LAPORAN TUGAS

Pemrograman Berbasis Java

Database Wilayah Indonesia Berdasarkan Master File Desa Badan Pusat Statistik (MFD BPS) Indonesia

Disusun oleh:

Dwi Putra Sudaryanto

12111075

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS MERCUBUANA YOGYAKARTA

DAFTAR ISI

DAFTAR ISI 2
BAB I PENDAHULUAN 4
1.1. Latar Belakang 4
1.2. Alasan Pemilihan Masalah 4
BAB II DATABASE 5
2.1. Rancangan Tabel 5
2.1.1. Tabel 5
2.1.2. Trigger 5
2.1.3. Store Prosedure 7
2.1.4. View 8
2.2. Relasi Database 9
BAB III USER INTERFACE 10
3.1. Welcome 10
3.2. Provinsi 10
3.1.1. Insert 10
3.1.2. Update 10
3.1.3. View 10
3.3. Kabupaten / Kota 11
3.2.1. Insert 11
3.2.2. Update 11
3.2.3. View 12
3.4. Kecamatan 12
3.3.1. Insert 12
3.3.2. Update 13
3.3.3. View 13
3.5. Desa / Kelurahan 14
3.4.1. Insert 14
3.4.2. Update 14
3.4.3. View 15
BAB IV MENU, STRUCTURE DAN TESTING 16
4.1. Menu 16
4.2. Structure 17
4.3. Testing Aplikasi 18
BAB V PENUTUP 19

5.1. Kesimpulan 19
5.2. Kesulitan 19

BAB I
PENDAHULUAN

1.1. Latar Belakang

Dokumen ini merupakan diskripsi dari aplikasi Database Wilayah Indonesia Berdasarkan Master File Desa Badan Pusat Statistik (MFD BPS) Indonesia yang dikembangkan menggunakan bahasa pemograman berbasis java. Dokumen ini dibuat untuk memenuhi tugas mata kuliah Pemograman Berbasis Java.

1.2. Alasan Pemilihan Masalah

Master File Desa Badan Pusat Statistik (MFD BPS) Indonesia merupakan acuan dasar yang digunakan oleh pengembang aplikasi baik website, mobile maupun desktop jika ingin mengembangkan sebuah aplikasi yang memiliki basis data daerah Indonesia.

Aplikasi ini dikembangkan untuk mempermudah siapapun baik personal maupun organisasi untuk mengelola database wilayah Indonesia menggunakan konsep MFD online untuk selanjutnya dapat dikembangkan kembali.

BAB II
DATABASE

2.1. Rancangan Tabel

2.1.1. Tabel

Tebel	Colom
ommu_core_zone_province	`province_id` smallint(5), `publish` tinyint(1), `province` varchar(64), `mfdonline` char(2), `creation_date`, `modified_date`
ommu_core_zone_city	`city_id` int(11), `publish` tinyint(1), `province_id` smallint(5), `city` varchar(64) , `mfdonline` char(4), `creation_date`, `modified_date`
ommu_core_zone_districts	`district_id` int(11), `publish` tinyint(1), `city_id` int(11), `district_name` varchar(64), `mfdonline` char(7), `creation_date`, `modified_date`
ommu_core_zone_village	`village_id`, `publish`, `district_id`, `village_name`, `zipcode`, `mfdonline`, `creation_date`, `modified_date`

2.1.2. Trigger

Name	Deskripsi

coreBeforeInsertCity	DECLARE province_id_tr SMALLINT;
	
	/*CALL getOmmuZoneProvinceIdWithCityMfdonline(NEW.mfdonline, province_id_tr);*/
	SELECT `province_id` INTO province_id_tr FROM `ommu_core_zone_province` WHERE `mfdonline`=LEFT(NEW.mfdonline,2);
	IF (province_id_tr IS NOT NULL) THEN
	SET NEW.province_id = province_id_tr;
	END IF;
	
	SET NEW.city = TRIM(NEW.city);

coreBeforeUpdateCity	DECLARE province_id_tr SMALLINT;
	
	IF (NEW.mfdonline <> OLD.mfdonline) THEN
	
	/*CALL getOmmuZoneProvinceIdWithCityMfdonline(NEW.mfdonline, province_id_tr);*/
	SELECT `province_id` INTO province_id_tr FROM `ommu_core_zone_province` WHERE `mfdonline`=LEFT(NEW.mfdonline,2);
	IF (province_id_tr IS NOT NULL) THEN

	SET NEW.province_id = province_id_tr; END IF; END IF; SET NEW.city = TRIM(NEW.city); SET NEW.modified_date = NOW();

coreBeforeInsertDistricts	DECLARE `city_id_tr` INT;
	
	/*CALL getOmmuZoneCityIdWithDistrictMfdonline(NEW.mfdonline, city_id_tr);*/
	SELECT `city_id` INTO city_id_tr FROM `ommu_core_zone_city` WHERE `mfdonline`=LEFT(NEW.mfdonline,4);
	IF (city_id_tr IS NOT NULL) THEN
	SET NEW.city_id = city_id_tr;
	END IF;
	
	SET NEW.district_name = TRIM(NEW.district_name);

coreBeforeUpdateDistricts	DECLARE `city_id_tr` INT;
	
	IF (NEW.mfdonline <> OLD.mfdonline) THEN
	
	/*CALL getOmmuZoneCityIdWithDistrictMfdonline(NEW.mfdonline, city_id_tr);*/
	SELECT `city_id` INTO city_id_tr FROM `ommu_core_zone_city` WHERE `mfdonline`=LEFT(NEW.mfdonline,4);
	IF (city_id_tr IS NOT NULL) THEN
	SET NEW.city_id = city_id_tr;
	END IF;
	END IF;
	SET NEW.district_name = TRIM(NEW.district_name);
	SET NEW.modified_date = NOW();

coreBeforeInsertProvince	SET NEW.province = TRIM(NEW.province);
coreBeforeUpdateProvince	SET NEW.province = TRIM(NEW.province); SET NEW.modified_date = NOW();
coreBeforeInsertVillage	DECLARE `district_id_tr` INT; /*CALL getOmmuZoneCityIdWithDistrictMfdonline(NEW.mfdonline, city_id_tr);*/ SELECT `district_id` INTO district_id_tr FROM `ommu_core_zone_districts` WHERE `mfdonline`=LEFT(NEW.mfdonline,7); IF (district_id_tr IS NOT NULL) THEN SET NEW.district_id = district_id_tr;

