangka Pemikiran Teoritis	3-4
3.3.1	Gambaran Produk Tugas Akhir	3-4
3.4	Profil Penelitian	3-5
BAB 4	ANALISIS DAN PERANCANGAN	4-1
4.1	Analisis Kebutuhan Perangkat Lunak	4-1
4.1.1	Analisis Pengguna Saat Ini.....	4-1
4.1.2	Analisis Kebutuhan Pengguna	4-3
4.1.3	Analisis Kebutuhan Fungsional	4-4
4.1.4	Analisis Kebutuhan Non Fungsional.....	4-4
4.1.5	Pemodelan Berbasis Skenario	4-5
4.1.5.1	Diagram <i>Use Case</i>	4-5
4.1.5.2	Deskripsi <i>Use Case Diagram</i>	4-7
4.1.5.3	Skenario <i>Use Case</i>	4-8
4.1.5.4	Diagram <i>Swimlane</i>	4-29
4.1.6	Pemodelan Data	4-38
4.1.6.1	Objek Data.....	4-39
4.1.6.2	Atribut Data.....	4-39
4.1.6.3	Relasi	4-43
4.1.6.4	Pemodelan Berbasis Perilaku	4-1
4.1.6.5	Mengidentifikasi Event-Event.....	4-1
4.1.7	Diagram Sequence	4-1
4.1.8	Pemodelan Berbasis Kelas	4-11
4.1.8.1	Mengidentifikasi Kelas-Kelas Analisis	4-11
4.1.8.2	Mengidentifikasi Atribut-Atribut.....	4-14
4.1.8.3	Mengidentifikasi Operasi-Operasi.....	4-19
4.1.8.4	Perancangan Kelas.....	4-22
4.1.9	Perancangan Arsitektur Aplikasi	4-23
4.2	Scrum Proses	4-24
4.2.1	Product Backlog	4-24
4.2.2	<i>Sprint</i>	4-25
4.2.2.1	<i>Sprint</i> ke -1	4-25
4.2.2.2	<i>Sprint</i> ke-2	4-26
4.2.2.3	<i>Sprint</i> ke-3	4-27

4.2.2.4	<i>Sprint</i> ke-4	4-28
4.2.3	<i>Sprint Retrospective</i>	4-29
BAB 5 Implementasi perangkat lunak		5-1
5.1	Konstruksi Perangkat Lunak	5-1
5.1.1	Kakas dan Kebutuhan Perangkat Lunak	5-1
5.1.2	Spesifikasi Perangkat Lunak	5-1
5.2	Pengkodean	5-1
5.2.1	Struktur Library	5-2
5.2.2	Struktur Kode	5-2
5.2.3	Struktur Apps	5-2
5.2.3.1	Auth	5-3
5.2.3.2	Bayi	5-4
5.2.3.3	Ibu Hamil	5-5
5.2.3.4	Wuspus	5-6
5.2.3.5	Posyandu	5-8
5.2.4	Routing	5-8
5.3	Rancangan Pengujian Sistem	5-9
5.3.1	Rancangan Pengujian Modul Auth	5-10
5.3.1.1	Cek Fungsional Registrasi Akun Kader	5-10
5.3.1.2	Cek Fungsional Login	5-10
5.3.2	Rancangan Pengujian Modul Bayi/balita	5-10
5.3.2.1	Cek Fungsional Menampilkan Data Bayi/Balita	5-10
5.3.2.2	Cek Fungsional Menambah Data Bayi/Balita	5-10
5.3.2.3	Cek Fungsional Mengubah Data Bayi/Balita	5-10
5.3.2.4	Cek Fungsional Menghapus Data Bayi/Balita	5-10
5.3.2.5	Cek Fungsional Menambahkan Data Imunisasi Bayi/Balita	5-10
5.3.2.6	Cek Fungsional Mengubah Data Imunisasi Bayi/Balita	5-10
5.3.2.7	Cek Fungsional Menambahkan Data Pengukuran Data Bayi/Balita	5-10
5.3.2.8	Cek Fungsional Mengubah Data Pengukuran Bayi/Balita	5-11
5.3.2.9	Cek Fungsional Menambah Data Kematian Bayi/Balita	5-11
5.3.3	Rancangan Pengujian Modul Ibu Hamil V	5-11
5.3.3.1	Cek Fungsional Menampilkan Data Ibu Hamil	5-11
5.3.3.2	Cek Fungsional Menambah Data Ibu Hamil	5-11

