

APLIKASI PENCARIAN RUTE MENUJU KAHYANGAN JAGAD DI BALI MENGGUNAKAN DIJKSTRA MELALUI MOBILE GMAPS

Ni Luh Made Susi Kartika Dewi¹⁾, I Made Darma Susila, S.Kom., M.Kom²⁾, Ida Bagus Suradarma, SE., MSi³⁾

Program Studi Sistem Informasi STMIK STIKOM Bali

Jl.Raya Puputan No.86 Renon, Denpasar Telp. (0361)244445 Fax 0361)264773

Website : <http://www.stikom-bali.ac.id> | Email : info@stikom-bali.ac.id

email: ¹⁾ Susikartikadewi69@gmail.com, ²⁾ darma_s@stikom-bali.ac.id, ³⁾ suradarma@stikom-bali.ac.id

ABSTRAK

Pencarian rute terpendek dalam perjalanan menuju kahyangan jagad di bali merupakan masalah sehari – hari, khususnya masyarakat bali dan wisatawan dalam memberikan informasi mengenai rute terpendek pada saat mencari rute terdekat. Terkadang banyak rute yang tersedia dalam mencapai suatu lokasi tujuan. Untuk menentukan pencarian rute awal menuju rute tujuan, maka di bangunlah Aplikasi Pencarian Rute Menuju Kahyangan Jagad Di Bali Menggunakan Dijkstra Melalui Mobile Gmaps. Aplikasi ini akan mempermudah Masyarakat mendapat informasi rute terpendek yang akan dilalui, di samping menghemat waktu dan biaya. Pada penelitian ini menggunakan metode dijkstra dalam pencarian lokasi Pura yang dicari berdasarkan kecamatan dan kelurahan yang ada di seluruh bali. Ditampilkan menggunakan *google map* pada Sistem *web*, sistem ini dapat memudahkan dalam menemukan lokasi, keterangan, dan hasil rute terdekat kahyangan jagad di Bali.

Kata kunci : Algoritma Dijkstra, Pencarian Rute, Kahyangan Jagad, Pemetaan, *Google maps*

ABSTRACT

Search the shortest route on the way to kahyangan Jagad Bali is a daily problem - the day, especially communities Balinese and tourists in providing information about the shortest route when looking for the nearest route. How to determine the initial route search toward route purpose. the Application Search Route to Kahyangan Jagad In Bali Using Dijkstra Through Mobile Gmaps. This application will facilitate the public to get the shortest route information will be passed to save time and costs. In this study is to use the method dijkstra searched based on existing sub districts and village in bali and are shown using google map on web systems, this system can facilitate in finding the location, description, and the results of the nearest kahyangan jagad in Bali.

Keywords: Dijkstra's algorithm, Search Routes, Universal heaven, Mapping, Google maps

1. Pendahuluan

Dengan maraknya perkembangan *persasive computing* belakangan ini membuat banyaknya aplikasi – aplikasi yang di kembangkan untuk mendukungnya. Sistem Informasi Geografis, sistem berbasis spasial, dan digital mapping juga berkembang di tahun – tahun belakangan ini. Terlebih dengan banyaknya platform terbuka yang menyediakan datanya secara terbuka di WWW (*World wide web*) membuat banyak sekali domain masalah baru yang belum terjamah, dan tentunya platform yang terbuka berpengaruh langsung dalam memakan waktu pengembangan *software* dan *observasi* masalah dengan fenomena yang menguntungkan, baik untuk diri sendiri, pendidikan formal maupun pengembangan perangkat lunak secara professional.

Bali terkenal dengan keindahan alam, objek wisata di Bali dan juga memiliki budaya yang religius, sehingga Bali yang dengan mayoritas agama Hindu, dikenal juga dengan Pulau Seribu Pura. Dengan adanya katagori tersebut maka penyampaian informasi sangat terbatas. Hal ini menjadi salah satu kendala dalam penyampaian informasi di bidang rute menuju pura kahyangan jagad. khususnya

