

BAB II

LANDASAN TEORI

2.1 Tinjauan Penelitian Terdahulu

Penelitian berkaitan dengan sistem informasi gereja telah dilaksanakan oleh beberapa peneliti sebelumnya. Penelitian tersebut dengan judul “Sistem Informasi Gereja HKBP Tanjung Balai Karimun” menggunakan metode *Waterfall* dengan penelitian yang dihasilkan adalah sistem informasi yang membantu pihak gereja dalam mengelola data yang bersangkutan dengan gereja dan jemaat di HKBP Tanjung Balai Karium (Panjaitan and Hasugian, 2018)

Penelitian lainnya berjudul “Implementasi Sistem Informasi Berbasis *Web* (Studi Kasus Gereja GKE Sion Palangkaraya)” menggunakan metode *Waterfall* dengan penelitian yang dihasilkan adalah sistem informasi *web* yang dapat memperkenalkan gereja, memotivasi pelayanan, kemudahan pengurusan pada GKE Sion Palangkaraya, dan jemaat dapat mendownload file yang ingin dibutuhkan (Nugroho and Jayanti, 2017).

Penelitian lainnya berjudul “Perancangan Sistem Informasi Jemaat dan pekerja Gereja Masehi Injili di Minahasa” menggunakan metode DAD (*Disciplined Agile Delivery*) dengan penelitian yang dihasilkan adalah sistem informasi yang dapat memudahkan pengguna mengakses informasi data sensus jemaat dan mencari informasi lain mengenai gereja (Pitoy, Hans and Kom, 2016)

Penelitian lainnya berjudul “Perancangan Sistem Pengolahan Data Jemaat Berbasis *Web* Pada Gereja GKPI Kota Jambi” menggunakan metode *Waterfall*, peneliti menghasilkan sistem yang dapat mengelola data jemaat, data keuangan, dan memudahkan pengguna mendapat informasi mengenai gereja secara online (Dame Christine Sagala & Ali Sadikin, 2018).

Penelitian lainnya berjudul “Perancangan *Website* Gereja Misi Injili Indonesia (GMII) Jemaat Sola Gratia” menggunakan metode *Waterfall* menghasilkan sistem informasi untuk meningkatkan kinerja mengelola data dan pemberian informasi yang cepat dan akurat dari majelis Gereja Misi Injili Indonesia (Yulius, Putra and Darmianto, 2016)

Dan penelitian lainnya berjudul “Perancangan Sistem Informasi Pembukuan UKM Konveksi Bim Collection Berbasis Website “ menggunakan metode *Obeject Oriented Analysis and Design (OOAD)* menghasilkan sistem informasi pembukuan yang terarah dengan rapi berkat adanya metode OOAD yang mencakup dua tahapan yaitu OOA (*Object Oriented Analysis*) dan OOD (*Object Oriented Design*) yang didokumentasikan dengan diagram-diagram UML (*Unified Modeling Language*) seperti *use case* diagram, *sequence* diagram, dan *object* diagram (Rahmawati, 2022)

Tabel 2. 1 Perbandingan Penelitian

No	Nama	Judul Penelitian	Metode	Hasil
1.	Ronaldo Panjaitan & Leonardi Paris	Sistem Informasi Gereja HKBP Tanjung Balai Karimun		Hasil dari penelitian ini berupa sistem informasi yang dapat membantu pihak gereja

	Hasugian, 2018		<i>Waterfall</i>	dalam proses mengelola tata acara, renungan, jadwal kegiatan gereja, informasi umum, data jemaat serta membantu jemaat untuk mengakses informasi mengenai gereja.
2.	Bayu Pratama Nugroho & Sherly Jayanti, 2017	Implementasi Sistem Informasi Berbasis Web (Studi Kasus Gereja GKE Sion Palangkaraya)	<i>Waterfall</i>	Sistem yang dibuat dapat membantu memperkenalkan gereja, memotivasi pelayanan, kemudahan pengurusan pada GKE Sion Palangkaraya, dan jemaat dapat mendownload file yang ingin dibutuhkan.
3.	Gorby Pitoi, Hans Wowor, & Yaulie Rindengan, 2017	Perancangan Sistem Informasi Jemaat dan pekerja Gereja Masehi Injili di Minahasa	DAD (<i>Disciplined Agile Delivery</i>)	Sistem ini dapat memudahkan pengguna dalam mengakses data sensus, dan mencari informasi yang di butuhkan.
4.	Dame Sagala, Ali Sadikin, & Beni	Perancangan Sistem Pengolahan Data Jemaat Berbasis	<i>Waterfall</i>	Sistem yang dibuat dapat mengelola data jemaat, data keuangan, dan memudahkan