5.3.3.3	Cek Fungsional Mengubah Data Ibu Hamil	5-11
5.3.3.4	Cek Fungsional Menghapus Data Ibu Hamil	5-11
5.3.3.5	Cek Fungsional Menambahkan Data Imunisasi Ibu Hamil	5-11
5.3.3.6	Cek Fungsional Mengubah Data Imunisasi Ibu Hamil	5-11
5.3.3.7	Cek Fungsional Menambahkan Data Pengukuran Data Ibu Hamil	5-11
5.3.3.8	Cek Fungsional Mengubah Data Pengukuran Ibu Hamil	5-11
5.3.3.9	Cek Fungsional Menambahkan Data Persalinan Data Ibu Hamil	5-12
5.3.3.10	Cek Fungsional Mengubah Data Persalinan Ibu Hamil	5-12
5.3.3.11	Cek Fungsional Menambah Data Kematian Ibu Hamil	5-12
5.3.4	Rancangan Pengujian Modul Wuspus	5-12
5.3.4.1	Cek Fungsional Menampilkan Data Wuspus	5-12
5.3.4.2	Cek Fungsional Menambah Data Wuspus	5-12
5.3.4.3	Cek Fungsional Mengubah Data Wuspus	5-12
5.3.4.4	Cek Fungsional Menghapus Data Wuspus	5-12
5.3.4.5	Cek Fungsional Menambahkan Data Imunisasi Wuspus	5-12
5.3.4.6	Cek Fungsional Mengubah Data Imunisasi Wuspus	5-12
5.3.4.7	Cek Fungsional Menambahkan Data Pemeriksaan Data Wuspus	5-12
5.3.4.8	Cek Fungsional Mengubah Data Pemeriksaan Wuspus	5-13
5.3.4.9	Cek Fungsional Menambah Data Kematian Wuspus	5-13
5.3.5	Rancangan Pengujian Modul Posyandu	5-13
5.3.5.1	Cek Fungsional Menampilkan Data Kegiatan Posyandu	5-13
5.3.5.2	Cek Fungsional Menambah Data Kegiatan Posyandu	5-13
5.3.5.3	Cek Fungsional Mengubah Data Kegiatan Posyandu	5-13
5.3.5.4	Cek Fungsional Mengunduh Data Bayi/Balita	5-13
BAB 6 KESIMPULAN DAN SARAN		6-1
6.1	Kesimpulan	6-1
6.2	Saran	6-1
DAFTAR PUSTAKA		xiv
LAMPIRAN		xiv

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Posyandu merupakan salah satu bentuk Upaya Kesehatan Bersumber Daya Masyarakat(UKBM) yang dikelola dan diselenggarakan dari, oleh, untuk dan bersama masyarakat dalam penyelenggaraan pembangunan kesehatan, guna memperdayakan masyarakat dalam penyelenggaraan pembangunan kesehatan, guna memperdayakan masyarakat dan memberikan kemudahan kepada masyarakat dalam memperoleh pelayanan kesehatan dasar, utamanya untuk mempercepat penurunan angka kematian ibu dan bayi[KEM11].

Layanan posyandu, yang seringkali dijalankan oleh kader kesehatan masyarakat, memberikan berbagai pelayanan kesehatan dasar, pemantauan pertumbuhan anak, imunisasi, gizi serta penyuluhan kesehatan kepada warga masyarakat, khususnya pada kelompok balita dan ibu hamil. Pencatatan manual Posyandu seringkali menimbulkan masalah tumpang tindih data, ketidakakuratan informasi, serta kesulitan akses bagi kader. Hal ini juga menyulitkan pengelolaan data akibat jumlah peserta yang besar dan memakan waktu yang tidak efisien[DID23].

Posyandu memiliki peran vital dalam memastikan akses kesehatan dasar yang mudah bagi seluruh masyarakat. Penggunaan internet dapat mempermudah pengelolaan data Posyandu. Untuk mengatasi kendala kader dibuatlah iPosyandu. iPosyandu adalah aplikasi yang berisikan informasi tentang data ibu hamil, data bayi dan balita, data wus pus. Namun, saat ini, situs iPosyandu masih menggunakan arsitektur *monolith*. *Monolith* adalah pendekatan pengembangan perangkat lunak tradisional yang menggunakan satu basis kode untuk menjalankan banyak fungsi bisnis. Dampaknya, seiring dengan bertambahnya kompleksitas aplikasi, kode yang semakin rumit bisa berperilaku secara tak terduga dan sulit untuk dikelola.