masyarakat bali dan wisatawan yang belum mengetahui jalan terpendek yang akan di lalui untuk menghemat biaya dan waktu. maka masyarakat kebingungan untuk menentukan rute mana yang lebih dekat untuk dilalui. Terkadang, banyak rute yang disediakan dalam mencapai suatu lokasi tujuan. Maka untuk mengatasi masalah ini yakni dengan menggunakan algoritma djikstra sebagai salah satu metode pencarian rute pura kayangan jagad untuk sampai ke tempat tujuan. Pada penelitian ini yaitu menggunakan Algoritma Dijkstra. Algoritma Dijkstra merupakan suatu algoritma yang digunakan secara luas pada metode pencarian rute, yakni sebuah proses penentuan rute yang efektif antar titik yang disebut *node*. Fasilitas-fasilitas yang sudah tersedia yaitu *google maps* yang akan dimanfaatkan untuk mengambil data dengan manual dengan pemodelan data dalam bentuk matematis dan di olah dengan algoritma Dijkstra. untuk pencarian lokasi Pura dapat dicari berdasarkan kecamatan dan kelurahan yang ada di seluruh bali dan akan ditampilkan menggunakan *google map* pada Sistem *web*, Sistem ini dapat memudahkan dalam menemukan lokasi, keterangan, dan hasil rute terdekat kahyangan jagad di Bali.

2. Tinjauan Pustaka

Dalam perancangan sistem ini, digunakan beberapa tinjauan pustaka yang digunakan sebagai acuan dan dasar teori.

2.1 Pengertian Pura

Mengenai istilah Pura yang dipergunakan sebagai tempat pemujaan umat Hindu diperkirakan pada jaman Dalem berkuasa di **Bali**. Sebelum dikenal istilah Pura, untuk menunjukkan tempat pemujaan Hindu di Bali dikenal istilah Kahyangan atau Hyang. bahkan pada jaman Bali kuno dipakai istilah “Ulon” yang berarti tempat suci atau tempat yang dipakai untuk berhubungan dengan ke-Tuhanan.^[1]

2.1.1 Pengertian Pura Kahyangan Jagad

Pura Kahyangan Jagad, yaitu Pura umum tempat pemujaan Hyang Widhi dengan segala prabhawanya serta roh suci leluhur, termasuk didalamnya Pura Sad Kahyangan dan Dang Kahyangan. Pura Kahyangan Jagad ialah Pura-pura Kahyangan Agung terutama yang terdapat di delapan penjuru mata angin dan pusat pulau Bali.

2.2 Definisi Algoritma Dijkstra

Algoritma Dijkstra merupakan suatu algoritma disebut juga sebagai single - source shortest path yang digunakan dalam menentukan jalur terpendek dari simpul sumber menuju simpul tujuan berdasarkan bobot pada sisi. Bobot pada sisi dapat berupa jarak, waktu, biaya, ataupun bobot yang lainnya. Algoritma Dijkstra bekerja dengan cara mengunjungi semua semua simpul-simpul yang terdapat pada graf dengan dimulai pada simpul sumber.^[2]

2.3 Google Maps

Google Maps merupakan sebuah layanan peta dunia virtual berbasis web yang disediakan oleh Google.^[3]

1. ASP.NET

ASP.NET adalah teknologi baru dalam pemrograman *web* yang merupakan kelanjutan dari teknologi ASP 3.0 Perbedaan utama dibanding *ASP klasik* adalah penggunaan.^[4]

2. SQL SERVER 2008

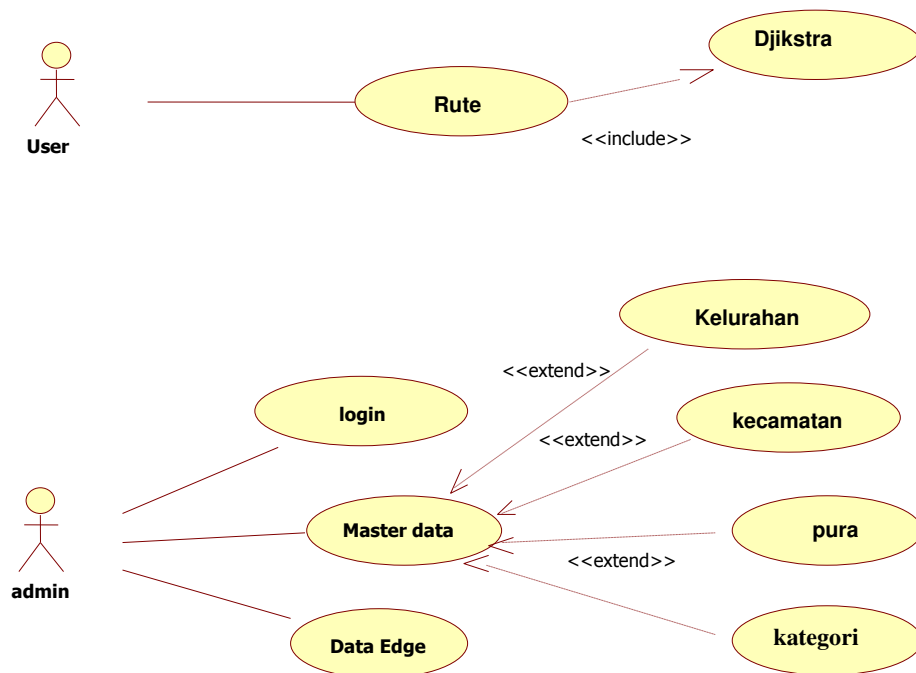
SQL(Structured Query Language) adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relasional. Standar *SQL* mula-mula didefinisikan oleh *ISO (International Standards Organization)* dan *ANSI (the American National Standards Institute)* yang dikenal dengan sebutan *SQL 86*.^[5]