	END IF; SET NEW.village_name = TRIM(NEW.village_name); SET NEW.zipcode = TRIM(NEW.zipcode);

coreBeforeUpdateVillage	DECLARE `district_id_tr` INT;
	
	IF (NEW.mfdonline <> OLD.mfdonline) THEN
	
	/*CALL getOmmuZoneCityIdWithDistrictMfdonline(NEW.mfdonline, city_id_tr);*/
	SELECT `district_id` INTO district_id_tr FROM `ommu_core_zone_districts` WHERE `mfdonline`=LEFT(NEW.mfdonline,7);
	IF (district_id_tr IS NOT NULL) THEN
	SET NEW.district_id = district_id_tr;
	END IF;
	END IF;
	SET NEW.village_name = TRIM(NEW.village_name);
	SET NEW.zipcode = TRIM(NEW.zipcode);
	SET NEW.modified_date = NOW();

2.1.3. Store Prosedure

Prosedure
/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `getOmmuZoneCityIdWithDistrictMfdonline`(IN `mfdonline_sp` CHAR, OUT `city_id_sp` INT) BEGIN SELECT `city_id` INTO city_id_sp FROM `ommu_core_zone_city` WHERE `mfdonline`=LEFT(mfdonline_sp,4); END */$$ DELIMITER ;
/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `getOmmuZoneCountryId`(IN `province_id_sp` SMALLINT, OUT `country_id_sp` SMALLINT) BEGIN SELECT `country_id` INTO country_id_sp FROM `ommu_core_zone_province` WHERE `province_id`=province_id_sp; END */$$ DELIMITER ;
/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE `getOmmuZoneProvinceId`(in `city_id_sp` INT, OUT `province_id_sp` SMALLINT) BEGIN SELECT `province_id` INTO province_id_sp FROM `ommu_core_zone_city` WHERE `city_id`=city_id_sp;

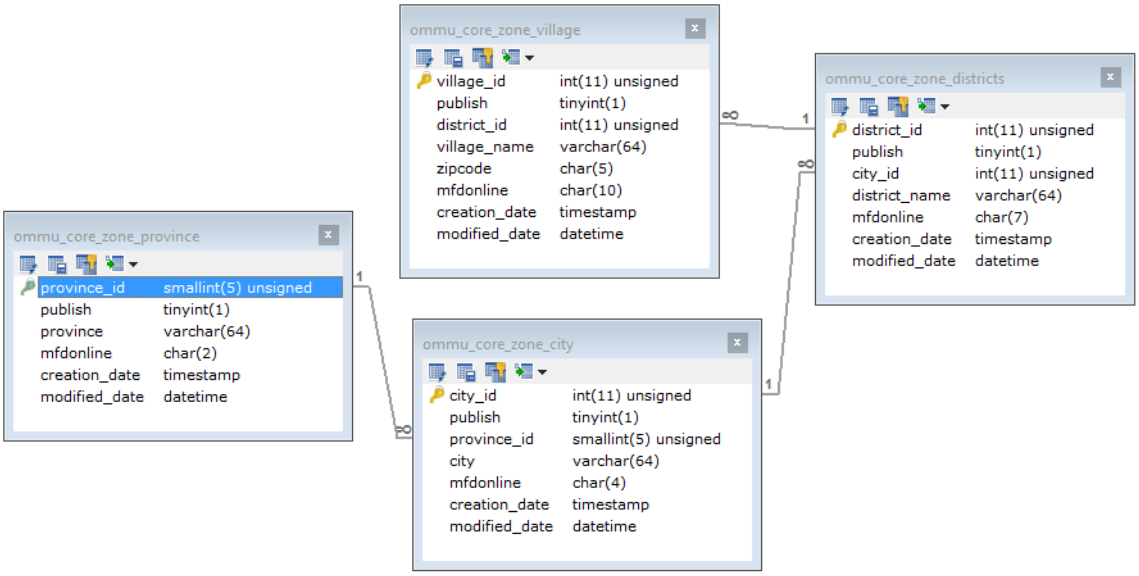
```
    END */$$
DELIMITER ;
```

```
/*!50003 CREATE DEFINER=`root`@`localhost` PROCEDURE
`getOmmuZoneProvinceIdWithCityMfdonline`(IN `mfdonline_sp` CHAR, OUT
`province_id_sp` SMALLINT)
BEGIN
     SELECT `province_id` INTO province_id_sp FROM
`ommu_core_zone_province` WHERE `mfdonline`=LEFT(mfdonline_sp,2);
     /*
     CALL getOmmuZoneProvinceIdWithCityMfdonline(NEW.mfdonline,
province_id_tr);
     SELECT `province_id` INTO province_id_tr FROM
`ommu_core_zone_province` WHERE `mfdonline`=LEFT(NEW.mfdonline,2);*/
    END */$$
DELIMITER ;
```

2.1.4. View

Name and View	Deskripsi
`_view_core_zone_city`(`city_id` int(11) unsigned , `city` varchar(64) , `province` varchar(64))	select `a`.`city_id` AS `city_id`, `a`.`city` AS `city`, `b`.`province` AS `province` from (`ommu_core_zone_city` `a` left join `ommu_core_zone_province` `b` on((`a`.`province_id` = `b`.`province_id`)))
`_view_core_zone_districts`(`district_id` int(11) unsigned , `district_name` varchar(64) , `city` varchar(64) , `province` varchar(64))	select `a`.`district_id` AS `district_id`, `a`.`district_name` AS `district_name`, `b`.`city` AS `city`, `b`.`province` AS `province` from (`ommu_core_zone_districts` `a` left join `_view_core_zone_city` `b` on((`a`.`city_id` = `b`.`city_id`)))
`_view_core_zone_village`(`village_id` int(11) unsigned , `village_name` varchar(64) , `district_name` varchar(64) , `city` varchar(64) , `province` varchar(64))	select `a`.`village_id` AS `village_id`, `a`.`village_name` AS `village_name`, `b`.`district_name` AS `district_name`, `b`.`city` AS `city`, `b`.`province` AS `province` from (`ommu_core_zone_village` `a` left join `_view_core_zone_districts` `b` on((`a`.`district_id` = `b`.`district_id`)))