Di lain sisi arsitektur *microservices* merupakan alternatif yang muncul dalam pengembangan perangkat lunak. Pada arsitektur *microservice* sistem dirancang untuk menyediakan layanan secara spesifik. Pendekatan ini memecah aplikasi menjadi serangkaian layanan yang independen, masing-masing mengurus satu fungsi bisnis[BLI22]. Hal ini akan lebih memudahkan sistem untuk beradaptasi pada perubahan kebutuhan.

Adapun untuk mendukung arsitektur *Microservice* tersebut dibutuhkan sebuah framework yang bernama Nest Js sebagai pendukung untuk membangun web iPosyandu. Nest.js dipilih karena memiliki fitur *modularitas*, struktural, dan berorientasi objek yang memudahkan pengembangan aplikasi web *Microservice*[NES23]. Framework ini dirancang untuk memfasilitasi pembangunan aplikasi *server side* yang mudah diuji. Perlu dicatat bahwa pengembangan dilakukan menggunakan TypeScript, memberikan keuntungan tipe data statis dan memanfaatkan fitur-fitur terbaru dari JavaScript, yang tidak hanya mendukung pengembangan *microservice* tetapi juga meningkatkan keamanan dan pemeliharaan kode secara keseluruhan.

Berdasarkan latar belakang yang telah diuraikan, judul yang dipilih adalah "Pembangunan Aplikasi Web iPosyandu Berbasis Microservice dengan Nest.js: Studi Kasus pada Layanan Posyandu Kader." Harapannya, pembangunan ini akan memfasilitasi peningkatan *skalabilitas* web iPosyandu dengan lebih baik dari segi teknologi dan memenuhi kebutuhan pengguna dengan lebih optimal.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka permasalahan yang dimunculkan pada penelitian ini adalah

1. Bagaimana membangun website iPosyandu Kader menggunakan Microservices?
2. Bagaimana mengimplementasikan framework Nest js pada Microservices iPosyandu Kader?

1.3 Tujuan Tugas Akhir

Berdasarkan identifikasi masalah tersebut, maka tujuan yang ingin penulis capai pada tugas akhir ini adalah.

1. Membangun Microservices pada website iPosyandu Kader
2. Mengimplementasikan framework Nest js pada Microservices iPosyandu Kader

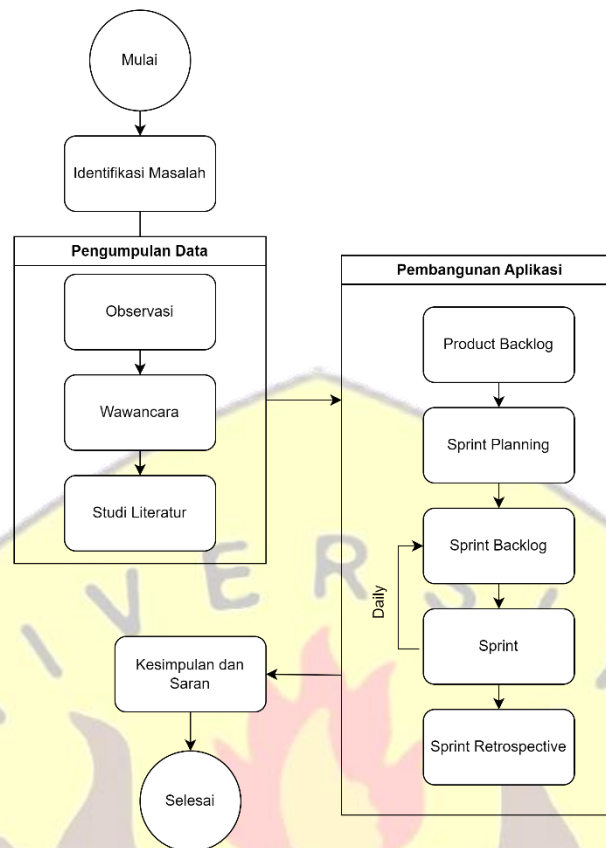
1.4 Lingkup Tugas Akhir

Penyelesaian tugas akhir akan dibatasi oleh beberapa poin sesuai dengan identifikasi dan tujuan dari tugas akhir, sebagai berikut:

