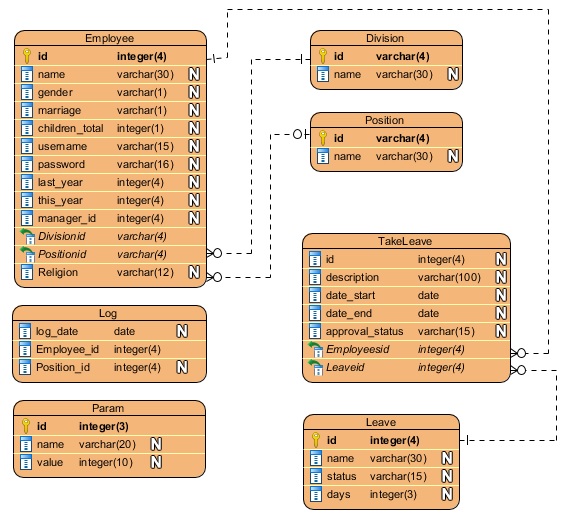
**Laporan Progress Aplikasi Pengambilan Cuti**

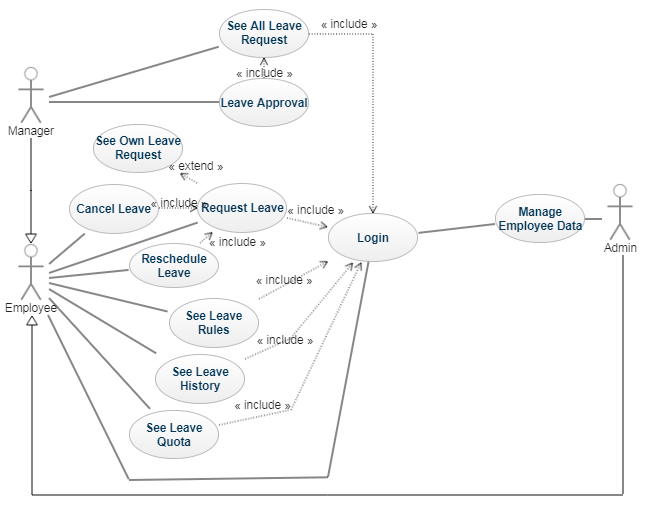
1. Entity Relationship Diagram



Pada ERD aplikasi ini, terdapat entity berupa employee, division, leavetaken, log, takeleave, position,param dan leave. Entity employees memiliki relasi one to many dengan take leave. Entity employees memiliki relasi many to one dengan division dan position. Enity log dan param tidak memiliki relasi. Entity leave memiliki relasi one to many dengan takeleave.

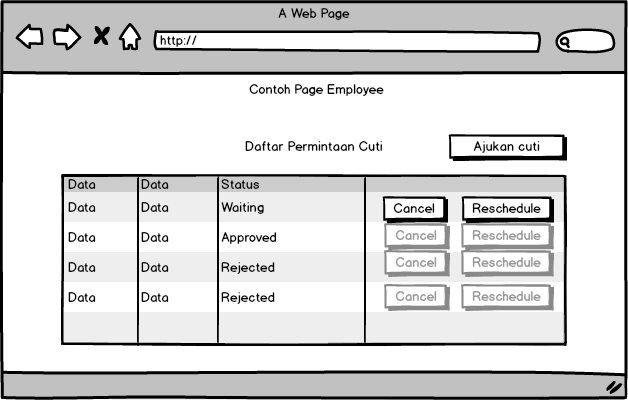
1. Unified Modelling Language (Use Case, Activity, Class)

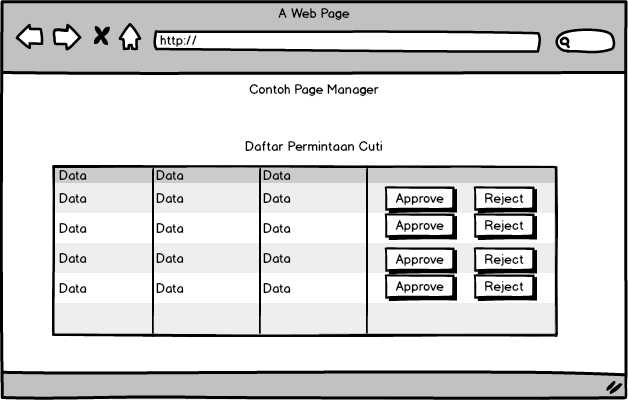
* Use Case Diagram

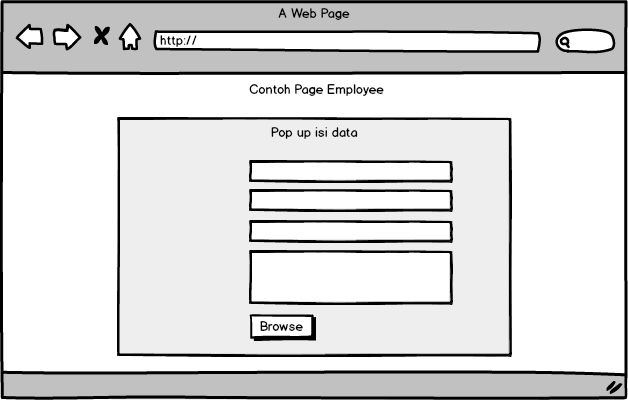
Dalam use case aplikasi ini, terdapat 3 user yaitu admin, manager dan employee dimana manager dan admin memiliki hubungan generalisasi terhadap employee.

* Employee dapat melakukan request leave(cuti), membatalkan request, menjadwal ulang cuti, melihat request cuti, melihat aturan cuti, melihat sejarah cuti, dan melihat quota cuti yang tersisa. Semua use case tersebut baru dapat dilakukan setelah login.
* Manager dapat melihat dan mensetujui permintaan cuti. Semua use case tersebut baru dapat dilakukan setelah login
* Admin dapat menambah, melihat, mengubah, dan menghapus data employee. Semua use case tersebut baru dapat dilakukan setelah login