3. Analisa Dan Perancangan Sistem

3.1 Analisa Sistem

Analisa sistem merupakan eknik pemecahan permasalahan yang menguraikan bagian-bagian komponen dengan khusus seberapa baik bagian-bagian komponen tersebut bekerja dan nerinteraksi untuk mencapai sasaran serta tujuan yang telah ditetapkan.

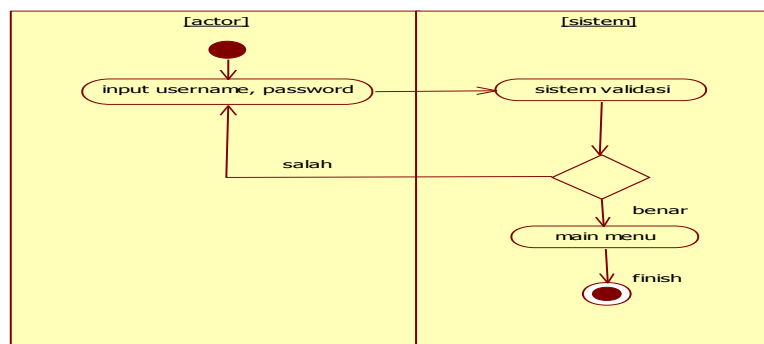
a. Use case Diagram untuk User dan Admin



Gambar 1 Use Case Diagram

Use Case Diagram digunakan untuk menganalisa suatu sistem yang terstruktur yang digunakan di dalam sistem ini. Pada gambar 1 use case diagram Aplikasi Pencarian Rute Terpendek memiliki 2 aktor utama, yakni aktor *Admin* dan aktor *User*. Aktor *user* memiliki 2 activity yakni Activity Diagram rute terdiri dari Activity Diagram *dijkstra* yang memerlukan data dari hasil rute dengan *Dijkstra*. Sedangkan *Admin* memiliki 3 activity yakni activity diagram *login*, activity Diagram *edge* dan activity diagram *master data* yang terdiri atas data Kelurahan, data kecamatan, kategori dan data pura. Activity diagram *edge* yang terbentuk dari hubungan tiap node. Activity diagram *master data* fungsinya untuk *insert*, *update* dan *delete* pada sistem, Untuk melihat lokasi dan tujuan *user* dan *admin*, sehingga rute dapat ditampilkan pencarian jalur terpendek antar *node* yang berhubungan tersebut.

b. Activity Diagram Login

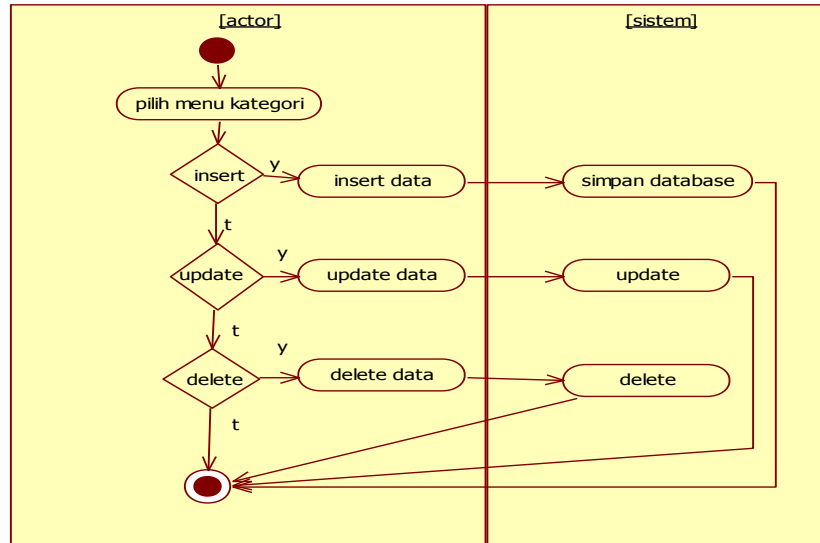


Gambar 2 Activity Diagram Login

Pada Gambar 2 activity diagram login ini, *admin* bisa melakukan *login* ke dalam system akan melakukan return page dengan menginputkan *username* dan *password* yang nantinya sistem melakukan

proses *login* data lalu melakukan *validasi*. Apabila benar, maka tampilan halaman *home*, jika tidak, kembali ke proses memasukkan *username* dan *password*.