1. Aplikasi menggunakan arsitektur microservice dengan tujuan kemudahan pembangunan yang hanya berfokus pada sisi backendnya saja
2. Bahasa pemrograman yang digunakan dalam pembangunan aplikasi adalah Typescript dengan menggunakan framework Nest Js
3. Penerapan metode scrum dalam manajemen proyek

1.5 Metodologi Tugas Akhir

Dalam proses pengumpulan data hingga pembangunan aplikasi iPosyandu menggunakan metode agile. Pengembangan perangkat lunak secara agile adalah pendekatan untuk membangun perangkat lunak dengan kerja sama tim yang kolaboratif. Nilai-nilai dan prinsip yang diterapkan dalam agile bersumber dari dan mendukung kerangka kerja scrum.



Gambar 1.1 Metodologi Penelitian Tugas Akhir

Berikut merupakan rincian dari metodologi tugas akhir ini, diantaranya:

1. Identifikasi Masalah

Tahap identifikasi masalah merupakan tahapan awal dari penelitian, yang dimana pada tahap tersebut dilakukan identifikasi masalah untuk mengetahui masalah yang sedang terjadi di aplikasi iPosyandu.

2. Pengumpulan Data

Pada tahap ini dilakukan observasi aplikasi iPosyandu, penelitian literatur untuk memahami aplikasi iPosyandu dan melakukan wawancara untuk mendapatkan informasi mengenai iPosyandu. Adapun cara-cara pengumpulan data yang dilakukan adalah sebagai berikut :

a. Observasi

Tahap ini mengumpulkan informasi dengan mengamati sistem atau proses iPosyandu yang ada saat ini.

b. Studi Literatur

Tahap ini dilakukan eksplorasi literatur yang didapat dari buku, jurnal ilmiah maupun e-book di internet terkait aplikasi iPosyandu, teknologi yang digunakan dan metode scrum.

c. Wawancara

Tahap ini dilakukan wawancara yang dilakukan dengan pengembang yang terlibat dalam pembuatan aplikasi sebelumnya untuk memperoleh wawasan, pengetahuan

teknis, kebutuhan teknologi, serta umpan balik terkait pengalaman mereka dalam mengembangkan aplikasi iPosyandu sebelumnya.

3. Pembangunan Aplikasi

Pada tahap ini dilakukan pembangunan aplikasi iPosyandu menggunakan metode scrum. Ada beberapa tahap dalam scrum, diantaranya:

a. Product Backlog

Product Backlog adalah daftar yang muncul dan tersusun tentang apa yang dibutuhkan untuk membangun aplikasi iPosyandu. Product backlog ini didapatkan dari hasil pengumpulan data yang sudah dilakukan.

b. Sprint Planning

Sprint Planning berfungsi untuk membahas apa yang akan dilakukan selama sprint. Rencana yang dihasilkan ini dibuat secara kolaboratif oleh seluruh Scrum Team.

c. Sprint Backlog

Sprint Backlog adalah gambaran waktu yang direncanakan developer selama sprint.

d. Sprint

Sprint merupakan periode waktu yang ditetapkan untuk pekerjaan. Dalam setiap periode sprint, yang umumnya berlangsung sekitar satu bulan dalam kalender, dilakukan peninjauan dan penyesuaian terhadap kemajuan menuju tujuan produk untuk memastikan prediktabilitas dalam pengembangan.

e. Sprint Review

Sprint Review bertujuan untuk memeriksa hasil dari Sprint sebelumnya.

f. Sprint Retrospective

Tahap ini merupakan kegiatan rutin di akhir sprint untuk mengevaluasi sprint tersebut dan membuat rencana perbaikan atau peningkatan sprint berikutnya.

4. Kesimpulan dan saran

Pada tahap ini, dibuat kesimpulan dan juga saran untuk meningkatkan kualitas aplikasi iPosyandu dan meningkatkan efisiensi dan efektivitas layanan posyandu.

1.6 Sistematika Penulisan Tugas Akhir

Laporan tugas akhir dibuat untuk mendokumentasikan pengerjaan tugas akhir. Maka dari itu, diusulkan sistematika penulisan yang menjelaskan mengenai bab-bab pada laporan tugas akhir beserta isinya secara rinci, serta keterkaitan antara bab sebelum dan sesudahnya. Adapun sistematika penulisannya adalah sebagai berikut:

BAB 1. PENDAHULUAN

Bab ini berisi penjelasan umum mengenai usulan penelitian yang dilakukan dalam pengerjaan tugas akhir. Di dalamnya berisi latar belakang masalah, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi pengerjaan tugas akhir, dan sistematika penulisan laporan tugas akhir.