	Irawan, 2018	Web Pada Gereja GKPI Kota Jambi		pengguna untuk mendapatkan informasi mengenai gereja secara online.
5.	Alfred Yulius Arthadi Putra, Amok Darmianto, & Christiance Liunome, 2016	Perancangan <i>Website</i> Gereja Misi Injili Indonesia (GMII) Jemaat Sola Gratia	<i>Waterfall</i>	Dengan adanya sistem informasi, kinerja dalam kegiatan penyampaian informasi pada Gereja Misi Injili Indonesia (GMII) Jemaat Sola Gratia Siantan Hulu Pontianak utara dapat lebih optimal.
6.	Diah Rahmawati, Muhamad Raihan Gufran, Nia Komalasari	Perancangan Sistem Informasi Pembukuan UKM Konveksi Bim Collection Berbasis Website	<i>OOAD</i> (<i>Object</i> <i>Oriented</i> <i>Analysis</i> <i>and</i> <i>Design</i>)	Dengan adanya sistem ini dapat digunakan untuk melakukan pembukuan pada Usaha Kecil Menengah(UKM) di kota Tangerang yang dipermudah dengan menggunakan metode OOAD

Referensi yang digunakan dalam penelitian ini berasal dari penelitian yang dilakukan oleh Alfred Arthadi Putra, Amok Darmianto, & Christiance Liunome, 2016 dengan judul “Perancangan *Website* Gereja Misi Injili Indonesia (GMII) Jemaat Sola Gratia” yang menggunakan metode *Waterfall*.

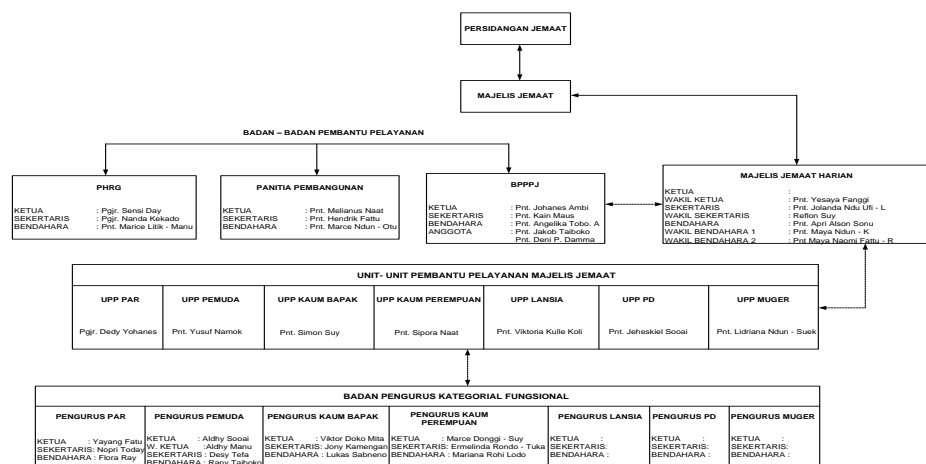
Sistem yang dibuat memiliki kelebihan dalam pengelolaan data acara, data keuangan gereja, jadwal kegiatan gereja, informasi umum, dan pendataan jemaat. Kekurangan dari penelitian yang dibuat belum memiliki informasi mengenai pendataan jemaat, struktur organisasi majelis, laporan data jemaat, dokumentasi kegiatan seperti foto, dan penelitian yang dilakukan menggunakan metode waterfall yang akan dilakukan pengembangan menggunakan metode lain yaitu metode OOAD (*Object Oriented Analysis dan Design*). Dari kekurangan penelitian sebelumnya, maka penelitian kali ini didapatkan informasi tambahan untuk merancang sistem informasi GMT Imanuel Beumopu yang lebih baik.