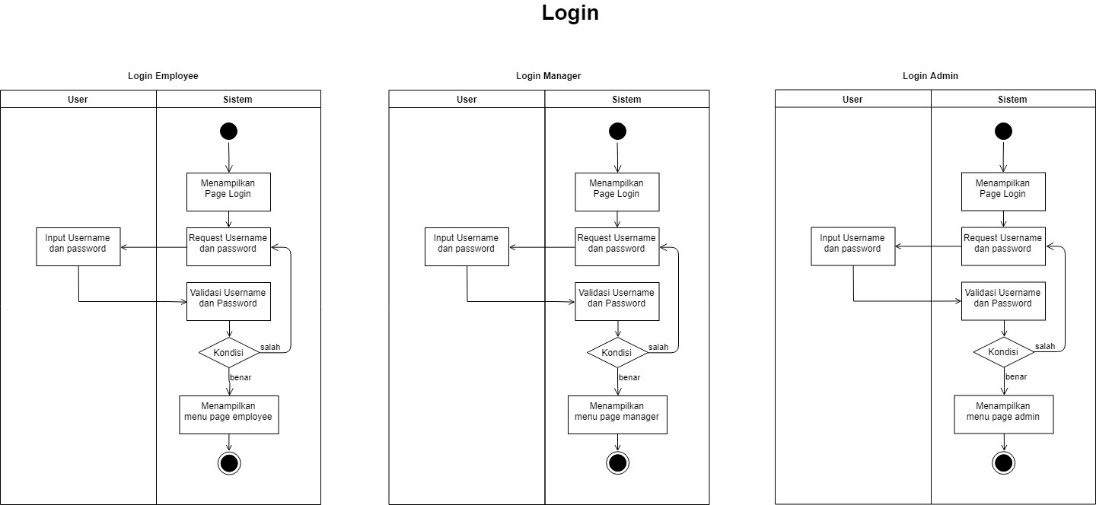
1. Contoh desain tampilan aplikasi

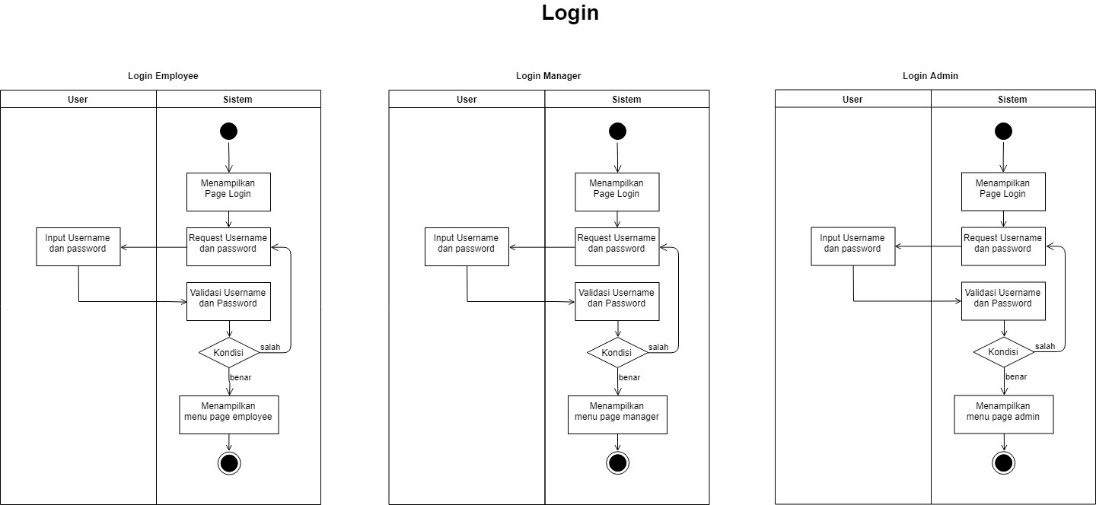


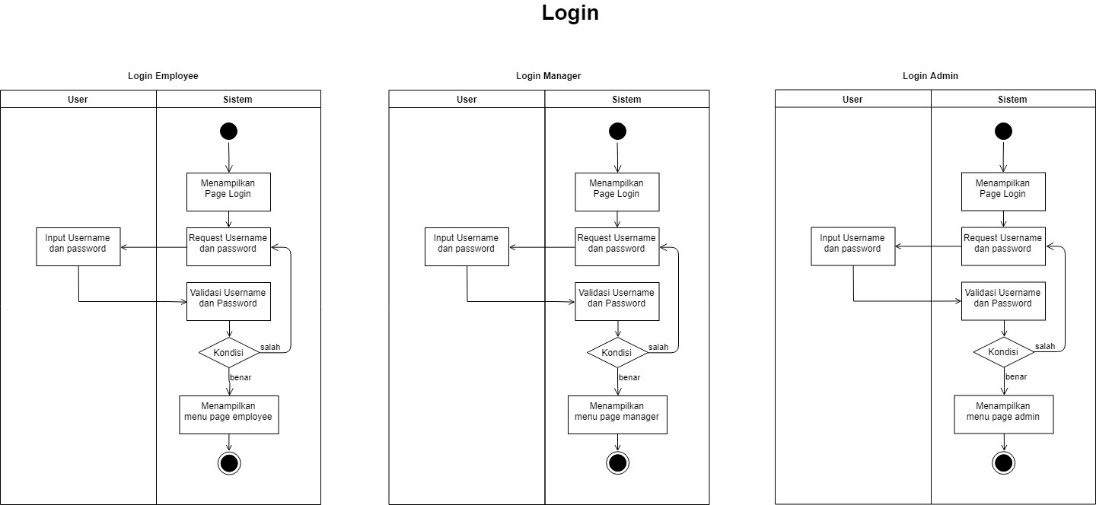




1. Activity Diagram
   1. Activity Login

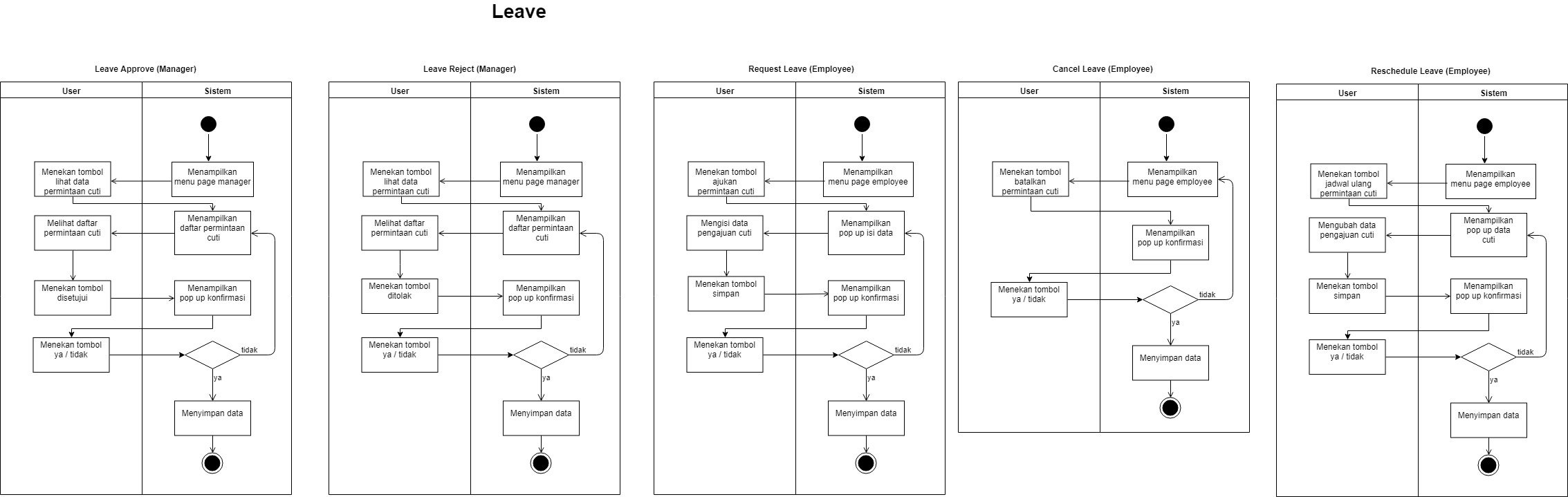


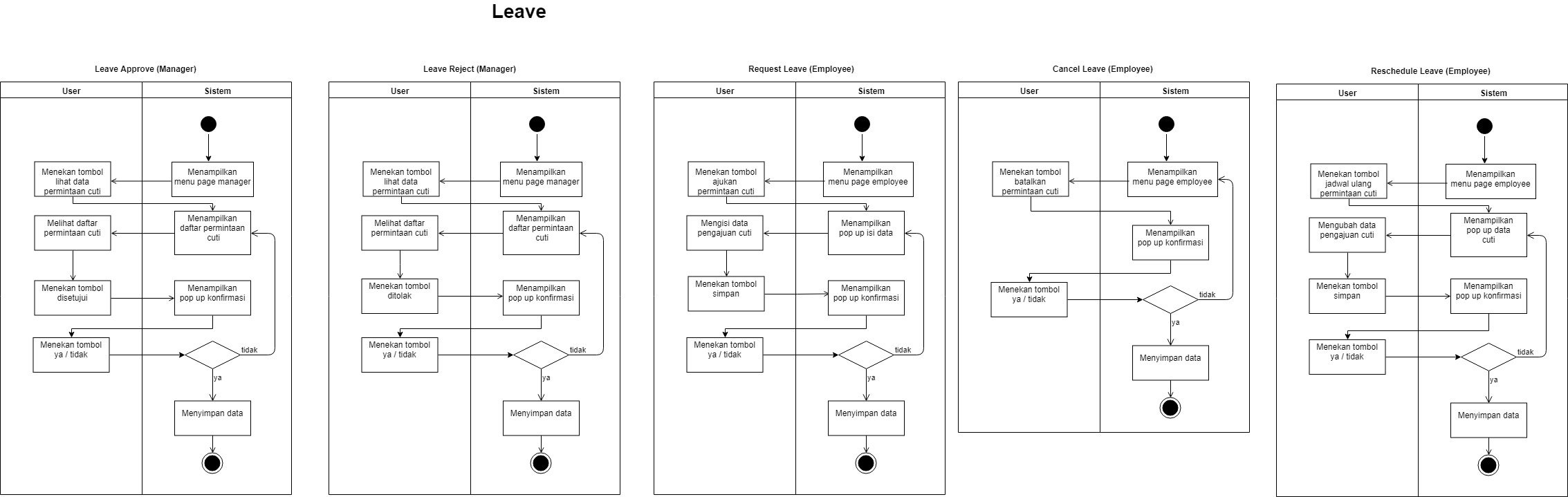


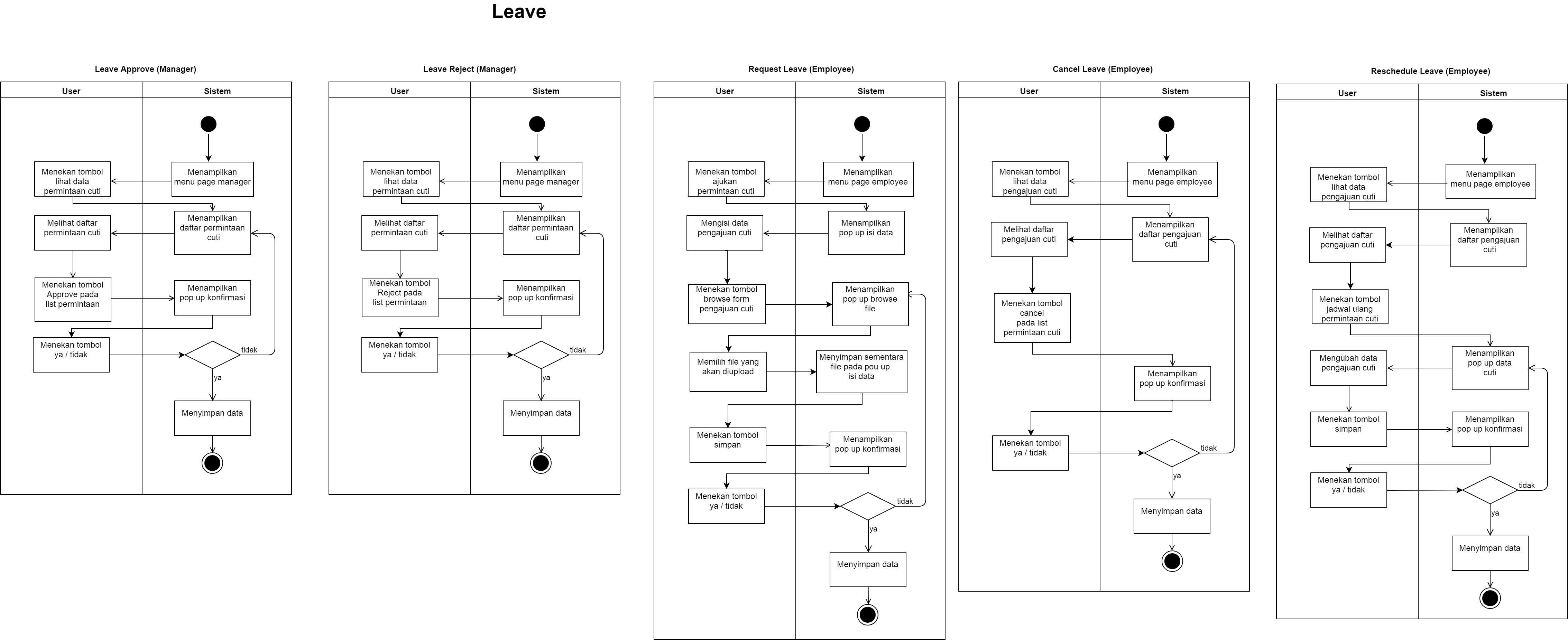


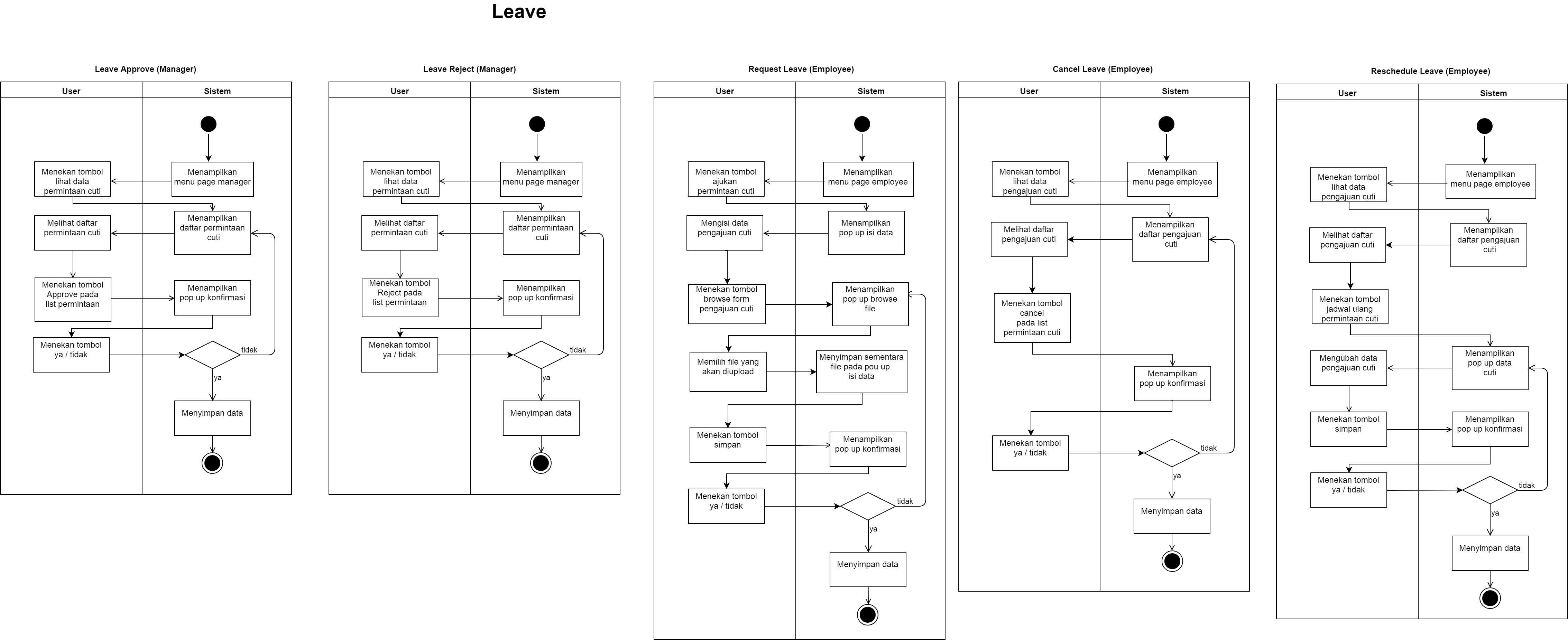
Pada awalnya, sistem akan menampilkan page login, lalu akan merequest username dan password, user mengisikan username dan password, lalu sistem akan memvalidasi username dan password. Jika salah, maka akan merequest kembali username dan password. Jika benar maka tampilkan menu berdasarkan siapa yang login (jika employee maka masuk page employee, jika manager maka masuk page manager, jika admin maka akan masuk page admin).

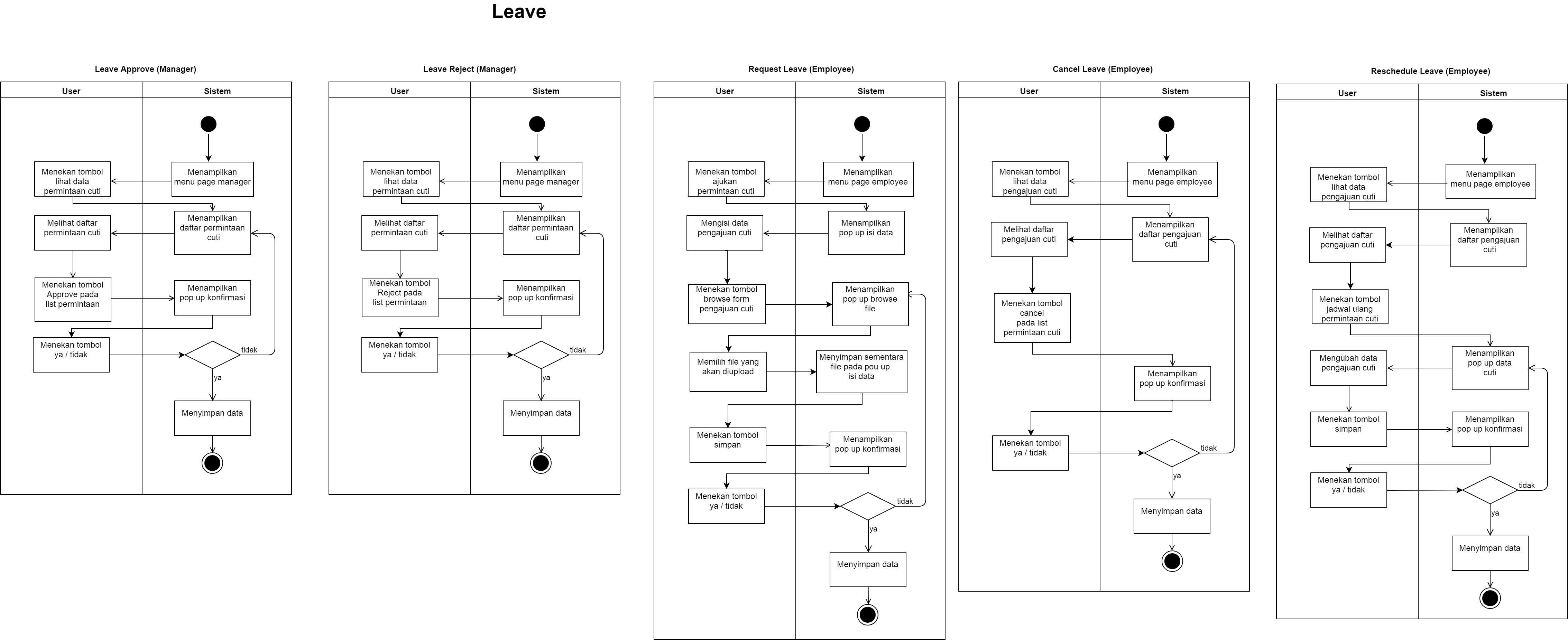
* 1. Activity request leave, cancel leave, reschedule leave, leave approve dan leave reject











Untuk request leave, pertama-tama sistem akan menampilkan menu page employee, lalu user menekan tombol ajukan permintaan cuti, sistem akan menampilkan pop up isi data, user mengisi data pengajuan cuti dan mengupload file form pengajuan cuti kemudian sistem menampilkan pop up konfirmasi, user memilih untuk menekan tombol ya atau tidak. Jika ya, maka sistem akan memvalidasi apakah data yang diisi apakah sudah sesuai, jika sesuai maka sistem akan menyimpan data pengajuan cuti. Jika tidak, maka sistem akan menampilkan pop up isi data lagi.