BAB 2. LANDASAN TEORI DAN PENELITIAN TERDAHULU

Bab ini berisi definisi, teori-teori serta konsep yang diperlukan dalam pengerjaan tugas akhir. Bab ini juga membahas mengenai jurnal-jurnal ilmiah terdahulu yang memiliki kemiripan dengan tugas akhir yang dikerjakan.

BAB 3. SKEMA PENELITIAN

Bab ini berisi penjelasan alur penyelesaian tugas akhir, analisis persoalan dan manfaat tugas akhir, kerangka pemikiran teoritis, dan profile tempat penelitian.

BAB 4. ANALISIS DAN PERANCANGAN APIKASI

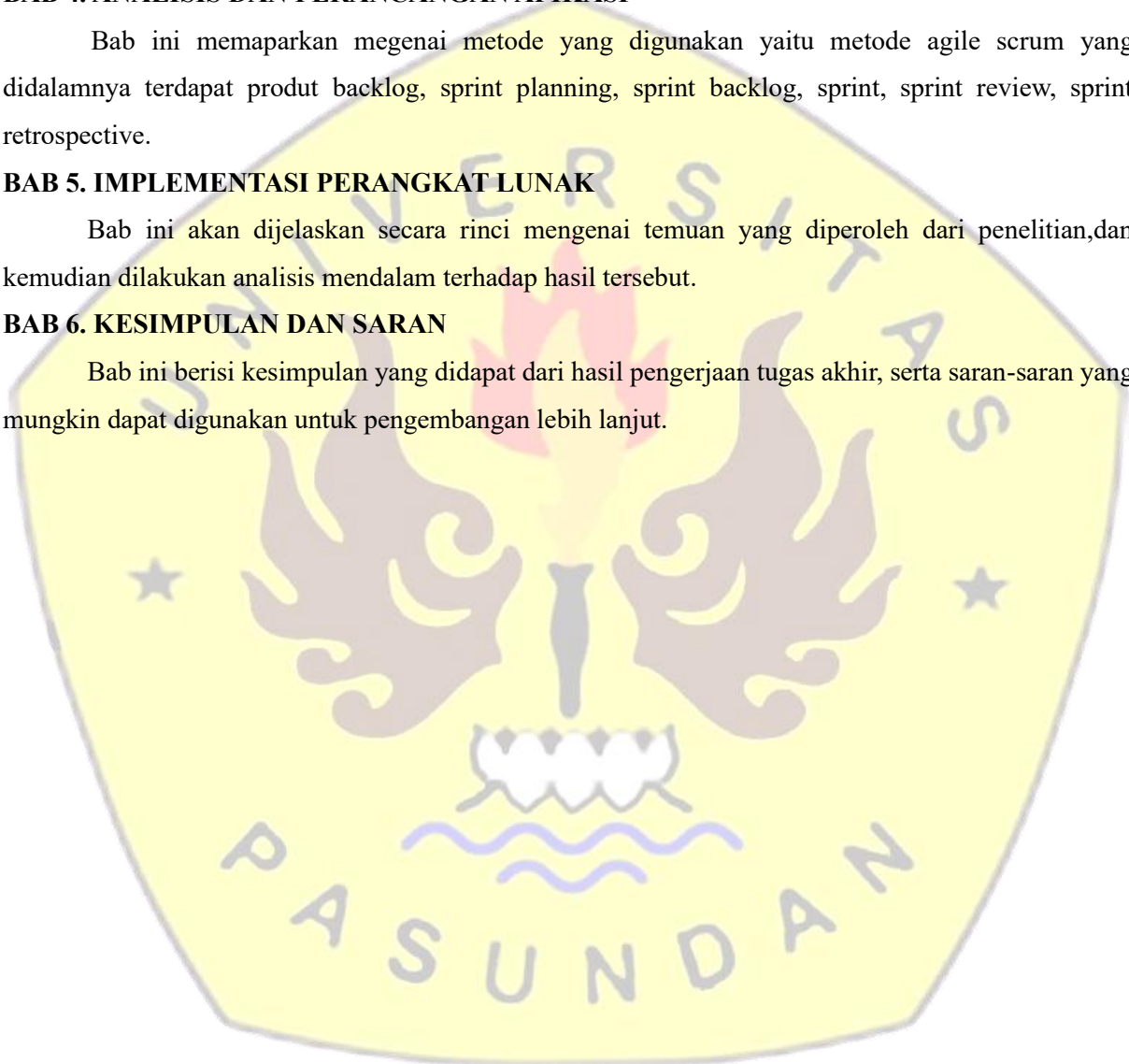
Bab ini memaparkan mengenai metode yang digunakan yaitu metode agile scrum yang didalamnya terdapat produk backlog, sprint planning, sprint backlog, sprint, sprint review, sprint retrospective.