2.2 Tinjauan Obyek Penelitian

a. Sejarah GMT Imanuel Beumopu

GMT Imanuel Beumopu didirikan pada tanggal 30 November 1979 oleh pendeta P. F. De Haan. GMT Imanuel Beumopu beralamat di Jl. Yohanis Fanggi, Lasiana, Kota Kupang.

b. Struktur Organisasi GMT Imanuel Beumopu



Gambar 2. 1 Struktur Organisasi

2.3 Konsep Dasar Sistem Informasi

Pada bagian ini memberikan penjelasan dasar mengenai sistem informasi secara umum yaitu penjelasan data, sistem, informasi, sistem informasi, *website*, *web browser*, dan *web server*.

a. Data

Data adalah kejadian nyata berupa fakta-fakta yang dialami setelah itu direkam dan disimpan pada komputer (Yulius, Putra and Darmianto, 2016). Data merupakan representasi dari suatu kejadian nyata yang diubah menjadi bentuk tertentu agar dapat rekaman, ingatan dan dapat diolah menjadi informasi yang berguna (Dame Christine Sagala, Ali Sadikin, 2018).

b. Sistem

Sistem merupakan komponen yang melakukan suatu modelling dan fungsionalitas untuk saling berkomunikasi didalam sistem berfungsi merubah inputan yang didapatkan oleh sistem menjadi manfaat bagi pengguna (Yulius, Putra and Darmianto, 2016).

Sistem merupakan suatu metode jaringan yang dibuat berbentuk sistematis untuk melakukan kegiatan utama perusahaan (Muhdar Abdurahman, Mudar Safi, 2019)

c. Informasi

Informasi merupakan hasil pengolahan data berupa kejadian-kejadian nyata yang digambarkan menjadi suatu bentuk yang bermanfaat bagi penerima untuk mengambil keputusan (Yulius, Putra and Darmianto,

2016). Informasi adalah data yang dikumpulkan menjadi informasi yang digunakan untuk menyelesaikan permasalahan tertentu dan memperoleh sejumlah data kembali untuk dijadikan informasi. Proses ini merupakan siklus informasi yang akan terus berulang (Karim, 2020).

d. Sistem Informasi

Sistem Informasi yang berarti elemen-elemen yang berinteraksi untuk menghimpun data dari kejadian nyata guna mendukung pengambilan keputusan saat ini atau mendatang (Yulius, Putra and Darmianto, 2016). Sistem informasi adalah gabungan dari *brainware*, *hardware*, *software*, dan jaringan komunikasi yang mengumpulkan, mengubah, dan memberikan informasi pada suatu organisasi (Muhdar Abdurahman, Mudar Safi, 2019)

e. Website

Website yang berarti kumpulan halaman *web* yang berada dalam sebuah domain untuk memberikan informasi (Prayitno and Safitri, 2015). *Website/web* adalah kumpulan halaman yang berelasi bisa mengandung teks, gambar, suara dan video yang mempunyai kelebihan untuk mengkoneksi link dengan sebuah dokumen yang bisa diakses melalui sebuah *browser* (Dame Christine Sagala, Ali Sadikin, 2018)

f. Web Browser

Web browser yang berarti software yang mendukung pengguna untuk berinteraksi dengan kumpulan data berupa teks, gambar, video, game ataupun informasi lainnya yang berada pada sebuah *website* dengan

menggunakan jaringan internet (Sopandi, Abdussomad and Nopostasari, 2018).

g. *Web Server*

Web server adalah suatu perangkat yang terdiri dari perangkat keras dan perangkat lunak yang mempersiapkan layanan akses bagi pengguna melalui aturan komunikasi HTTP ataupun sejenisnya seperti FTP dan HTTPS terhadap data-data yang ada pada suatu URL ke pengguna (Prayitno and Safitri, 2015).