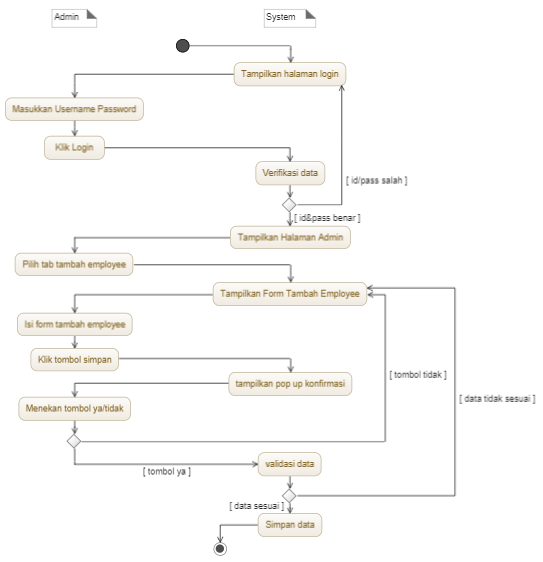
Untuk cancel leave, pertama-tama sistem akan menampilkan menu page employee, user menekan tombol batalkan permintaan cuti, sistem menampilkan pop up konfirmasi, user dapat memilih untuk menekan tombol ya atau tidak. Jika ya, maka sistem akan menyimpan data pengajuan cuti. Jika tidak, maka sistem akan menampilkan menu employee lagi.

Untuk reschedule leave, pertama-tama sistem akan menampilan menu page employee, user menekan tombol reschedule permintaan cuti, sistem akan menampilkan pop up data cuti, user mengubah data pengajuan cuti, user menekan tombol simpan, sistem akan menampilkan pop up konfirmasi, user dapat memilih untuk menekan tombol ya atau tidak. Jika ya, maka sistem akan memvalidasi apakah data sudah benar atau belum ,jika ya maka sistem akan menyimpan data. Jika tidak, maka sistem akan menampilkan pop up data cuti lagi.

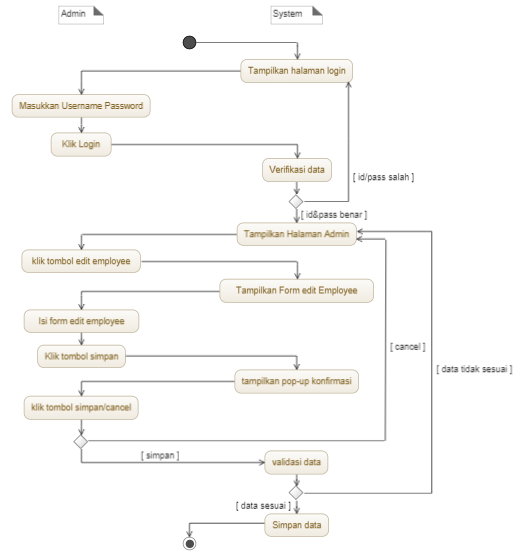
Untuk leave approve, pertama-tama sistem akan menampilan menu page manager, manager menekan tombol lihat data permintaan cuti, sistem menampilkan data permintaan cuti, manager melihat daftar permintaan cuti, manager menekan tombol disetujui, sistem menampilkan pop up konfirmasi, manager memilih untuk menekan tombol ya atau tidak. Jika tombol ya, maka permintaan cuti disetujui. Jika tombol tidak maka sistem akan menampilkan data permintaan cuti lagi.

Untuk leave reject, pertama-tama sistem akan menampilan menu page manager, manager menekan tombol lihat data permintaan cuti, sistem menampilkan data permintaan cuti, manager melihat daftar permintaan cuti, manager menekan tombol ditolak, sistem menampilkan pop up konfirmasi, manager memilih untuk menekan tombol ya atau tidak. Jika tombol ya, maka permintaan cuti disetujui. Jika tombol tidak maka sistem akan menampilkan data permintaan cuti lagi.

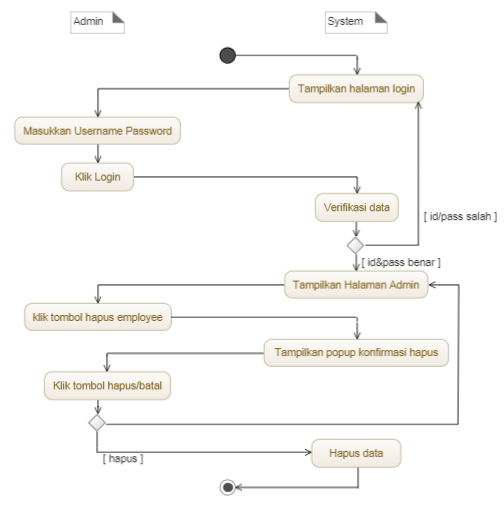
* 1. Activity tambah, lihat, edit, hapus employee



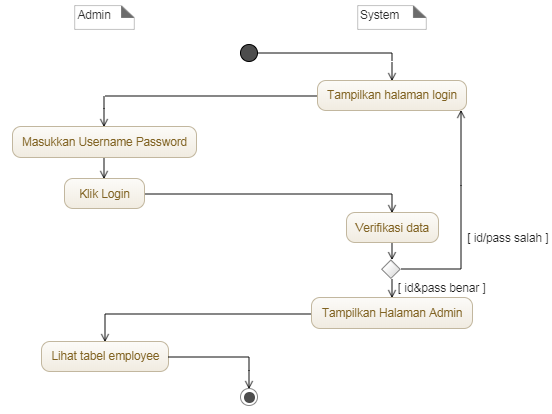
Untuk activity tambah employee, sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman admin, user memilih tab employee, sistem menampilkan form tambah employee, admin mengisi form tambah employee, admin mengklik tombol simpan, lalu sistem akan menyimpan isi form.



Untuk activity edit employee sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman admin, admin menekan tombol edit employee, sistem menampilkan form edit employee, admin mengisi form edit employee, admin mengklik tombol simpan, lalu sistem akan menyimpan isi form.

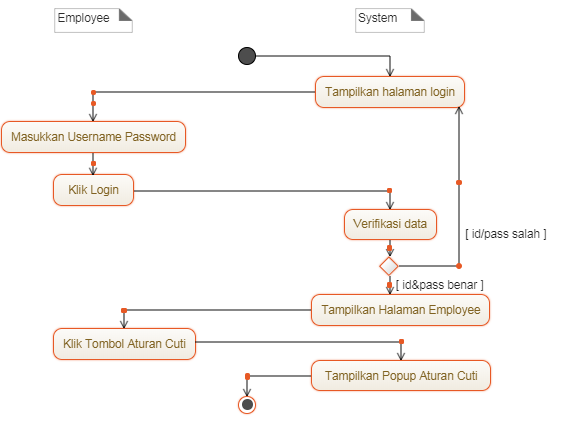


Untuk activity hapus employee sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman admin, admin menekan tombol hapus employee, sistem menampilkan pop up konfirmasi hapus, admin menekan tombol hapus atau batal. Jika tombol hapus ditekan, maka sistem akan menghapus data. Jika batal maka sistem akan menampilkan halaman admin lagi.

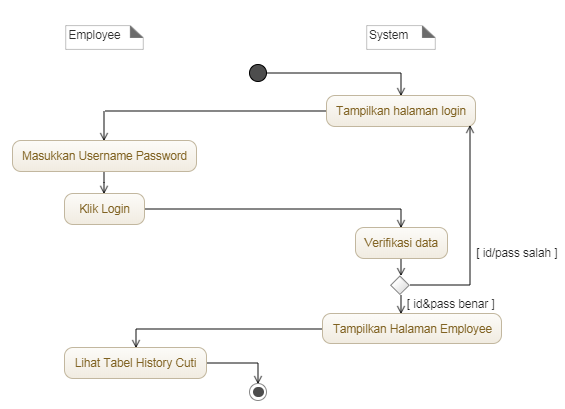


Untuk activity lihat employee sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman admin, lalu admin dapat melihat tabel employee.

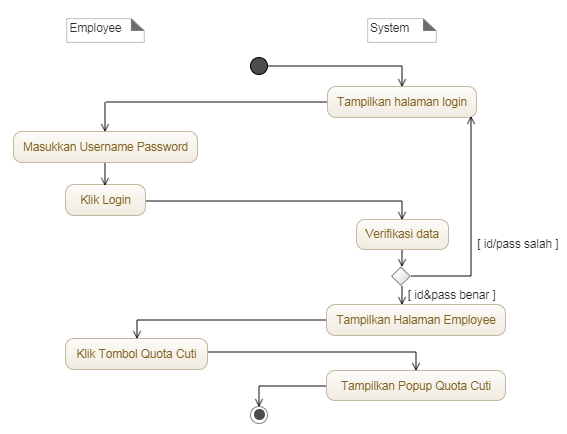
* 1. Activity lihat aturan cuti, sejarah cuti, dan lihat quota cuti



Untuk activity lihat aturan cuti sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman employee, employee mengklik tombol aturan cuti, dan sistem akan menampilkan popup aturan cuti.