2.2. Relasi Database

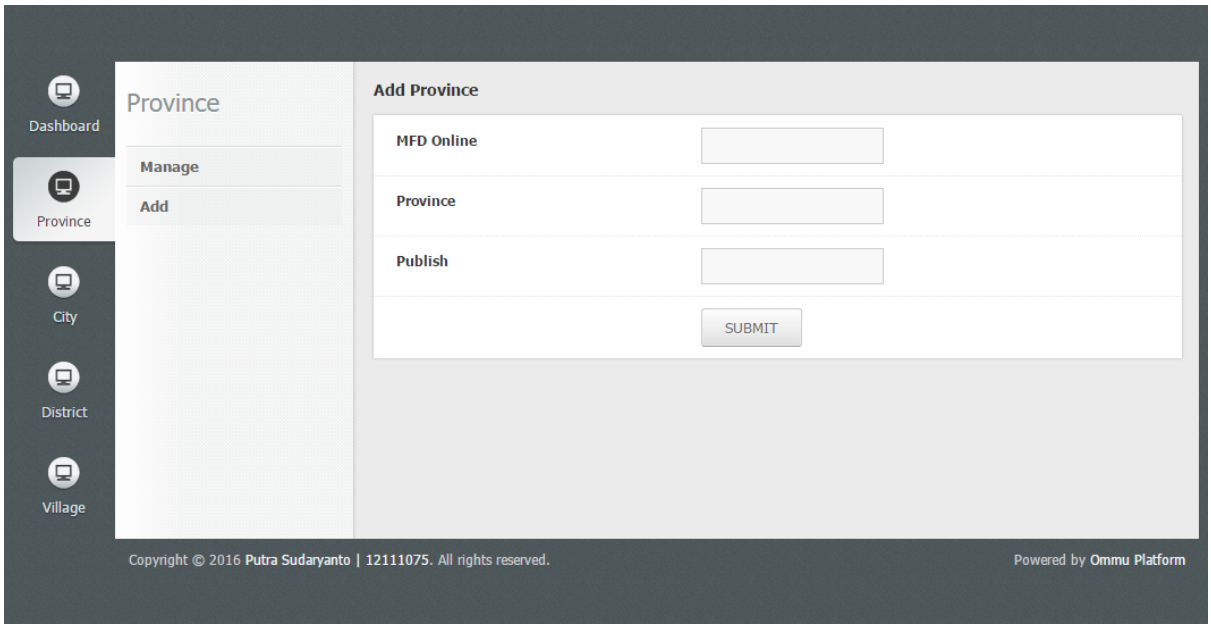


BAB III
USER INTERFACE

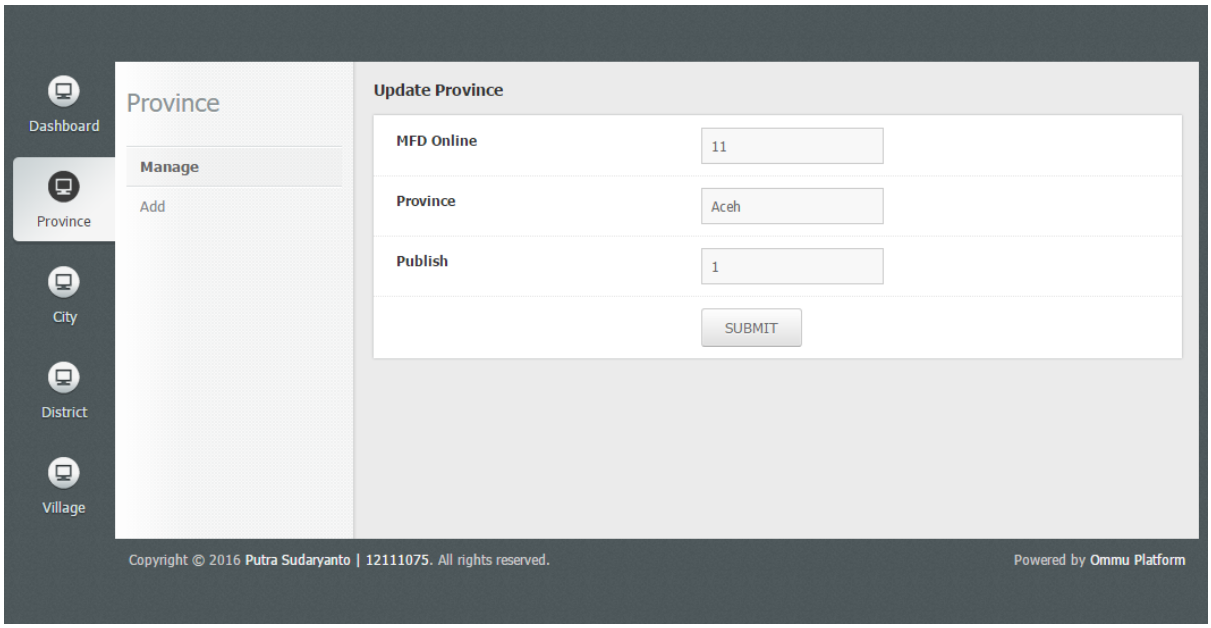
3.1. Welcome

3.2. Provinsi

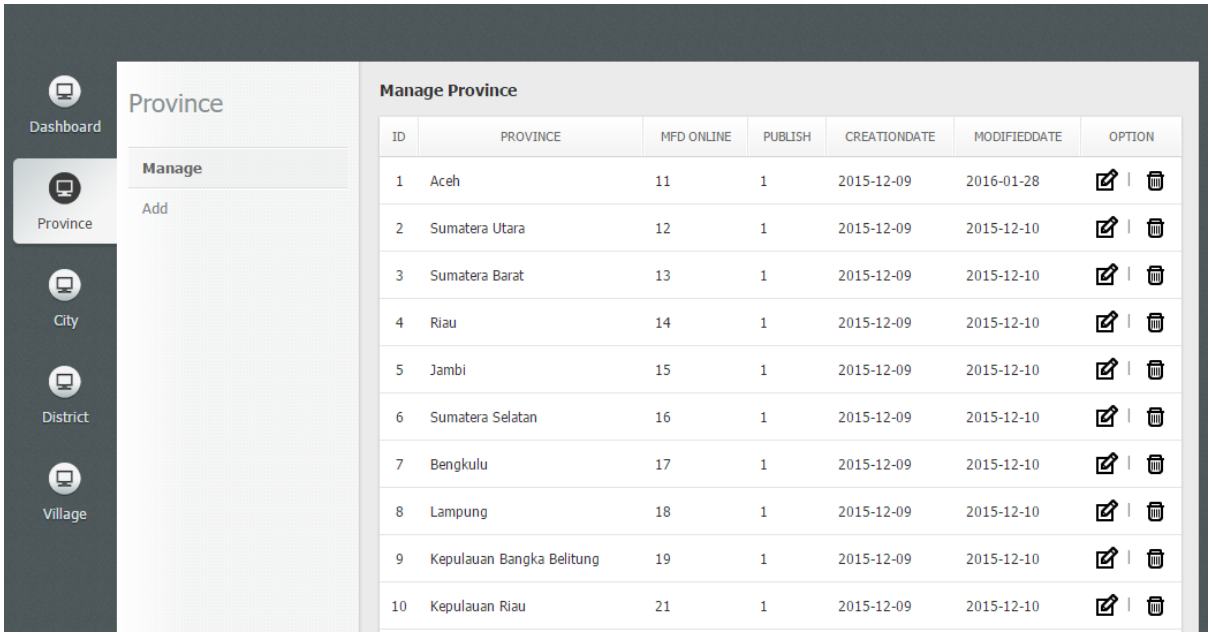
3.1.1. Insert



3.1.2. Update

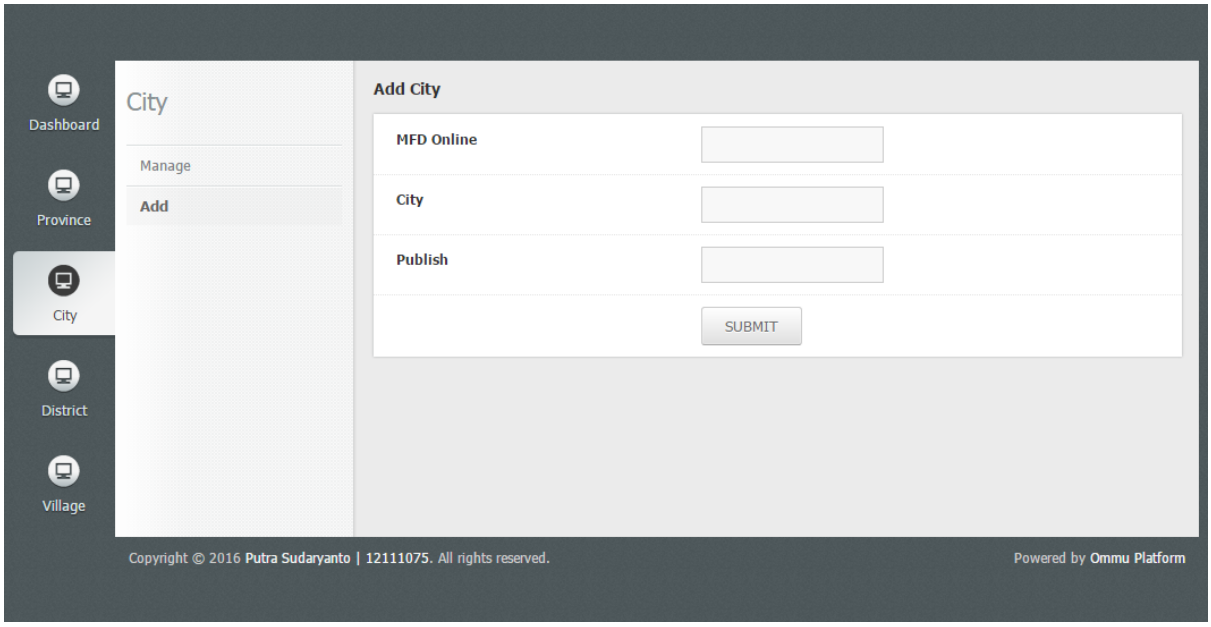


3.1.3. View

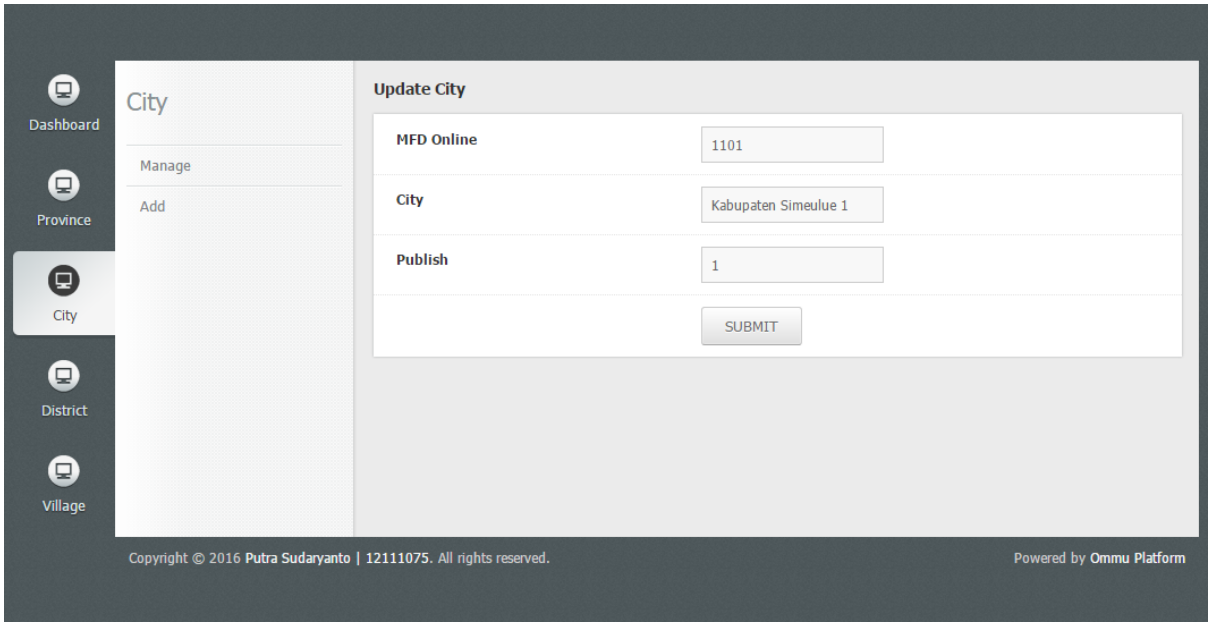


3.3. Kabupaten / Kota

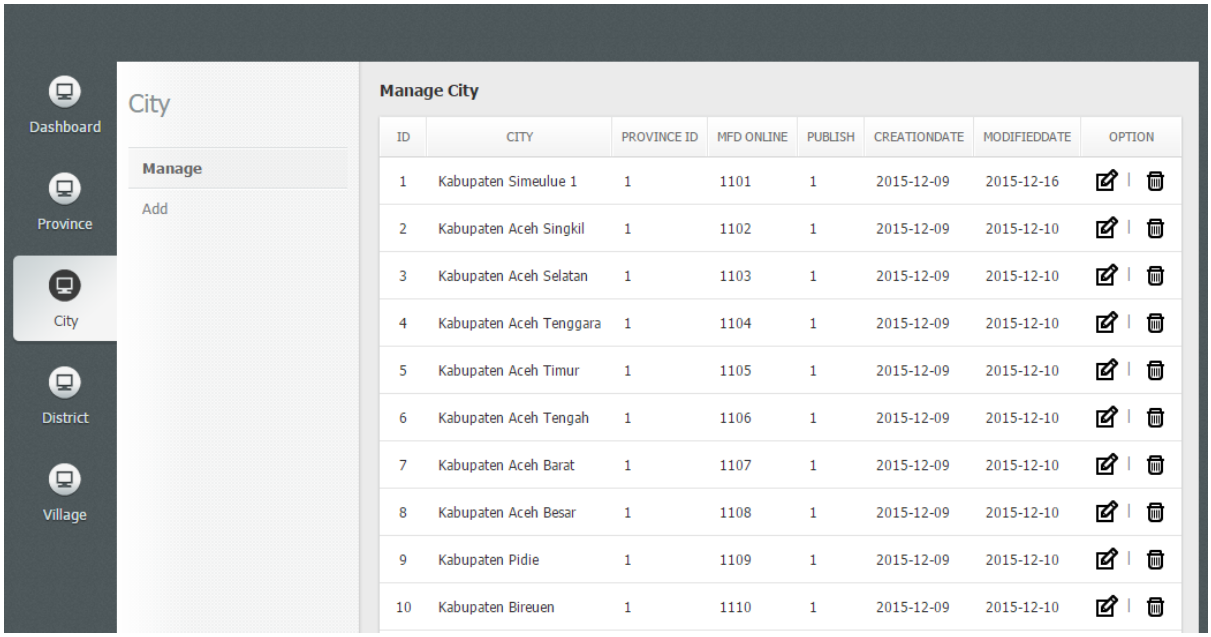
3.2.1. Insert



3.2.2. Update

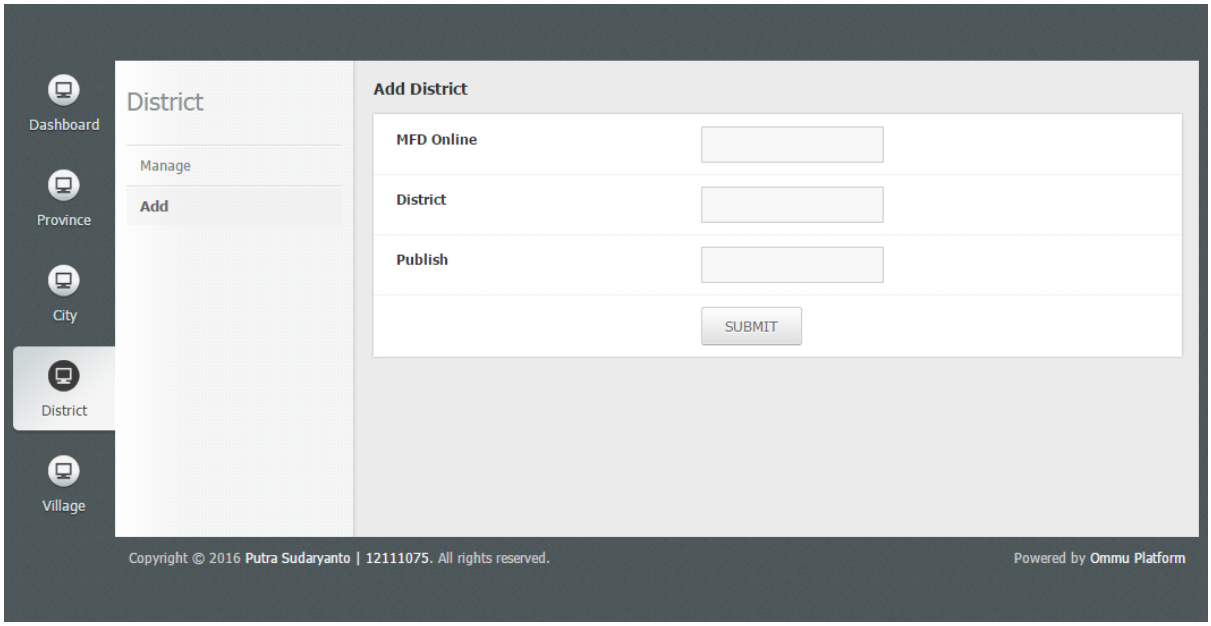


3.2.3. View

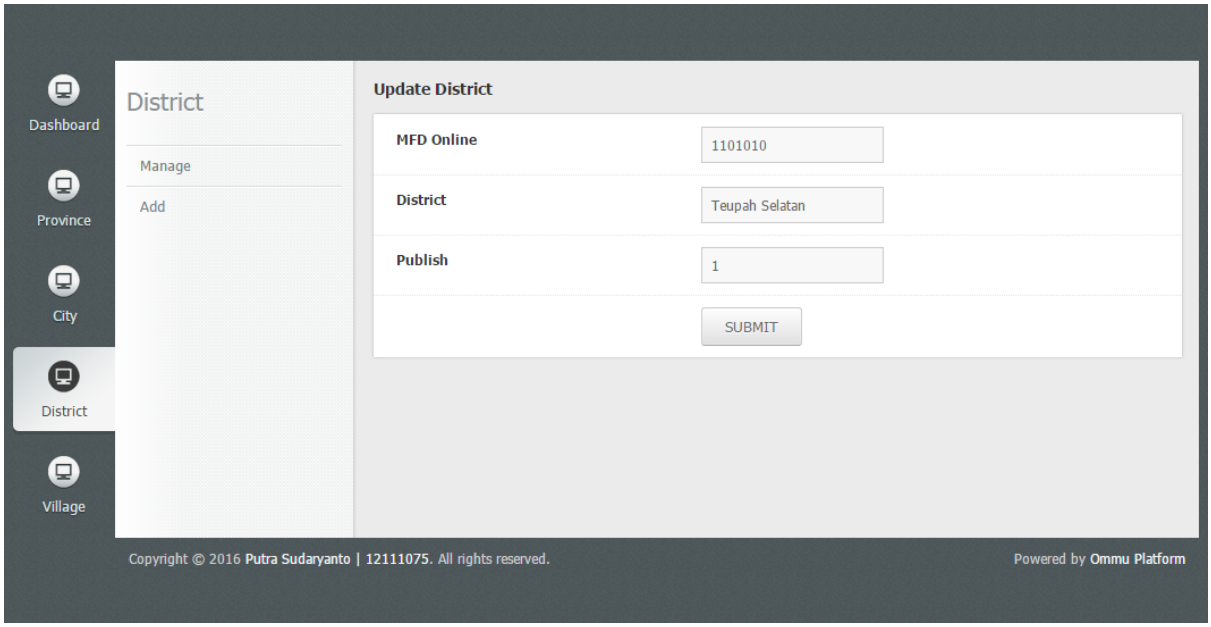


3.4. Kecamatan

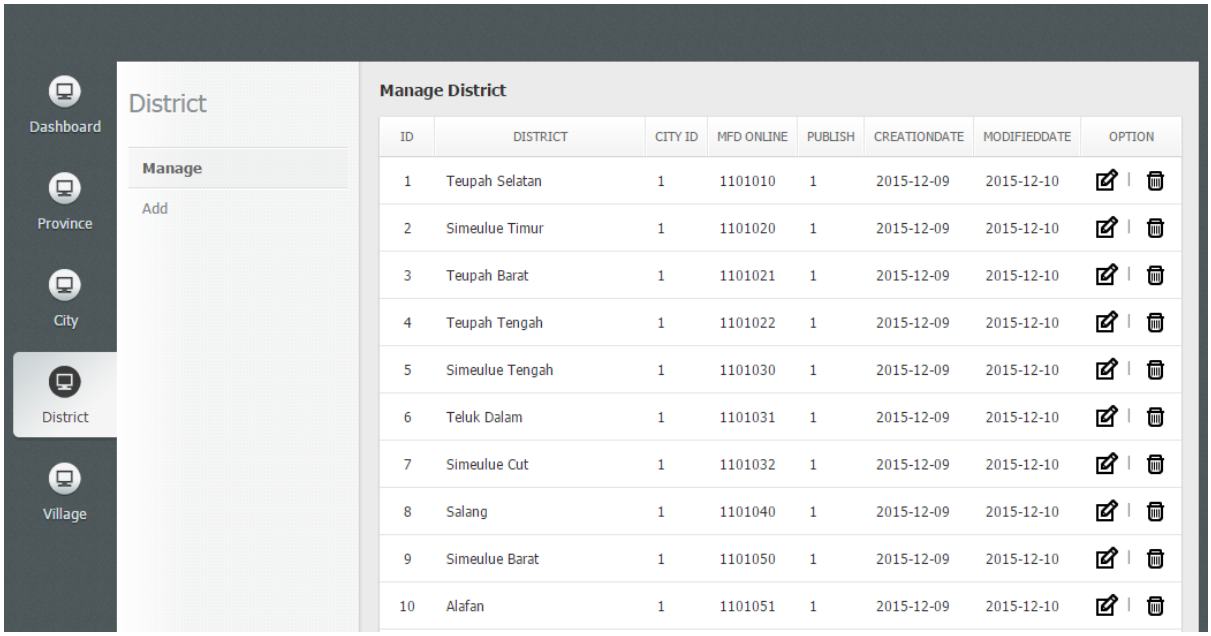
3.3.1. Insert



3.3.2. Update

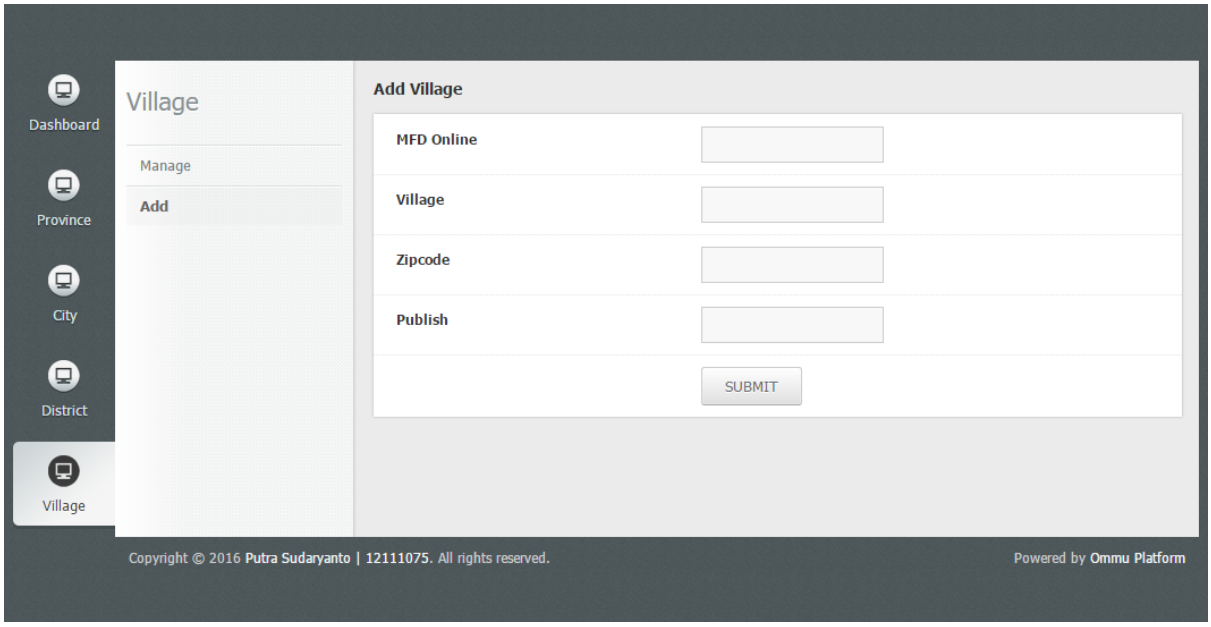


3.3.3. View

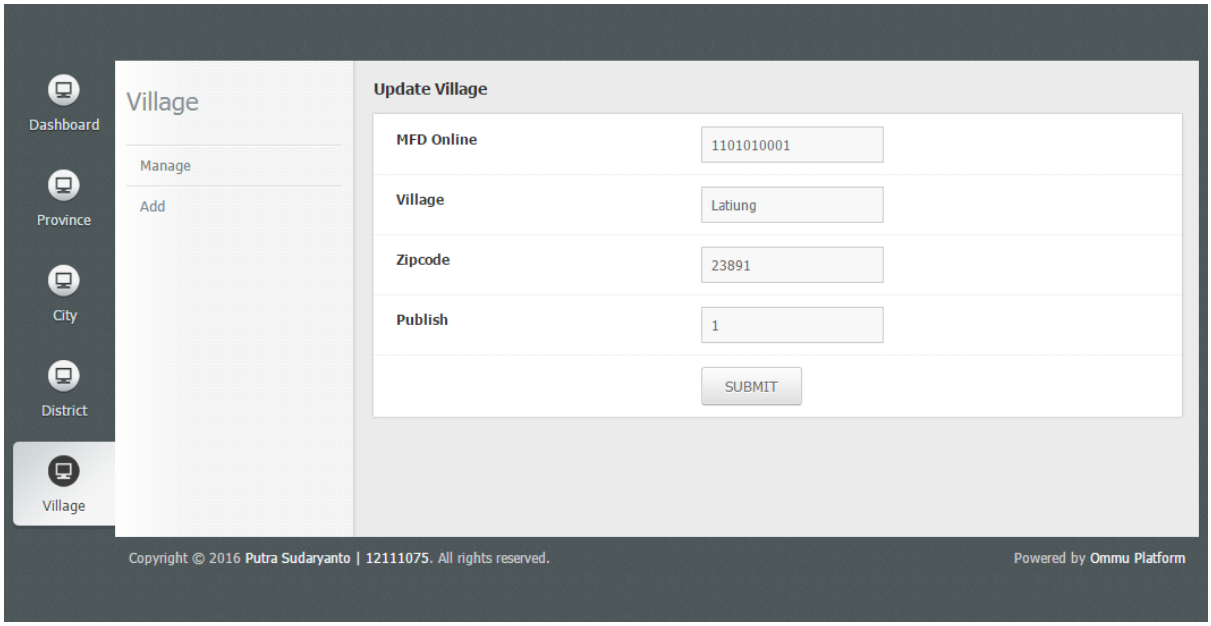