2.4 Bahasa Pemrograman

Dalam membangun sebuah *website* diperlukan beberapa bahasa pemrograman dan bahasa *scripting* seperti:

a. *Hypertext Processor* (PHP)

Bahasa program berbentuk skrip yang berada di *server* sehingga PHP sering disebut sebagai *Server Side Scripting* bahwa dalam penggunaan PHP dibutuhkan *web server* dan *web browser* untuk melihat hasilnya. PHP dibuat untuk membentuk suatu tampilan berdasarkan keinginan pembuat. Dalam membentuk suatu tampilan PHP membutuhkan kode HTML sebagai pondasi awal dari kerangka layout *web*. Selanjutnya PHP bertugas untuk memproses data dari sisi *server*, sehingga terbentuk suatu tampilan yang dinamis. *Hypertext Preprocessor* (PHP) dikembangkan pertama kali oleh Rasmus Lerdorf yang merupakan seorang pengembang perangkat lunak yang dirilis pada tahun 1995. Setelah itu, PHP resmi diterima

sebagai sebuah bahasa pemrograman baru dan banyak peminat di seluruh dunia (Purbadian, 2015)

b. *Hypertext Markup Language (HTML)*

HTML bahasa program yang dibangun untuk membantuk pembuat *website*. Kemudian *website* yang merupakan kumpulan halama dapat dilihat oleh pengguna yang tersambung ke internet. Hal tersebut terus berkembang dari masa ke masa yang mempunyai tuntutan dan persyaratan dibawah arahan W3C. HTML terdiri dari kode-kode yang diketik dalam bentuk teks-file, teks tersebut disimpan dengan *type* html dan dapat dilihat melalui *browser* dan menerjemahkan teks tersebut (Pitoy, Hans and Kom, 2016).

c. *JavaScript*

JavaScript adalah bahasa *script* yang didasari oleh objek untuk pemakai lebih leluasa mengontrol banyak aspek pada suatu dokumen HTML. Objek yang digunakan berupa suatu dokumen, URL, windows, frame, form, item lain yang semuanya itu mempunyai properti yang saling berkaitan, masing-masing memilik nama, lokasi, warna nilai, dan atribut lain (Taryana, 2020)

2.5 Aplikasi Pendukung

Berikut aplikasi yang digunakan untuk mendukung pembuatan *website* diantaranya:

a. XAMPP

XAMPP adalah aplikasi yang memungkinkan sebuah *web* dinamis bisa diakses secara local menggunakan *web server* local yang dapat berjalan di semua sistem operasi. Kata XAMPP mempunyai arti X adalah *Cross Platform* dimana xampp dapat berjalan di semua sistem operasi, A adalah *apache* sebagai *web server*nya, M adalah MySQL sebagai *database Management System* (DBMS), dan PP adalah PHP dan Perl sebagai bahasa yang didukungnya (Huda, 2019).

b. *Visual Studio Code* (VSCode)

Visual Studio Code merupakan aplikasi *editor source* yang dibuat oleh *Microsoft* pengguna sistem operasi Windows, MacOS, dan Linux. Aplikasi merupakan dukungan untuk debugging, GIT (*Group Inclusive Tour*) *Controll* yang disematkan, penyelesaian kode cerdas, kode refactoring, penyorotan sintaks, dan lainnya. Aplikasi *Visual Studio Code* merupakan aplikasi gratis dan *open-source*. *Visual studio code* digunakan untuk menyebarkan *Node.js* untuk tampilan yang berjalan pada *blinklayout*. *Visual studio code* tidak menggunakan *atom* melainkan menggunakan komponen editor yang diberi kode dengan nama *monaco* yang sebelumnya disebut *visual studio online* (Joni, 2019).

c. *CodeIgniter* (CI)

CodeIgniter merupakan sebuah framework untuk membuat sebuah *web* dengan menggunakan bahasa *Hypertext Preprocessor* (PHP). Pada *CodeIgniter* terdapat kelas yang berbentuk library dan helper berfungsi

untuk membantu proses pembangunan suatu program aplikasi *web*. *CodeIgniter* menggunakan aturan *Model View Controller* (MVC) untuk menata dan memudahkan dalam membangun suatu aplikasi. *View* merupakan bagian yang menangani *presentation logic*, *Model* merupakan teknik memanipulasi data berupa *Insert, update, delete, search*, dan menangani validasi dari bagian *Controller*. *Controller* merupakan bagian untuk menerima permintaan data dari pengguna dan memproses data tersebut (Fadila and Oktivasari, 2015).