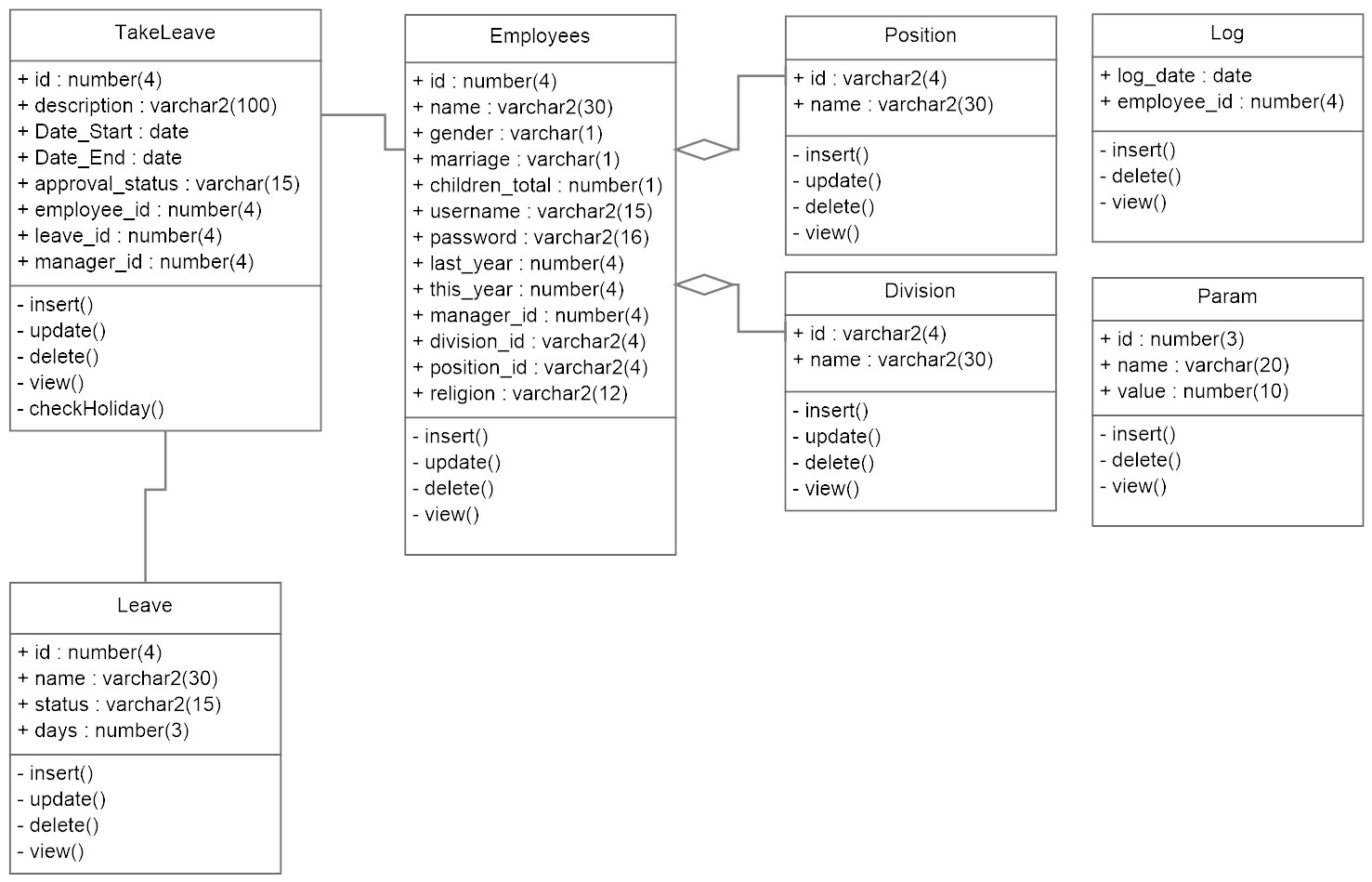


Untuk activity lihat history cuti sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman employee, employee lalu melihat tabel sejarah cuti.



Untuk activity lihat history cuti sistem menampilkan halaman login, admin memasukkan username dan password, admin mengklik tombol login, sistem akan memverifikasi data. Jika salah maka sistem akan menampilkan halaman login lagi. Jika benar maka sistem akan menampilkan halaman employee, employee menekan tombol quota cuti, lalu sistem akan menampilkan pop up quota cuti.

1. Class Diagram

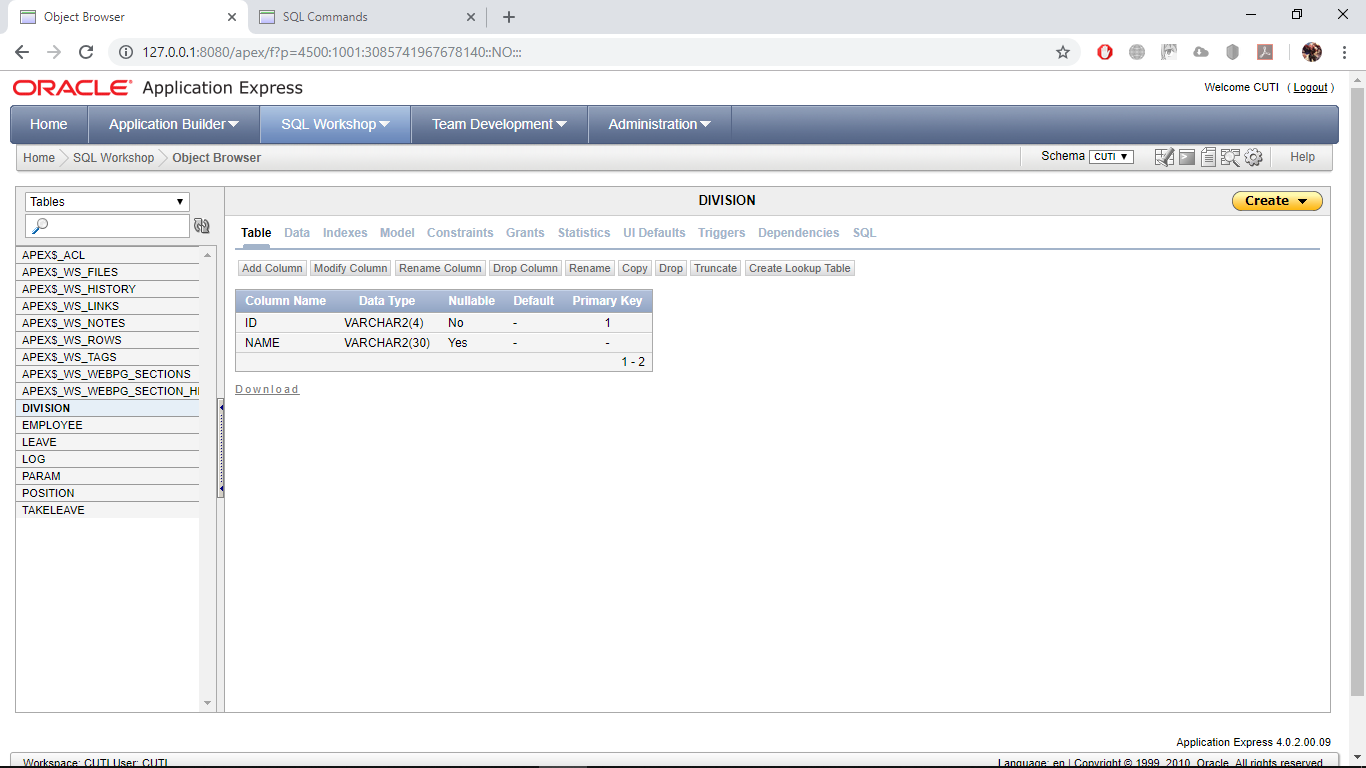


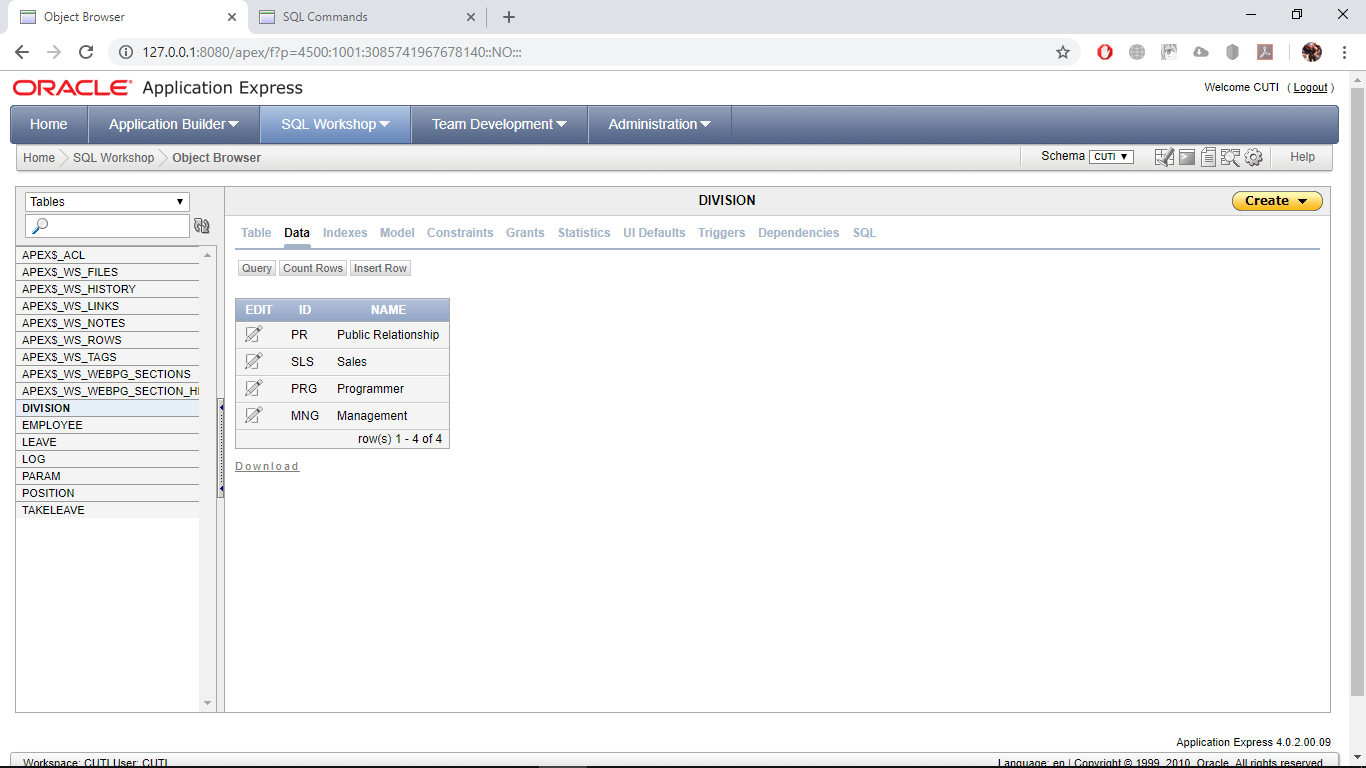
Pada class diagram aplikasi request cuti, terdapat beberapa class yaitu:

* 1. class takeleave yang memiliki atribut berupa id, deskripsi, tanggal mulai, tanggal berakhir, status approval, id employee, dan id leave. Class takeleave memiliki method berupa insert, update delete, view, compare leave, check holiday, dan update quota.
  2. Class employee yang memiliki atribut berupa id, nama, gender, status nikah, jumlah anak, username, password,this\_year,last\_year, id divisi, dan id posisi. Class employee memiliki method berupa insert, update delete dan view.
  3. Class position yang memiliki atribut berupa id dan nama. Class position memiliki method berupa insert, update delete dan view.
  4. Class division yang memiliki atribut berupa id dan nama. Class position memiliki method berupa insert, update delete dan view.
  5. Class log yang memiliki atribut berupa tanggal login dan id employee. Class log memiliki method berupa insert, delete, dan view.
  6. Class leave yang memiliki atribut berupa id, nama, status, dan jumlah hari. Class leave memiliki method berupa insert, delete, dan view.

**DATABASE & PL SQL**

1. Database
2. Division





Procedure pada divisi ada 3:

* Procedure hapus

create or replace procedure deleteDivision(ids in varchar2)  
IS  
begin  
delete from division where id = ids;  
end;​