3.5. Desa / Kelurahan

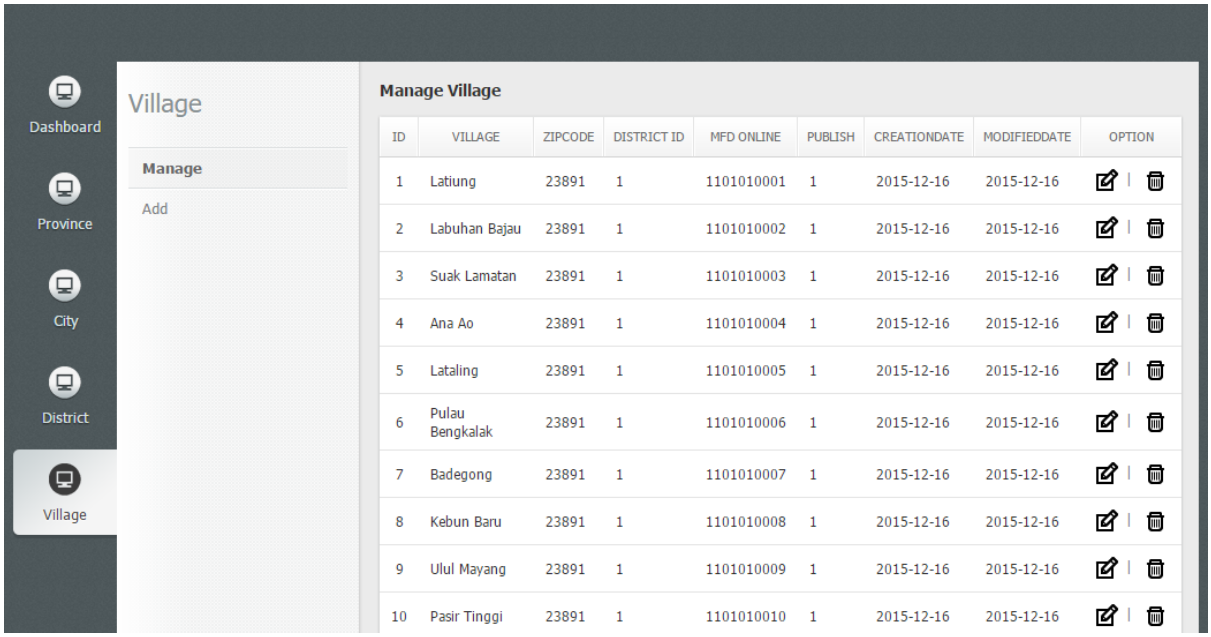
3.4.1. Insert



3.4.2. Update


Dashboard
Province
City
District
Village
Village
Manage
Add
Update Village
MFD Online
1101010001
Village
Latiung
Zipcode
23891
Publish
1
SUBMIT
Copyright © 2016 Putra Sudaryanto | 12111075. All rights reserved.
Powered by Ommu Platform

3.4.3. View


Dashboard
Province
City
District
Village
Village
Manage
Add
Manage Village
ID VILLAGE ZIPCODE DISTRICT ID MFD ONLINE PUBLISH CREATIONDATE MODIFIEDDATE OPTION
1 Latiung 23891 1 1101010001 1 2015-12-16 2015-12-16
2 Labuhan Bajau 23891 1 1101010002 1 2015-12-16 2015-12-16
3 Suak Lamatan 23891 1 1101010003 1 2015-12-16 2015-12-16