2.6 Basis Data

Database/basis data merupakan kumpulan data yang saling terkait secara rinci, *database* dapat dianggap sebagai suatu penyusunan data yang teratur dan disimpan dalam media penyimpanan permanen yaitu harddisk dengan tujuan agar dapat tidak mudah hilang dan dapat diakses dengan mudah dan cepat. *Database* terbagi atas beberapa macam yaitu data hierarkis, *database* jaringan, dan *database* relasional. *Database* relasional merupakan *database* yang paling terkenal dan telah diterapkan pada berbagai platform dari PC hingga minikomputer. Sebuah *database* relational tersusun atas sejumlah tabel sebagai contoh: *database* perpustakaan mencakup tabel-tabel seperti tabel pengunjung, tabel admin, tabel buku dan lain sebagainya (Kadir, 2008).

2.7 Konsep Dasar Entity Relationship Diagram (ERD)

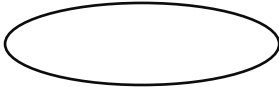
Entity Relationship Diagram (ERD) adalah suatu sistem yang menampilkan relasi antar entitas yang saling terhubung. ERD menyediakan




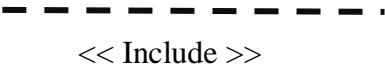
pola bentuk terstruktur dari data sistem. Dalam ERD data ditampilkan dalam bentuk simbol *entity* yang saling berelasi untuk menyediakan data yang dibutuhkan oleh sistem (Yuliawan and Sunarto, 2013).

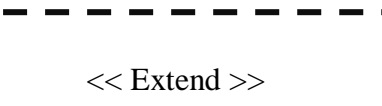
2.8 Konsep Dasar *Use Case*

Use Case Diagram menggambarkan hubungan antara aktor dan use case melalui garis-garis yang menghubungkan mereka. Gambaran tersebut membantu para pengembang perangkat lunak dan pemilik kepentingan lainnya untuk memahami fungsionalitas utama sistem, bagaimana pengguna akan berinteraksi dengan sistem, serta skenario-skenario yang mungkin terjadi dalam penggunaan nyata (Hendini, 2018). Simbol-simbol umum dalam *use case* yaitu:

Tabel 2. 2 Simbol-simbol *Use Case*

Simbol	Keterangan
	<i>Use case</i> merupakan fitur sistem yang digunakan aktor untuk mewakili aktivitas/tindakan dalam sistem.

 <p style="text-align: center;">Actor</p>	<p>Aktor (<i>Actor</i>) merupakan gambaran dari pengguna (manusia), sistem eksternal, atau komponen lain yang mempunyai peran berinteraksi dengan sistem atau proses dalam <i>use case</i>. Perlu diketahui bahwa Aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i>.</p>
	<p>Interaksi yang dilakukan antara <i>actor</i> dan <i>use case</i> secara langsung tetapi tidak mengindikasikan data digambarkan dalam bentuk garis tanpa tanda panah.</p>
	<p>Interaksi yang dilakukan antara <i>actor</i> dan <i>use case</i> secara pasif dengan sistem digambarkan dalam bentuk garis panah terbuka</p>
	<p><i>Include</i> merupakan gambaran suatu <i>use case</i> mengandung langkah-langkah yang diperlukan dari <i>use</i></p>





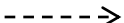
	case lain. Contohnya pemanggilan sebuah fungsi program.
	Extend merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi

2.9 Konsep Dasar *Class Diagram*

Class Diagram menggambarkan struktur statis dari sistem perangkat lunak. Diagram ini menunjukkan hubungan antara berbagai kelas (objek atau entitas) dalam sistem, serta atribut dan metode yang dimiliki oleh kelas-kelas tersebut. Diagram kelas membantu dalam memodelkan struktur data dan perilaku kelas dalam sistem, yang pada gilirannya membantu dalam pemahaman dan perancangan arsitektur perangkat (Hendini, 2018). Simbol-simbol umum dalam *class diagram* yaitu

Tabel 2. 3 Simbol-simbol Class Diagram




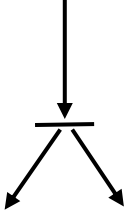
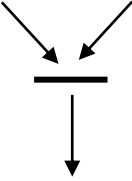
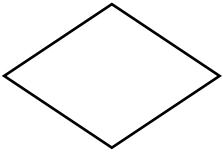
Simbol	Nama	Keterangan
Nama Kelas +Atribut +Operasi	Kelas	Kelas atau sebuah kapsul yang menampung nama kelas, atribut dan operasi didalamnya.