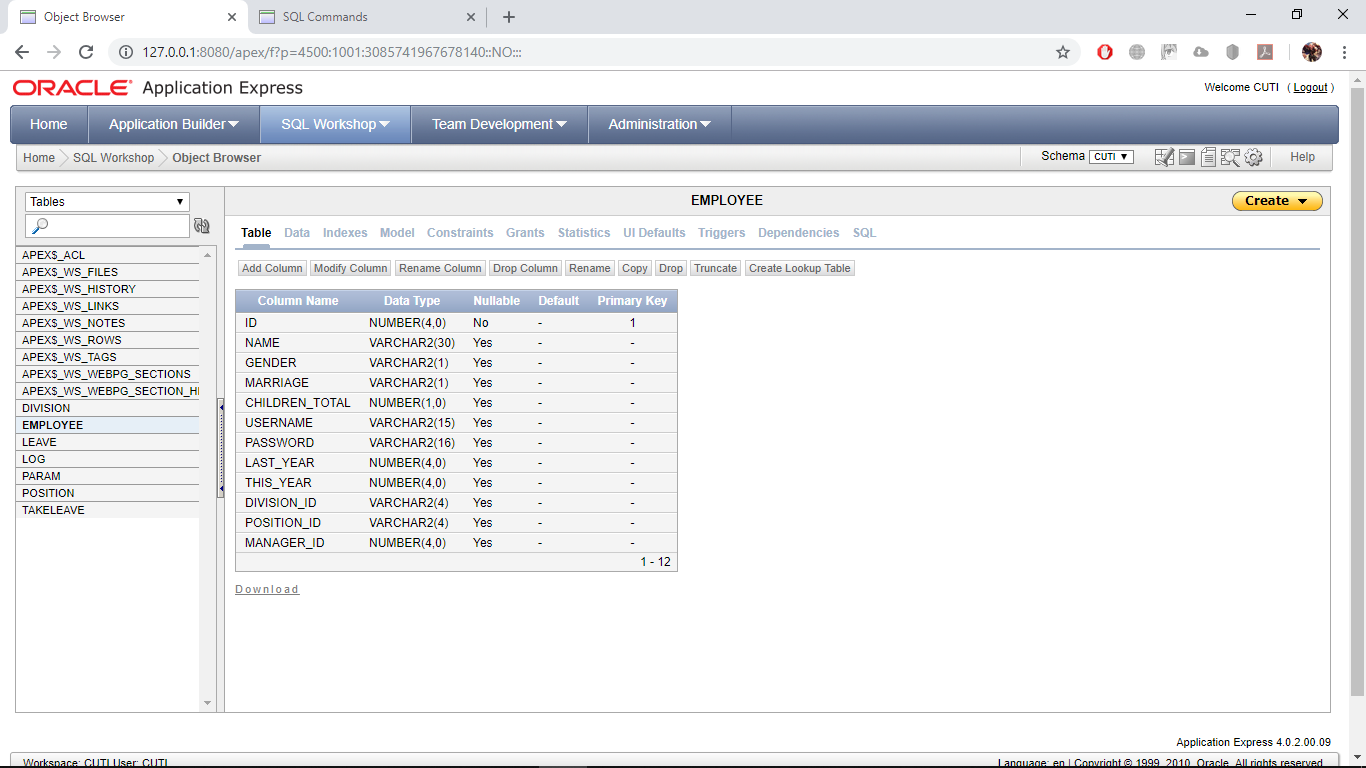
* Procedure Insert

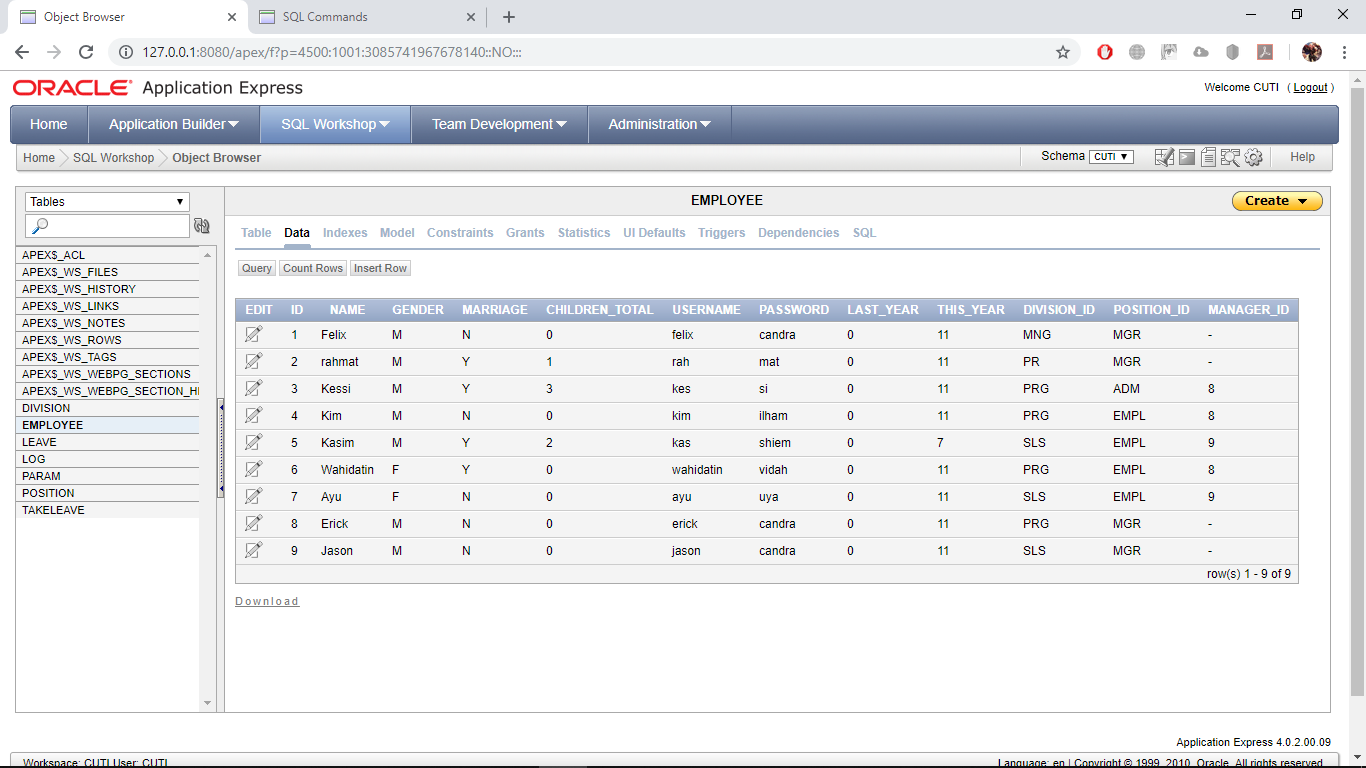
create or replace procedure insertDivision(ids in varchar2, names in varchar2)  
IS  
begin  
insert into division values(ids, names);  
end;​

* Procedure Ubah

create or replace procedure updateDivision(ids in varchar2, names in varchar2)  
IS  
begin  
update division set name = names where id = ids;  
end;

1. Employee





Procedure dan Trigger

* Procedure Insert

create or replace procedure insertEmployee(id in number, name in varchar2, gender in varchar2, marriage in varchar2, children\_total in number, username in varchar2, password in varchar2, last\_year in number, this\_year in number, division\_id in varchar2, position\_id in varchar2, manager\_id in number)  
is  
begin  
insert into employee values(id, name, gender, marriage, children\_total, username, password, last\_year, this\_year, division\_id, position\_id, manager\_id);  
end;​

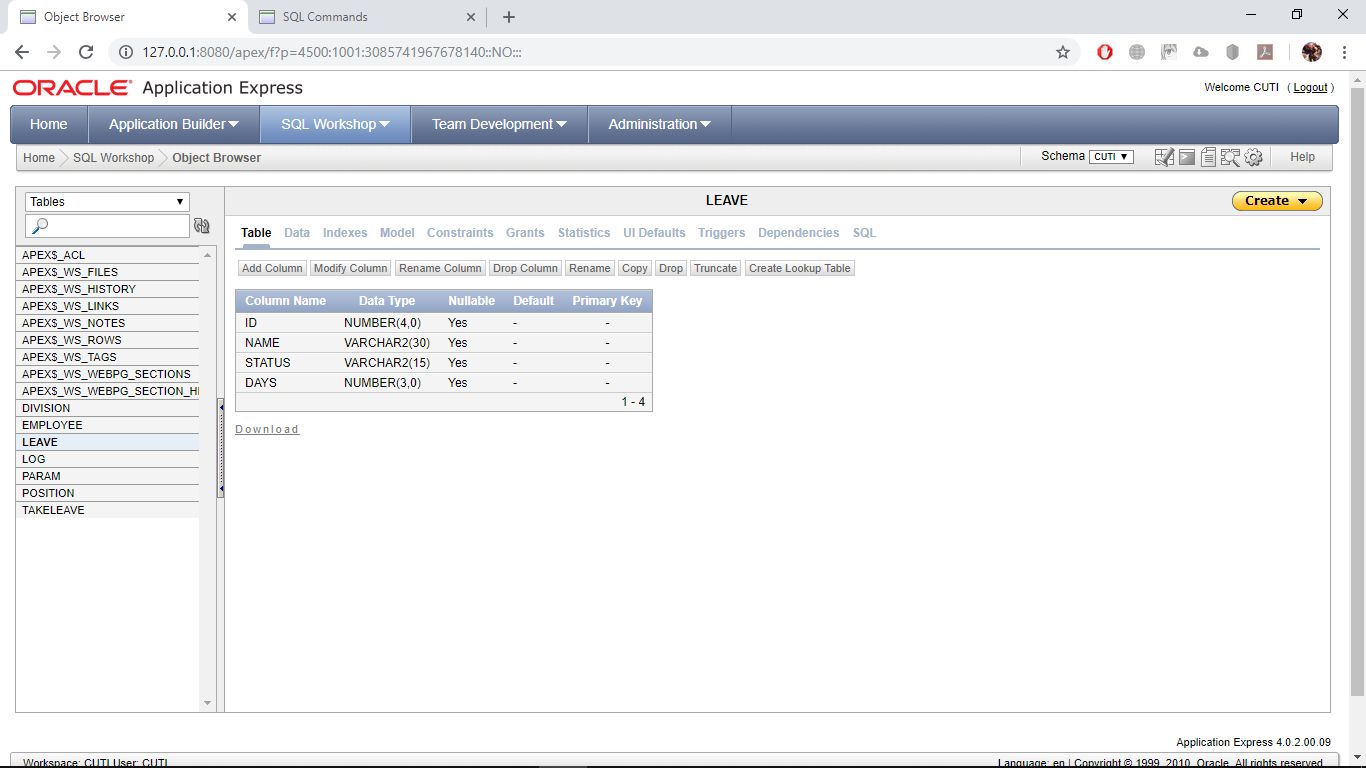
* Procedure Hapus

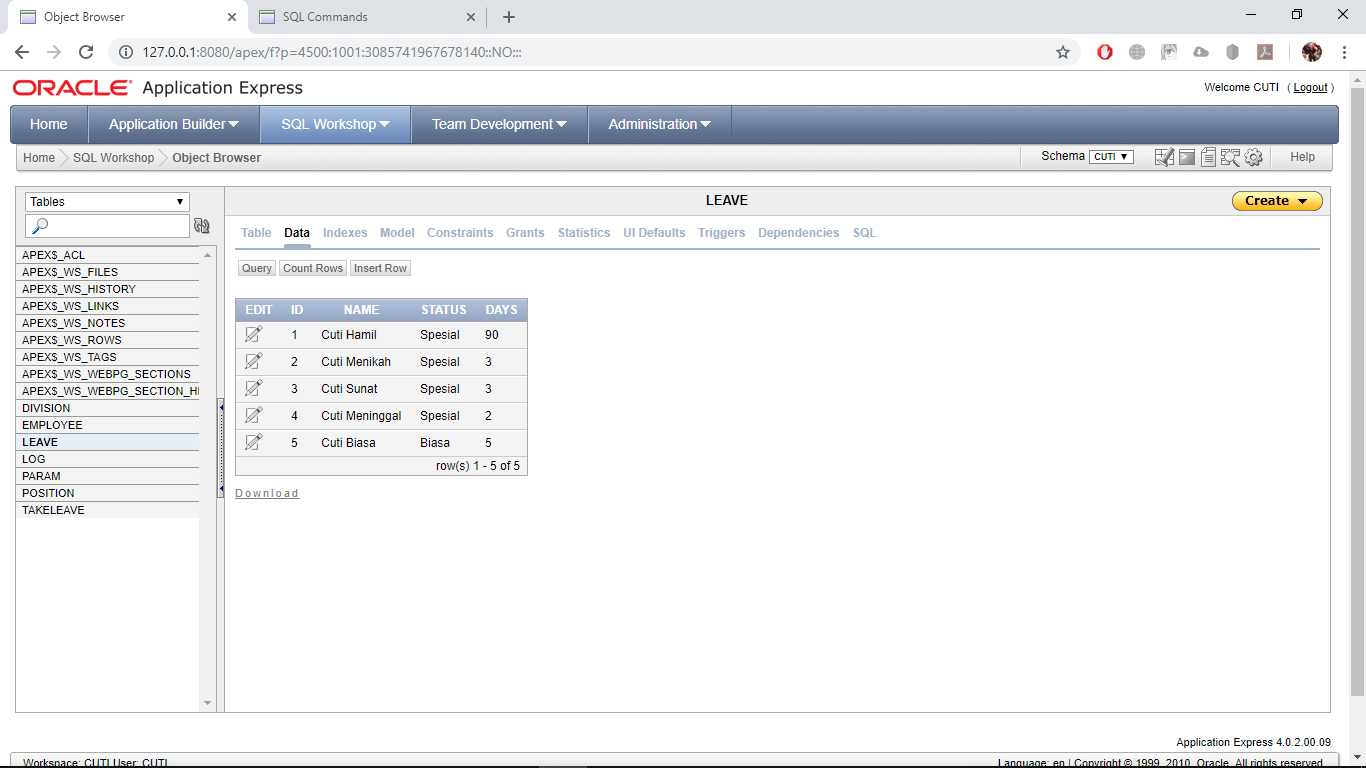
create or replace procedure deleteEmployee(ids in number)  
is  
begin  
delete from employee where id = ids;  
end;​

* Procedure Ubah

create or replace procedure updateEmployee(ids in number, names in varchar2, genders in varchar2, marriages in varchar2, children\_totals in number, usernames in varchar2, passwords in varchar2, last\_years in number, this\_years in number, division\_ids in varchar2, position\_ids in varchar2, manager\_ids in number)  
is  
begin  
update employee set name = names, gender = genders, marriage = marriages, children\_total = children\_totals, username = usernames, password = passwords, last\_year = last\_years, this\_year = this\_years, division\_id = division\_ids, position\_id = position\_ids, manager\_id = manager\_ids where id = ids;  
end;​