4 Ana Ao 23891 1 1101010004 1 2015-12-16 2015-12-16
5 Lataling 23891 1 1101010005 1 2015-12-16 2015-12-16
6 Pulau Bengkalak 23891 1 1101010006 1 2015-12-16 2015-12-16
7 Badegong 23891 1 1101010007 1 2015-12-16 2015-12-16
8 Kebun Baru 23891 1 1101010008 1 2015-12-16 2015-12-16
9 Ulul Mayang 23891 1 1101010009 1 2015-12-16 2015-12-16
10 Pasir Tinggi 23891 1 1101010010 1 2015-12-16 2015-12-16

BAB IV
MENU, STRUCTURE DAN TESTING

4.1. Menu

Mainmenu	Submenu	Content Menu
Dashboard	Manage	Update
		Delete
	Add	-
Province	Manage	Update
		Delete
	Add	-
City	Manage	Update
		Delete
	Add	-
District	Manage	Update
		Delete
	Add	-
Village	Manage	Update
		Delete
	Add	-

4.2. Structure

- Controller
  - ✓ City.java
  - ✓ District.java
  - ✓ Province.java
  - ✓ Village.java
  - ✓ Site.java
- Database
  - ✓ Database.java
- Model
  - ✓ CityModel.java
  - ✓ DistrictModel.java
  - ✓ ProvinceModel.java
  - ✓ VillageModel.java
- Page
  - ✓ CityPage.java
  - ✓ DistrictPage.java
  - ✓ ProvincePage.java
  - ✓ VillagePage.java
- View
  - ✓ City
    - admin_delete.jsp
    - admin_form.jsp
    - admin_manage.jsp
  - ✓ District
    - admin_delete.jsp
    - admin_form.jsp
    - admin_manage.jsp
  - ✓ Province
    - admin_delete.jsp
    - admin_form.jsp
    - admin_manage.jsp
  - ✓ Site
    - admin_delete.jsp
    - admin_form.jsp
    - admin_manage.jsp
  - ✓ Village

- admin_delete.jsp
- admin_form.jsp