	Antar Muka/interfase	Interfase dalam pemrograman berorientasi objek.
	Asosiasi/Asosiation	Relasi antar kelas dengan makna secara umum.
	Asosiasi Berarah/Directed Asosiation	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lainnya.
	Generalisasi	Relasi antar kelas secara spesial dari umum ke khusus.
	Kebergantungan/dependency	Relasi antar kelas dengan makna ketergantungan antar kelas.

2.10 Konsep Dasar Activity Diagram

Diagram aktivitas (*activity diagram*) menggambarkan arah kerja atau aktivitas dalam suatu proses atau skenario. Diagram ini membantu dalam memvisualisasikan bagaimana tugas, aktivitas, dan keputusan berinteraksi dalam proses fungsionalitas sistem. (Hendini, 2018). Simbol-simbol yang umum dalam *activity diagram* yaitu:

Tabel 2. 4 Simbol-simbol Activity Diagram



Gambar	Keterangan
	<p><i>Start Point</i> menandakan titik awal dari aliran aktivitas.</p>
	<p><i>End Point</i> menandakan akhir dari suatu aktivitas.</p>
	<p><i>Activities</i> menandakan suatu proses aktivitas atau tugas yang harus dilakukan dalam proses.</p>
	<p><i>Fork Note</i> menandakan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.</p>
	<p><i>Join</i> (penggabungan) menandakan penggabungan antara alur-alur paralel menjadi satu alur tunggal.</p>
	<p><i>Decision Node</i> menandakan pengambilan keputusan dan mengarahkan aliran ke berbagai arah sesuai kondisi yang diberikan.</p>



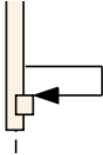


	<p><i>Swimlane</i> menandakan pembagian aktivitas untuk menunjukkan siapa melakukan apa.</p>
---	--

2.11 Konsep Dasar Sequence Diagram

Diagram urutan (sequence diagram) menggambarkan interaksi antara berbagai objek dalam sistem melalui urutan pesan atau panggilan yang dikirimkan antara objek-objek tersebut. Tujuan utama dari diagram urutan adalah untuk menggambarkan aliran peristiwa atau urutan komunikasi yang terjadi dalam suatu skenario atau proses dalam sistem (Hendini, 2018). Simbol-simbol yang umum dalam *sequence diagram* yaitu:

Tabel 2. 5 Simbol-simbol Sequence Diagram

Simbol	Keterangan
	<p><i>Entity Class</i> merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.</p>
	<p><i>Boundary Class</i> merupakan kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih Aktor dengan sistem, seperti tampilan form entry dan form cetak.</p>

	<p><i>Control class</i> merupakan suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat activation.</p>

2.12 Konsep Dasar Gereja

Gereja berasal dari bahasa Portugis yaitu igreja, sedangkan dari bahasa Yunani yaitu ekklesia yang berarti panggilan keluar (ek=keluar; klesia dari kata kaleo=memanggil) jadi artinya perkumpulan orang yang dipanggil keluar dari kegelapan dunia. Gereja merupakan badan yang diatur berdasarkan konsep kepercayaan kepada Allah. Kumpulan gereja di

Indonesia hidup dalam keanekaragaman yang dapat dilihat dari berbagai sudut yaitu etnis, corak kekristenan, pengakuan iman, pemberitaan injil, dan organisasi (Dame Christine Sagala, Ali Sadikin, 2018).

2.13 Konsep Dasar Gereja Masehi Injili di Timor (GMIT)

G.P.H Locker menyampaikan keinginannya untuk membangun gereja Timor yang mandiri sejak tahun 1933. Pada jaman itu gereja Maluku dan gereja Minahasa dinilai siap oleh Pengurus Gereja namun penilaian berbeda kepada gereja Timor karena dinilai belum siap mandiri. Pada pertengahan tahun 1946 gereja Timor menuntut untuk mandiri. Setelah melewati beberapa proses panjang, pada akhirnya diputuskan bahwa Gereja Masehi Injili di Timor siap untuk mandiri pada 31 Oktober 1947. Gereja ini terletak di Prov. Nusa Tenggara Timur (NTT) (Wetangterah, 2011).