1. Leave





Procedure :

* Insert

create or replace procedure insertLeave(id in number, name in varchar2, status in varchar, days in number)

is

begin

insert into leave values(id, name, status, days);

end;​

* Update

create or replace procedure updateLeave(ids in number, names in varchar2, statuss in varchar, dayss in number)

is

begin

update leave set name = names, status = statuss, days = dayss where id = ids;

end;​

* Delete

create or replace procedure deleteLeave(ids in number)

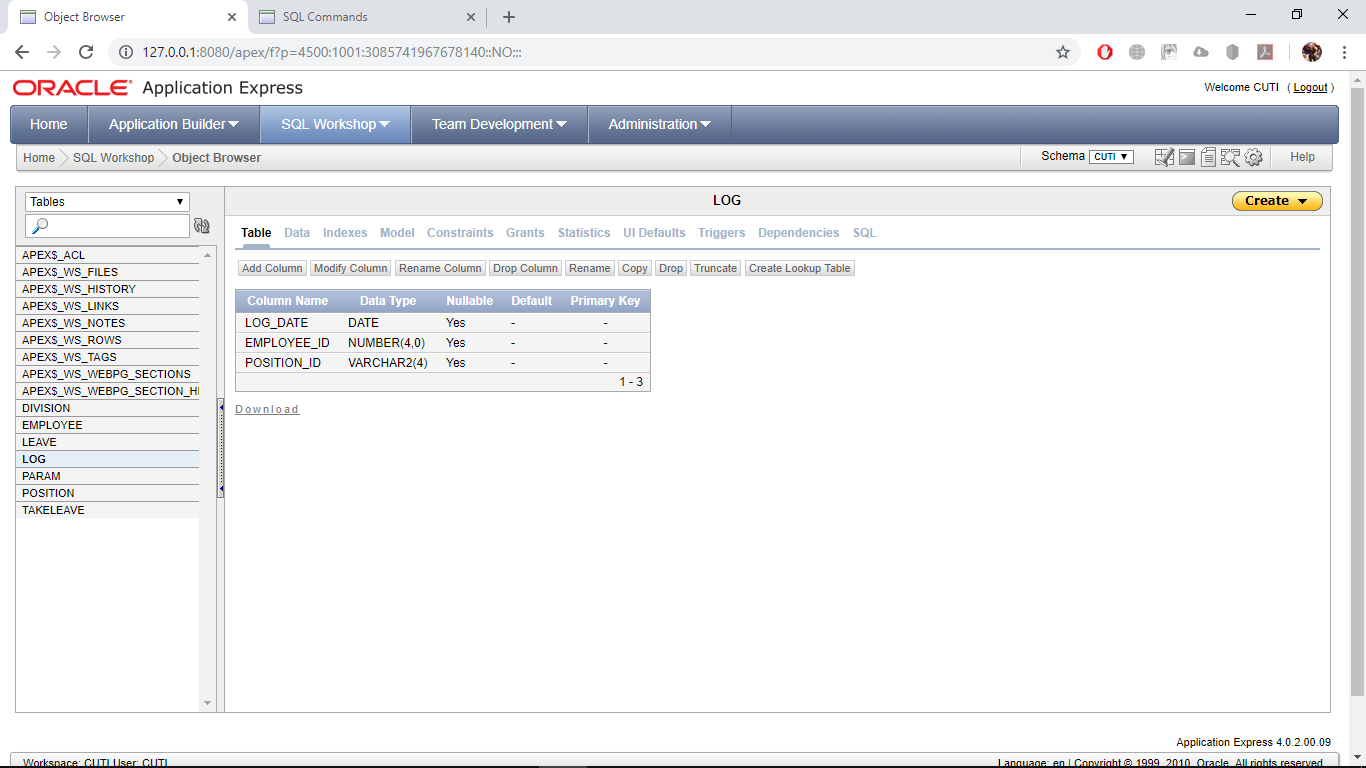
is

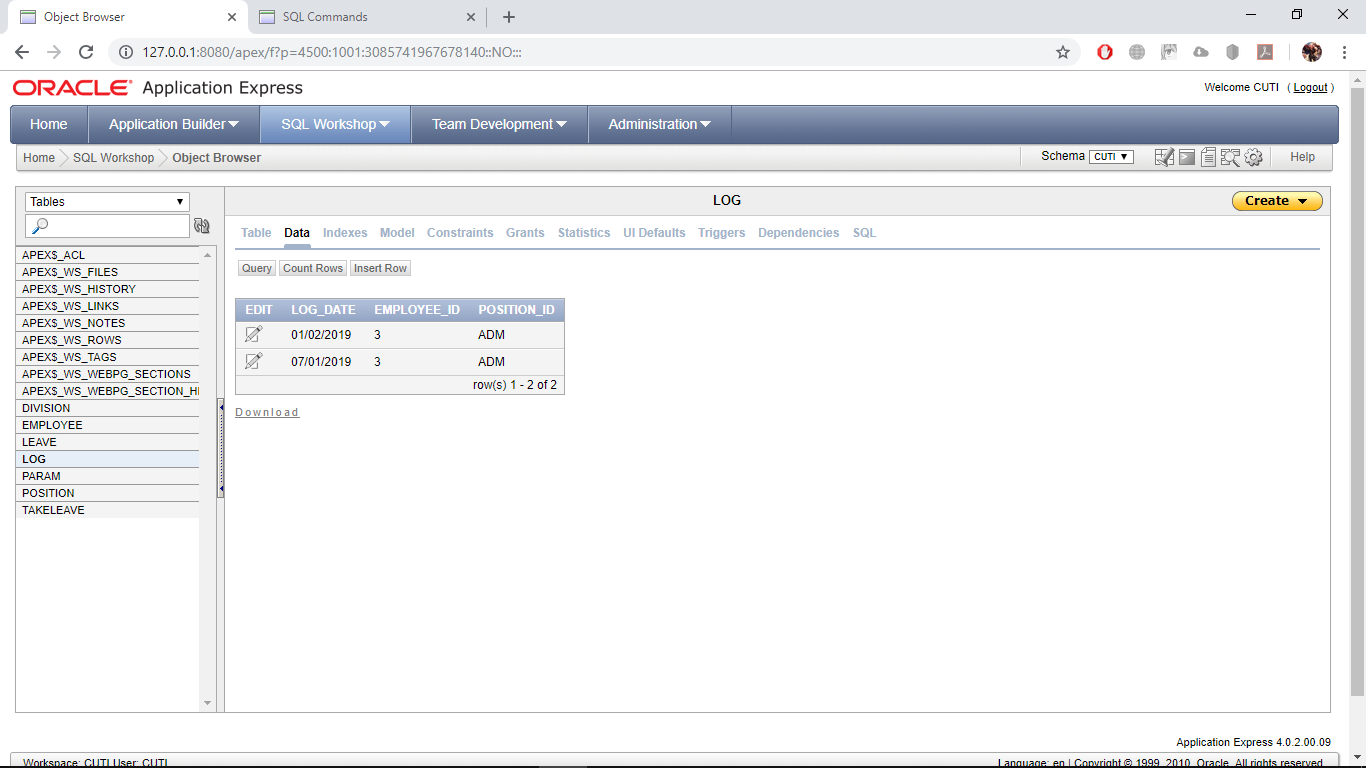
begin

delete from leave where id = ids;

end;​

1. Log





Procedure :

* insert

create or replace procedure insertLog(log\_date in date,employee\_id in number,position\_id in varchar2)

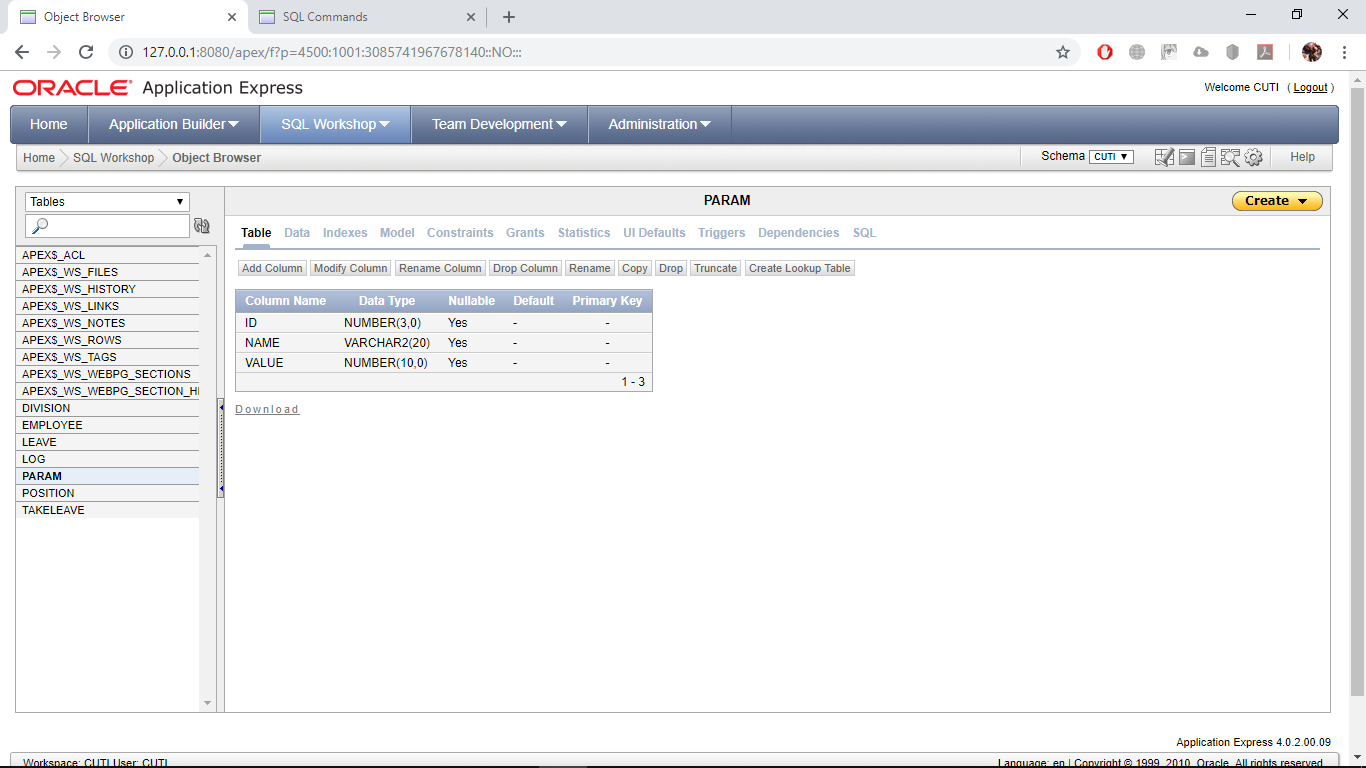
is

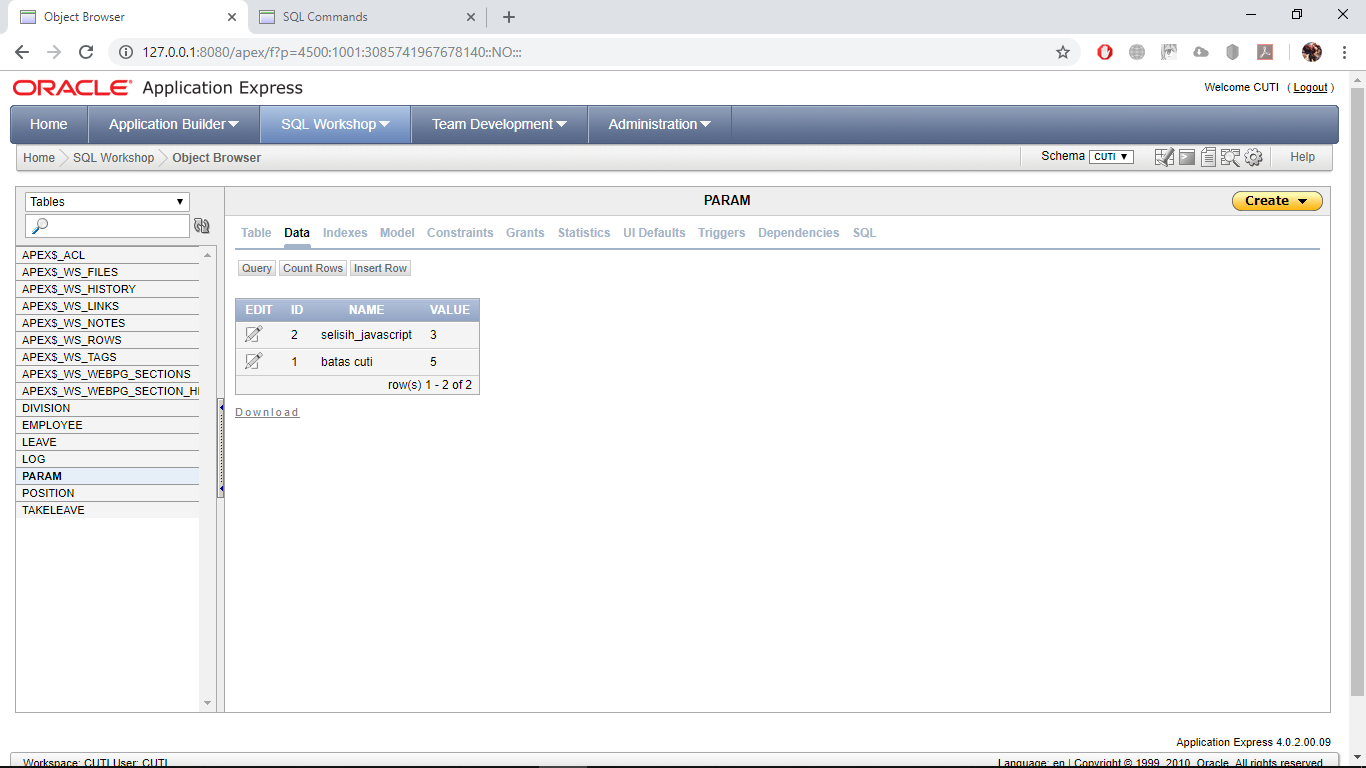
begin

insert into log values(to\_date(log\_date,'DD/MM/YYYY'),employee\_id,position\_id);