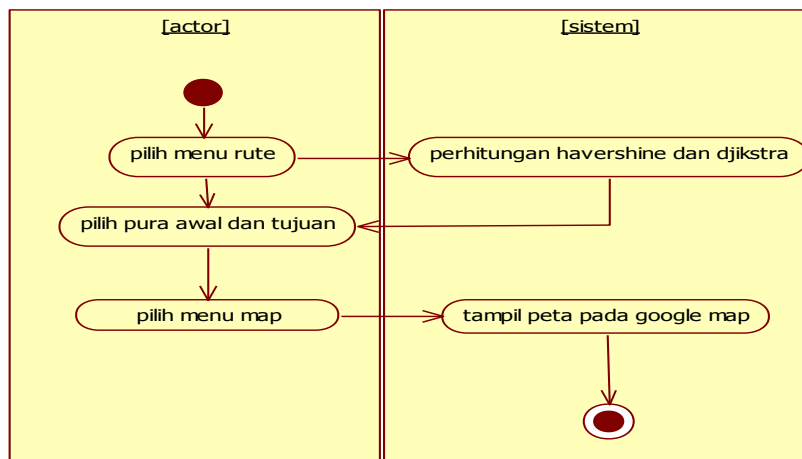
c. Activity Diagram Kategori



Gambar 3 Activity Diagram Kategori

Pada gambar 3 *activity diagram kategori* ini *admin* pilih menu kategori pada *insert* data dapat validasi jika ya dapat *insert* data dan di sistem dapat menyimpan ke database, jika tidak admin dapat melakukan *update* data, jika ya sistem akan melakukan *update*, jika tidak melakukan *update* maka admin dapat melakukan *delete* data jika ya sistem akan melakukan *delete* jika tidak akan melakukan proses *finish* setelah tersimpan *insert* data dan melakukan *update* data pada sistem maka dapat melakukan proses *finish*.

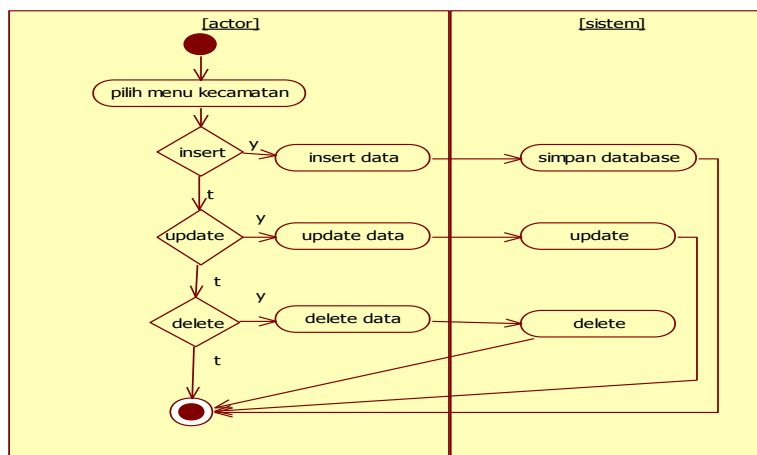
d. Activity Diagram Route



Gambar 4 Activity Diagram Route

Pada Gambar 4 *activity Diagram Route* ini user memilih menu rute, lalu memilih awal dan tujuan menuju pura sistem akan memperhitungkan *dijkstra* dan *havershine* di setiap nodenya, user dapat memilih map sistem akan menampilkan rute terpendek pada google map.