- admin_manage.jsp

✓ web.xml

4.3. Testing Aplikasi

Mainmenu	Function Menu	Result	
		Yes	No
Dashboard	Manage	√	
	Add	√	
	Update	√	
	Delete	√	
Province	Manage	√	
	Add	√	
	Update	√	
	Delete	√	
City	Manage	√	
	Add	√	
	Update	√	
	Delete	√	
District	Manage	√	
	Add	√	
	Update	√	
	Delete	√	
Village	Manage	√	
	Add	√	
	Update	√	
	Delete	√	

BAB V
PENUTUP

5.1. Kesimpulan

Aplikasi yang dikembangkan dengan bahasa pemrograman Java tidah hanya selalu berupa aplikasi desktop tapi juga bisa berupa aplikasi berbasis website. Dengan bantuan JDBC kita bisa mengembangkan sebuah aplikasi berbasis website dengan bahasa pemrograman java yang bersifat dinamis dan terhubung dengan dengan basis data.

5.2. Kesulitan

Kesulitan yang saya alami dalam mengembangkan aplikasi berbasis website dengan Bahasa pemrograman java dan fitur JDBC, diantaranya:

1. Model, untuk memasukkan sebuah value hasil dari relation database harus mendefiniskan variable yang akan dimasukkan. Sering terjadi kesalahan dan keluapaan nama variable karena netbean tidak ada suggestion nama variable.
2. Berbeda dengan nama variable yang dapat digunakan, untuk java _ (underscore) tidak bisa digunakan dengan error *properly,* ini yang menjadikan java tidak terlalu familiar dibandingkan bahasa pemrograman yang lain.
3. Dropdown, Autosuggest, Checkbox and Radio Condition (Validation), terlalu manual
4. Dalan kesulitan-kesulitan lainnya.. ☺☺