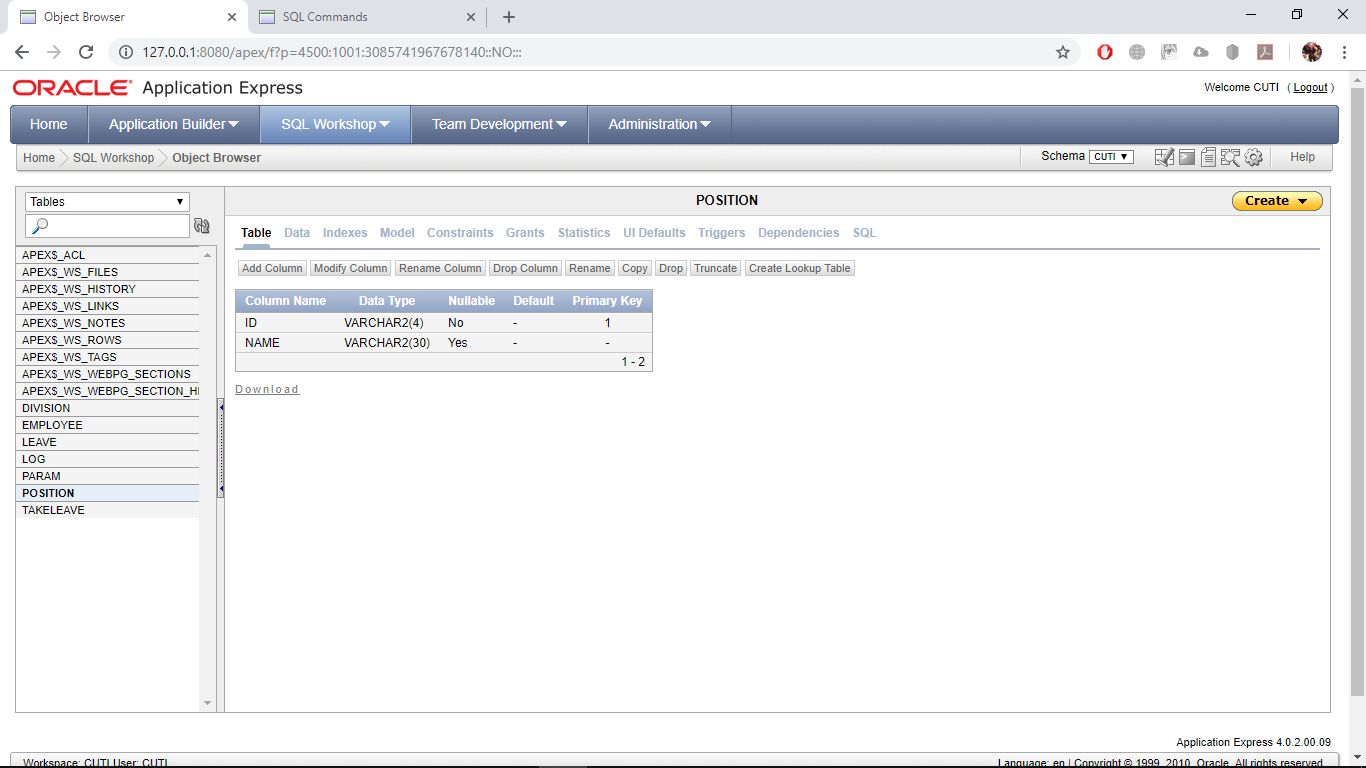
end;​

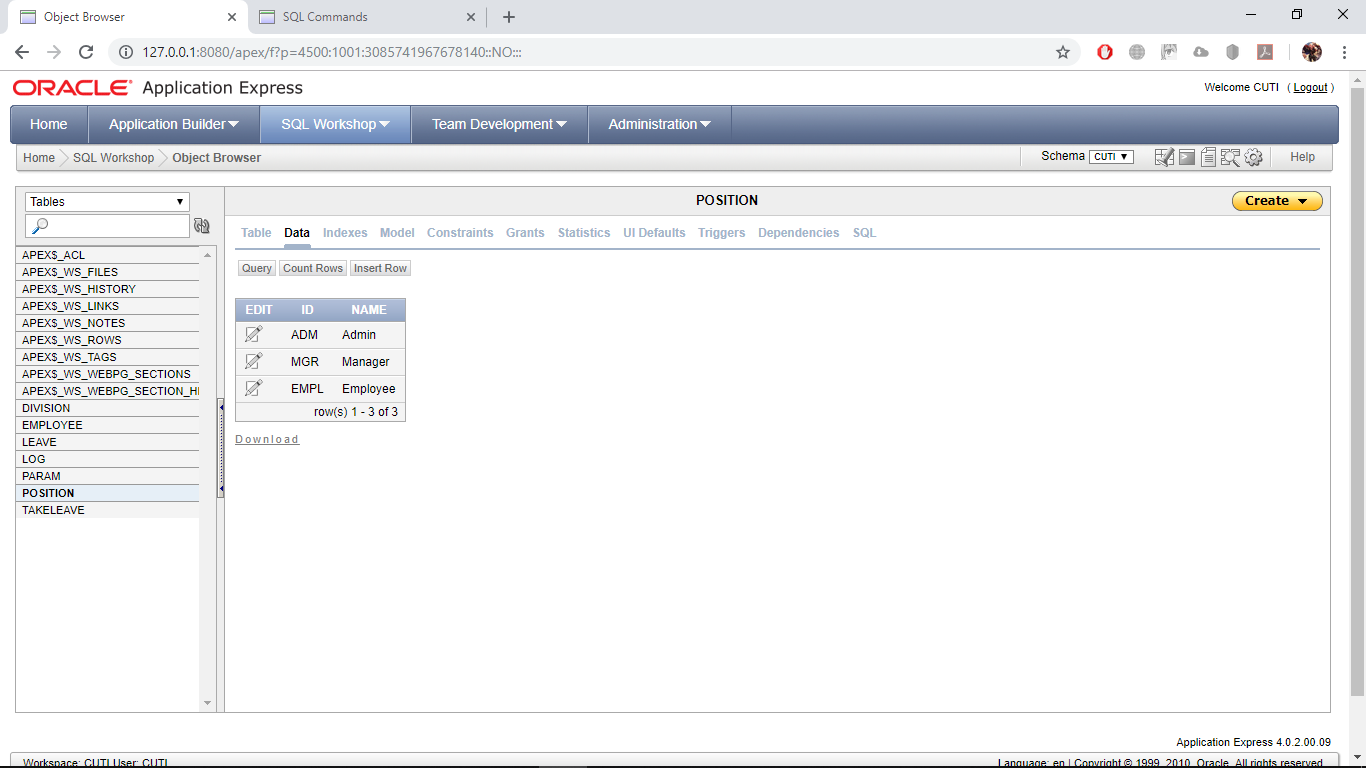
1. Param





1. Position





Procedure :

* insert

create or replace procedure insertPosition(id in varchar2, name in varchar2)

is

begin

insert into position values(id, name);

end;​

* update

create or replace procedure updatePosition(ids in varchar2, names in varchar2)

is

begin

update position set name = names where id = ids;

end;​

* delete

create or replace procedure deletePosition(ids in varchar2)