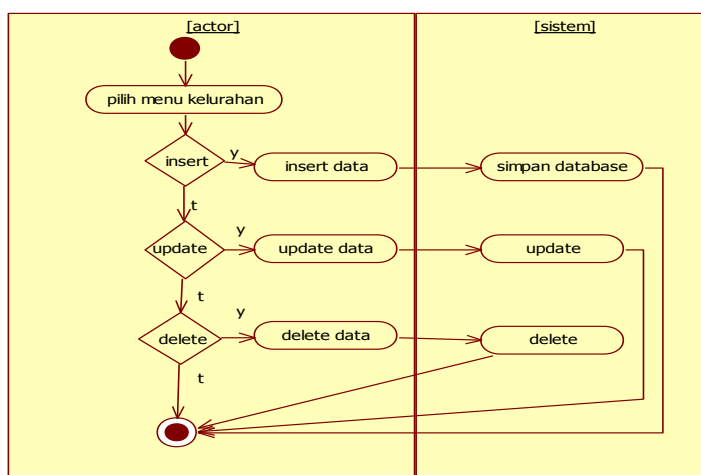
e. Activity Diagram Kecamatan



Gambar 5 Activity Diagram Kecamatan

Pada gambar 5 Activity Diagram Kecamatan ini admin pilih menu kecamatan pada *insert* data dapat validasi jika ya dapat *insert* data dan di sistem dapat menyimpan ke database, jika tidak admin dapat melakukan *update* data, jika ya sistem akan melakukan *update*, jika tidak melakukan *update* maka admin dapat melakukan *delete* data jika ya sistem akan melakukan *delete* jika tidak akan melakukan proses *finish* setelah tersimpan *insert* data dan melakukan *update* data pada sistem maka dapat melakukan proses *finish*.

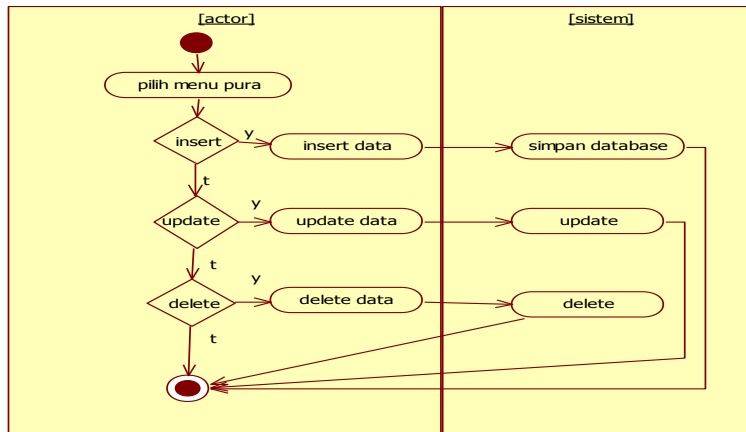
f. Activity Diagram Kelurahan



Gambar 6 Activity Diagram Kelurahan

Pada gambar 6 Activity Diagram Kelurahan ini admin pilih menu kelurahan pada *insert* data dapat validasi jika ya dapat *insert* data dan di sistem dapat menyimpan ke database, jika tidak admin dapat melakukan *update* data, jika ya sistem akan melakukan *update*, jika tidak melakukan *update* maka admin dapat melakukan *delete* data jika ya sistem akan melakukan *delete* jika tidak akan melakukan proses *finish* setelah tersimpan *insert* data dan melakukan *update* data pada sistem maka dapat melakukan proses *finish*.

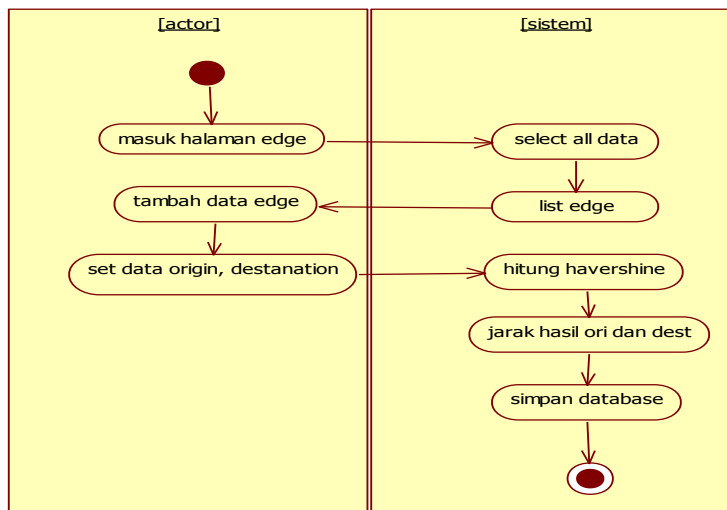
g. Activity Diagram Pura



Gambar 7 Activity Diagram Pura

Pada Gambar 7 Activity Diagram Pura ini admin pilih menu Pura pada *insert* data dapat validasi jika ya dapat *insert* data dan di sistem dapat menyimpan ke database, jika tidak admin dapat melakukan *update* data, jika ya sistem akan melakukan *update*, jika tidak melakukan *update* maka admin dapat melakukan *delete* data jika ya sistem akan melakukan *delete* jika tidak akan melakukan proses *finish* setelah tersimpan *insert* data dan melakukan *update* data pada sistem maka dapat melakukan proses *finish*.