BAB 5. IMPLEMENTASI PERANGKAT LUNAK

Bab ini akan dijelaskan secara rinci mengenai temuan yang diperoleh dari penelitian, dan kemudian dilakukan analisis mendalam terhadap hasil tersebut.

BAB 6. KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan yang didapat dari hasil pengerjaan tugas akhir, serta saran-saran yang mungkin dapat digunakan untuk pengembangan lebih lanjut.



DAFTAR PUSTAKA

- [BIU19] Biundo, J. Authentication and Sessions for MVC Apps with NestJS, (2019) <https://dev.to/nestjs/aSuthentication-and-sessions-for-mvc-apps-with-nestjs-55a4> (accessed 27 September 2023).
- [BLI22] Blinowski, G., Ojdowska, A., Przybylek, A. Monolithic vs. Microservice Architecture: A Performance and Scalability Evaluation. 2022.
- [DID23] Didah, Susanti, A. I., Rinawan, F. R., & Ferdian, D. Peningkatan kapasitas dan pengelola Posyandu melalui inovasi aplikasi iPosyandu. 2023
- [FER20] Ferguson, N. What is the difference between frontend and backend Web Development. 2020. <https://careerfoundry.com/en/blog/web-development/whats-the-difference-between-frontend-and-backend/>. [accessed 27 September 2023].
- [HAD19] Hadji, S., Taufik, M., & Mulyono, S. Implementasi metode Scrum pada pengembangan aplikasi delivery order berbasis website (studi kasus pada rumah makan Lombok Idjo Semarang). 2019
- [HAS22] Hasanudin. Asgar A., Hartono B. Rancangan Bangun Rest Api Aplikasi Weshare sebagai Upaya mempermudah Pelayanan Donasi Kemanusiaan. Jurnal Informatika Teknologi dan Sains vol 4 no 1. 2022
- [JOH23] Johnston, J. A beginner's guide to web application development (2020) <https://www.budibase.com/blog/web-application-development/>. (accessed 27 September 2023).
- [KEM11] Kementerian Kesehatan RI Indonesia. Pedoman Umum Pengelolaan Posyandu. Jakarta: Kementerian Kesehatan RI. 2011
- [MIL23] Milosavljević, M., Matić, M., Jović, N., & Antić, M. (2023). Comparison of Message Queue Technologies for Highly Available Microservices in IoT
- [NES23] NestJS Documentation. 2023, <https://nestjs.com/> (accessed 27 September 2023)
- [NIN21] Ningtyas, P. Perancangan Modul Administrasi Aplikasi Iposyandu Berbasis Web. 2021
- [PAM18] Pamungkas, Ridho (2018) TEORI DAN IMPLEMENTASI PEMROGRAMAN WEB. UNIPMA Press, Madiun
- [PHA20] Pham, Duc, A. Developing back-end of a web application with NestJS framework
- [POS23] Postgresql Documentation. 2023, <https://www.postgresql.org/>, (accessed 28 September 2023)
- [POT20] Potdar, A. M., Narayan, D. G., Kengond, S., & Mulla, M. M. Performance Evaluation of Docker Container and Virtual Machine, Procedia Computer Science.2020
- [RIN19] Rinawan, F.R. Fedri Ruluwedrata Rinawan, dr., M.Sc.PH, PhD, Sosok di Balik Aplikasi “iPosyandu”. 2019. <https://www.unpad.ac.id/profil/fedri-ruluwedrata-rinawan-dr-m-sc-ph-phd-sosok-di-balik-aplikasi-iposyandu/>. (accessed 27 September 2023).