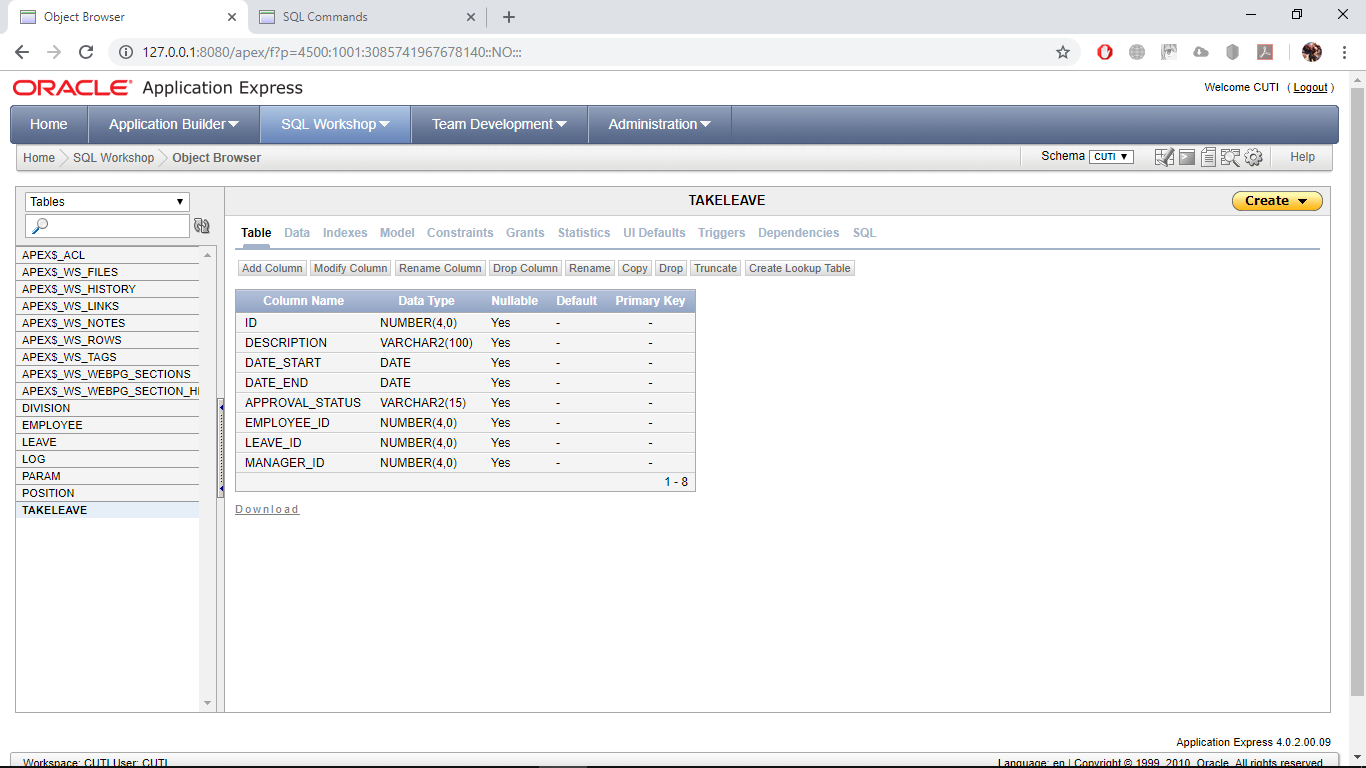
is

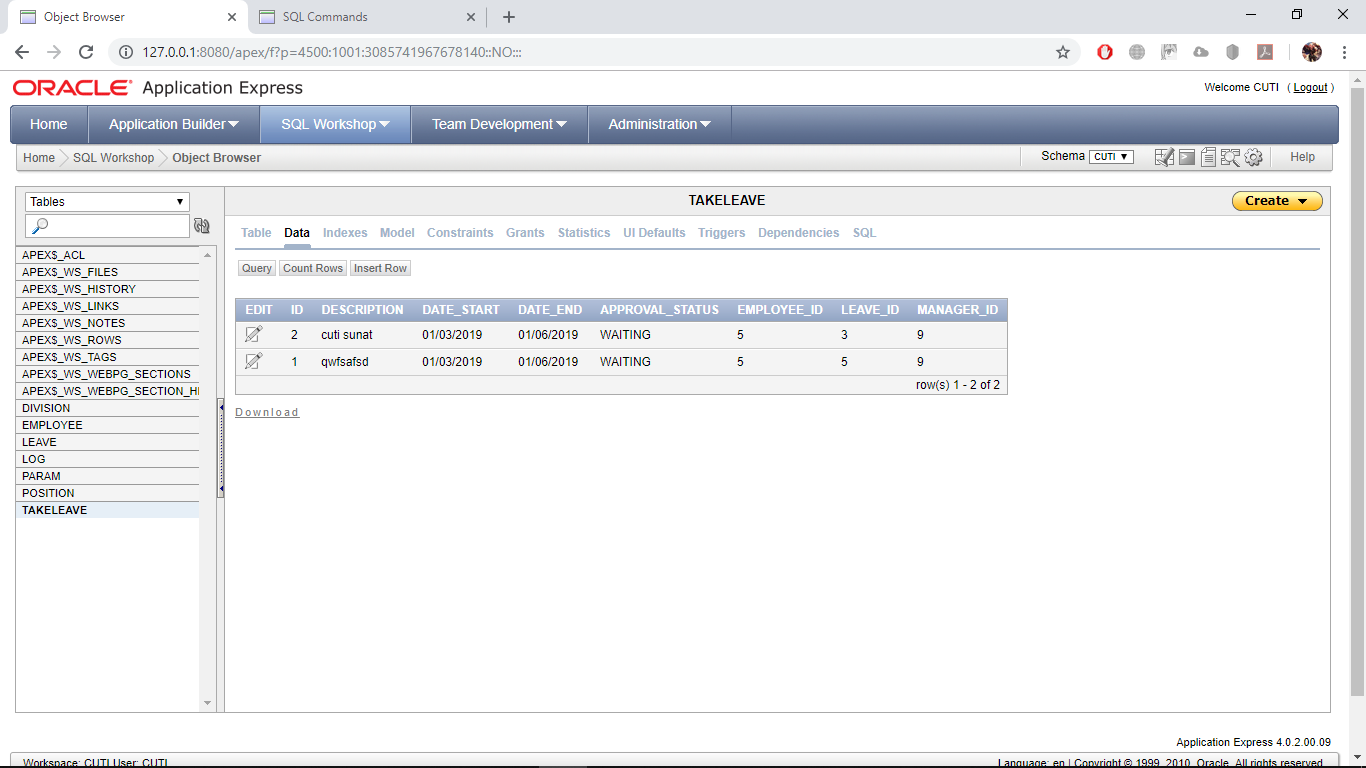
begin

delete from position where id = ids;

end;​

1. Take Leave





Procedure:

* insert

create or replace procedure insertTakeLeave(id in varchar2, description in varchar2, date\_start in date, date\_end in date, approval\_status in varchar2, employee\_id in number, leave\_id in number, manager\_id in number)

is

begin

insert into takeleave values(id, description, to\_date(date\_start,'DD/MM/YYYY'), to\_date(date\_end,'DD/MM/YYYY'), approval\_status, employee\_id, leave\_id, manager\_id);

end;​

* update

create or replace procedure updateTakeLeave(ids in varchar2, descriptions in varchar2, date\_starts in date, date\_ends in date, approval\_statuss in varchar2, employee\_ids in number, leave\_ids in number, manager\_ids in number)

is

begin

update takeleave set description = descriptions, date\_start = to\_date(date\_starts,'DD/MM/YYYY'), date\_end = to\_date(date\_ends,'DD/MM/YYYY'), approval\_status = approval\_statuss, employee\_id = employee\_ids, leave\_id = leave\_ids, manager\_id = manager\_ids where id = ids;

end;​

* delete

create or replace procedure deleteTakeLeave(ids in varchar2)

is

begin

delete from takeleave where id = ids;

end;​

1. Trigger

* Trigger Cancel Leave

create or replace trigger cancel\_leave  
after update on takeleave  
for each row  
when(new.approval\_status='CANCELED')  
declare  
selisih number;  
eid number;  
status2 leave.status%type;  
begin  
select status into status2 from leave where id = :new.leave\_id;  
if(status2 = 'Biasa') then  
select value into selisih from param where id = 2;  
eid := :new.employee\_id;  
update employee set this\_year=this\_year+selisih where id = eid;  
end if;  
end;​

* Trigger reset quota this year

create or replace trigger reset\_quota\_this\_year  
after insert on log   
for each row   
when(new.log\_date=to\_date('02/01','DD/MM') and new.position\_id='ADM')  
begin  
update employee set this\_year=12;  
end;​

* Trigger reset Quota Last Year

create or replace trigger reset\_quota\_last\_year

after insert on log

for each row

when(new.log\_date=to\_date('01/07','DD/MM') and new.position\_id='ADM')

begin

update employee set last\_year=0;

end;​

* Rejected

create or replace trigger rejected

after update on takeleave

for each row

when(new.approval\_status='REJECTED')

declare

selisih number;

eid number;

status2 leave.status%type;

begin

if(status2='Biasa') then

select status into status2 from leave where id = :new.leave\_id;

select value into selisih from param where id = 2;

eid := :new.employee\_id;

update employee set this\_year=this\_year+selisih where id=eid;

end if;

end;​

* on insert (kurangi kuota)

create or replace trigger subtract\_quota

after insert on takeleave

for each row

declare

selisih number;

eid number;

status2 leave.status%type;

begin

select status into status2 from leave where id = :new.leave\_id;

if(status2 = 'Biasa') then

select value into selisih from param where id = 2;

eid := :new.employee\_id;

update employee set this\_year=this\_year-selisih where id=eid;

end if;

end;​

* reschedule

create or replace trigger reschedule

after update on takeleave

for each row

declare

selisih number;

selisih\_lama number;

eid number;

status2 leave.status%type;

begin

select status into status2 from leave where id = :new.leave\_id;

if(status2 = 'Biasa') then

select value into selisih from param where id = 2;

eid := :new.employee\_id;

update employee set this\_year=this\_year+selisih\_lama where id=eid;

select value into selisih from param where id = 2;

update employee set this\_year=this\_year-selisih where id=eid;

end if;end;​

1. Procedure View

* Getdivision (data divisi untuk admin)

create or replace procedure getDivision

IS

cursor d\_data is select \* from division;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('id : '||d\_rec.id);

dbms\_output.put\_line('name : '||d\_rec.name);

end loop;

end;

* getemployee (data employee untuk admin)

create or replace procedure getEmployee

IS

cursor d\_data is select \*

from employee;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('ID : '||d\_rec.id);

dbms\_output.put\_line('Name : '||d\_rec.name);

dbms\_output.put\_line('Gender : '||d\_rec.gender);

dbms\_output.put\_line('Marriage : '||d\_rec.marriage);

dbms\_output.put\_line('Child : '||d\_rec.children\_total);

dbms\_output.put\_line('Username : '||d\_rec.username);

dbms\_output.put\_line('Password : '||d\_rec.password);

dbms\_output.put\_line('Last Year Quota : '||d\_rec.last\_year);

dbms\_output.put\_line('This Year Quota : '||d\_rec.this\_year);

dbms\_output.put\_line('Division : '||d\_rec.division\_id);

dbms\_output.put\_line('Position : '||d\_rec.position\_id);

dbms\_output.put\_line('Religion : '||d\_rec.religion);

end loop;

end;

* getleave (data kategori cuti untuk admin)

create or replace procedure getLeave

IS

cursor d\_data is select \* from leave;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('id : '||d\_rec.id);

dbms\_output.put\_line('name : '||d\_rec.name);

dbms\_output.put\_line('status : '||d\_rec.status);

dbms\_output.put\_line('days : '||d\_rec.days);

end loop;

end;

* getleaveemp (data kategori cuti untuk emloyee)

create or replace procedure getLeaveEmp(ids in number)

IS

cursor d\_data is select \* from employee where id=ids;

cursor consatu is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Menikah';

cursor condua is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Menikah' and name != 'Cuti Umroh' and name != 'Cuti Haji';

cursor contiga is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Melahirkan';

cursor conempat is select name from leave where name != 'Cuti Sunat' and name != 'Cuti Melahirkan' and name != 'Cuti Umroh' and name != 'Cuti Haji';

cursor conlima is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Menikah';

cursor conenam is select name from leave where name != 'Cuti Menikah' and name != 'Cuti Melahirkan' and name != 'Cuti Umroh' and name != 'Cuti Haji';

cursor contuju is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Sunat';

cursor kelapa is select name from leave where name != 'Cuti Melahirkan' and name != 'Cuti Sunat' and name != 'Cuti Umroh' and name != 'Cuti Haji';

consatu\_rec consatu%rowtype;

condua\_rec condua%rowtype;

contiga\_rec contiga%rowtype;

conempat\_rec conempat%rowtype;

conlima\_rec conlima%rowtype;

conenam\_rec conenam%rowtype;

contuju\_rec contuju%rowtype;

kelapa\_rec kelapa%rowtype;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

if(d\_rec.gender = 'F' and d\_rec.marriage = 'Y' and d\_rec.religion = 'Islam') then

for consatu\_rec in consatu loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||consatu\_rec.name);

end loop;

elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'Y' and d\_rec.religion != 'Islam') then

for condua\_rec in condua loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||condua\_rec.name);

end loop;

elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'N' and d\_rec.religion = 'Islam') then

for contiga\_rec in contiga loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||contiga\_rec.name);

end loop;

elsif(d\_rec.gender = 'F' and d\_rec.marriage = 'N' and d\_rec.religion != 'Islam') then

for conempat\_rec in conempat loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||conempat\_rec.name);

end loop;

elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'Y' and d\_rec.religion = 'Islam' and d\_rec.children\_total>0) then

for conlima\_rec in conlima loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||conlima\_rec.name);

end loop;

elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'Y' and d\_rec.religion != 'Islam' and d\_rec.children\_total>0) then

for conenam\_rec in conenam loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||conenam\_rec.name);

end loop;

elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'N' and d\_rec.religion = 'Islam') then

for contuju\_rec in contuju loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||contuju\_rec.name);

end loop;

elsif(d\_rec.gender = 'M' and d\_rec.marriage = 'N' and d\_rec.religion != 'Islam') then

for kelapa\_rec in kelapa loop

dbms\_output.put\_line('-----------');

dbms\_output.put\_line('Nama Cuti: '||kelapa\_rec.name);

end loop;

else

dbms\_output.put\_line('apalah');

end if;

end loop;

end;

* getposition (data posisi untuk admin)

create or replace procedure getPosition

IS

cursor d\_data is select \* from position;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('id : '||d\_rec.id);

dbms\_output.put\_line('name : '||d\_rec.name);

end loop;

end;

* gettakeleave (data cuti untuk manager)

create or replace procedure getTakeLeave

IS

cursor d\_data is select e.id,e.name,l.name as cuti,t.description,t.date\_start,t.date\_end,t.approval\_status

from takeleave t join employee e on t.employee\_id=e.id join leave l on l.id=t.leave\_id

where t.approval\_status='WAITING';

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('id : '||d\_rec.id);

dbms\_output.put\_line('name : '||d\_rec.name);

dbms\_output.put\_line('cuti : '||d\_rec.cuti);

dbms\_output.put\_line('description : '||d\_rec.description);

dbms\_output.put\_line('date\_start : '||d\_rec.date\_start);

dbms\_output.put\_line('date\_end : '||d\_rec.date\_end);

dbms\_output.put\_line('date\_end : '||d\_rec.approval\_status);

end loop;

end;

* gettakeleaveemp (data cuti untuk employee)

create or replace procedure getTakeLeaveEmp(ids in number)

IS

cursor d\_data is select l.name as cuti,t.description,t.date\_start,t.date\_end,t.approval\_status

from takeleave t join employee e on t.employee\_id=e.id join leave l on l.id=t.leave\_id

where e.id=ids;

d\_rec d\_data%rowtype;

begin

for d\_rec in d\_data loop

dbms\_output.put\_line('----------');

dbms\_output.put\_line('cuti : '||d\_rec.cuti);

dbms\_output.put\_line('description : '||d\_rec.description);

dbms\_output.put\_line('date\_start : '||d\_rec.date\_start);

dbms\_output.put\_line('date\_end : '||d\_rec.date\_end);

dbms\_output.put\_line('date\_end : '||d\_rec.approval\_status);

end loop;

end;