h. Activity Diagram Edge

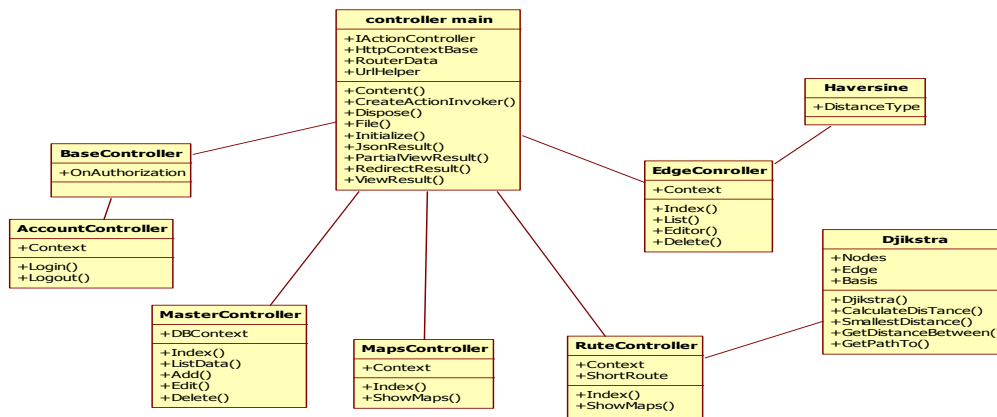


Gambar 8 Activity Diagram Edge

Pada Gambar 8 Activity Diagram Edge ini admin pilih halaman *edge* sistem akan melakukan *select all* data di sistem melakukan proses *list edge* admin akan melakukan *tambah data edge* dan *set* data

origin, destination dimana sistem akan melakukan hitungan *havershine* dan mendapatkan hasil jarak *origin* dan *destination* setelah itu di simpan di database.

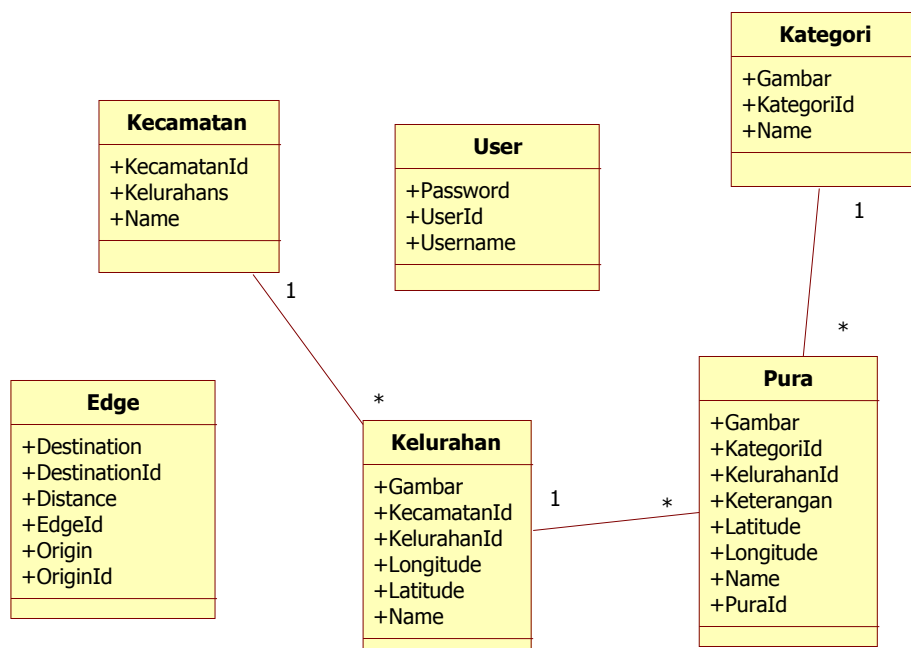
i. Class Diagram



Gambar 9 Class Diagram

Pada gambar 9 dan 10 Class Diagram ini merupakan basis logika dalam program terdapat pada controller dan class diagram data. Controller main merupakan class bawaan dari sistem. Dari main controller tersebut diturunkan menjadi beberapa controller lain sesuai dengan proses yang diperlukan dalam aplikasi. Master Controller dijadikan satu untuk master data seperti kelurahan, kecamatan, user, Pura, Kategori. Sedangkan Edge memerlukan kelas lain yaitu Haversine dalam perhitungan jarak. Kemudian Route Controller di dalamnya terdapat kelas djikstra untuk menentukan rute terpendek.

j. Class Diagram Data



Gambar 10 Class Diagram Data

4. Desain Antarmuka Sistem

Pada Desain antarmuka sistem penentuan penerima bantuan ini dii gambarkan dan dijelaskan mengenai desain yang digunakan beserta tampilan yang ada didalamnya

a. Form Halaman Utama pada login

Pada Gambar 4.1 Form Halaman *utama* adalah halaman yang pertama kali muncul apabila seorang user ingin mengakses sistem tersebut. Sistem ini akan menampilkan beberapa menu di antaranya home, lokasi, peta, rute dan admin.



Gambar 4.1 Form Halaman Utama

b. Form Halaman Lokasi

Pada gambar 4.2 form Halaman lokasi adalah halaman yang menampilkan list lokasi pada pura secara detail dan mempunyai beberapa menu – menu di antaranya : gambar, nama pura, kategori dan dapat memberikan informasi secara detail. Jika detail di klik akan menampilkan informasi yaitu : kategori, nama, keterangan, latitude, longitude, kecamatan dan kelurahan.



Gambar 4.2 Form Halaman Lokasi

c. Form Halaman Peta

Pada gambar 4.3 form halaman peta adalah menampilkan posisi pura yang telah di tentukan dengan sisi *server*.



Gambar 4.3 Form Halaman Peta

d. Form Halaman Route

Pada gambar 4.4 form halaman rute adalah halaman form ini proses pencarian dan perhitungan rute terpendek dengan menggunakan algoritma djikstra dapat mulai di implementasikan. Setiap lokasi pura yang berada pada kelurahan dan kecamatan tertentu akan menjadi sebuah *node* dalam perhitungan Algoritma Dijkstra, sedangkan jarak antar masing - masing *node* yang berhubungan menjadi *edge*. *Edge* di dapat dengan menggunakan perhitungan *haversine*. Setelah user menentukan start awal dan tujuan maka sistem akan memulai perhitungan dari titik awal ke *node* berdekatan. Semua *node* akan dihitung dan dicari jalur terpendek antar *node* yang berhubungan tersebut dan pada gambar 4.5 halaman peta rute terpendek adalah halaman menampilkan halaman peta rute terpendek

Nama	Jarak	Rute
Pura Luthar Marabara - Pura Kasek/Kag. Gede	10.000	Pura Luthar Marabara - Pura Kasek/Kag. Gede
Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede	0.000	Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede
Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede	0.000	Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede
Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede	0.000	Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede
Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede	0.000	Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede
Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede	0.000	Pura Kasek/Kag. Gede - Pura Kasek/Kag. Gede

Total Jarak yang ditempuh : 30,365 km

Gambar 4.4 Form Halaman Route



Gambar 4.5 Halaman Peta Rute Terpendek

e. Form Halaman Login Admin

Pada gambar 4.6 form halaman *login Admin* akan menginputkan *username* dan *password* dan menampilkan halaman admin di antaranya *user*, *edge*, kelurahan, kecamatan, pura, kategori, *log out*.

Gambar 4.6 Form Halaman Login Admin

f. Form Halaman Kategori

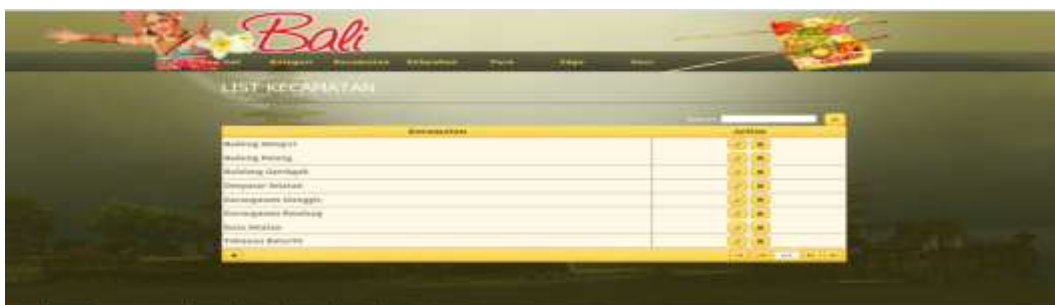
Pada gambar 4.7 form Halaman kategori pada admin adalah sebagai pengolahan data pada kategori untuk pengolahan *edit*, *delete* yang akan di lakukan oleh admin dan akan menampilkan diantaranya gambar dan kelurahan.



Gambar 4.7 Form Halaman Kategori

g. Form Halaman Kecamatan

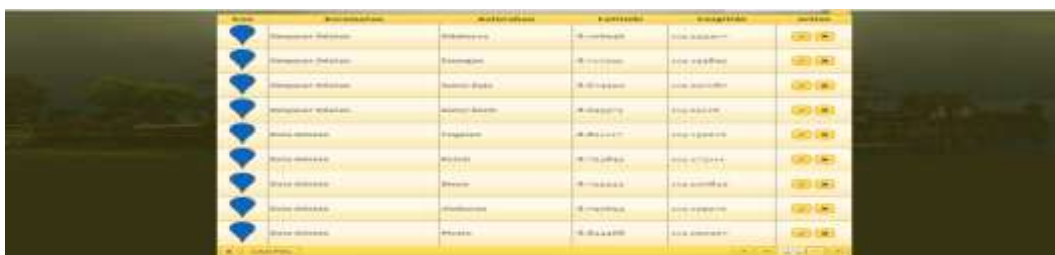
Pada gambar 4.8 Halaman kecamatan pada admin adalah sebagai pengolahan data pada kecamatan untuk pengolahan *edit*, *delete* yang akan di lakukan oleh admin dan akan menampilkan kecamatan.



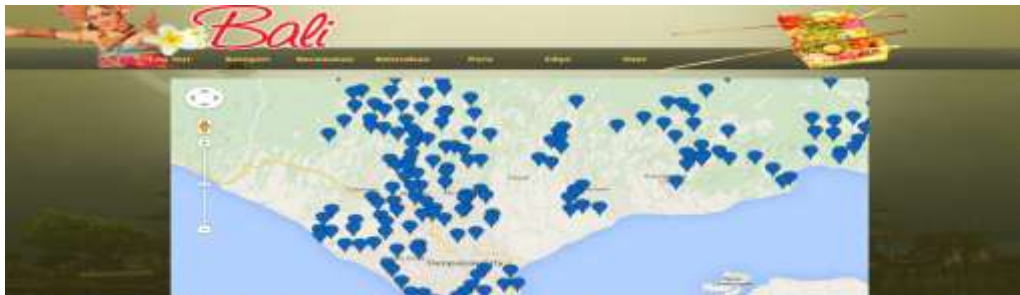
Gambar 4.8 Form Halaman Kecamatan

h. Form Halaman Kelurahan

Pada gambar 4.9 Form Halaman kelurahan pada admin adalah sebagai pengolahan data pada kelurahan untuk pengolahan *edit*, *delete* yang akan di lakukan oleh admin dan akan menampilkan diantaranya kecamatan dan kelurahan dan pada gambar 4.10 menampilkan peta kelurahan.



Gambar 4.9 Form Halaman Kelurahan



Gambar 4.10 Peta Kelurahan

i. Form Halaman Pura

Pada gambar 4.11 form Halaman Pura pada admin adalah sebagai pengolahan data pada *node* untuk pengolahan *edit*, *delete* dan tombol detail akan menampilkan pura secara detail yang akan di lakukan oleh admin.



Gambar 4.11 Form Halaman Pura

j. Form Halaman Edge

Pada gambar 4.12 Form Halaman Edge pada admin adalah sebagai pengolahan data pada *edge* untuk pengolahan *edit* dan *delete* yang akan di lakukan oleh admin.



Gambar 4.12 Form Halaman Edge

5. KESIMPULAN

Proses rute terpendek menggunakan metode djikstra dimana node – node diambil berdasarkan kelurahan untuk mendapatkan rute terpendek di perlukan perhitungan haversine. Haversine digunakan untuk menghitung jarak sesuai dengan edge yang terhubung antar node yaitu kelurahan ke kelurahan lainnya. Sistem ini dapat menampilkan hasil rute terpendek dan dapat membantu user yang kesulitan dalam menemukan lokasi dan keterangan Kahyangan Jagad di Bali.

Daftar Pustaka

- [1] Setiawan, I Ketut, Maret. Menelusuri Asal-Usul Tempat Suci Di Bali Dalam Rangka Pengelolaan Sumber daya Budaya. Jurusan Arkeologi Fakultas Sastra Universitas Udayana 2002.
- [2] Gross, L. J dan Yellen, J..Graph Theory and Its Applications. Chapman & Hall/CRC 2006.
- [3] Google Maps, <http://maps.google.com> (Wikipedia.org). (diakses pada 2008)
- [4] Amri, M.Choirul. Pengantar ASP.NET. Tanggal akses 28 Januari 2007. <http://www.ilmukomputer.com/2006/08/19/pengantar-aspnet/>--
- [5] Abdul Kadir.. Pengenalan Sistem Informasi. Yogyakarta : Andi 2003