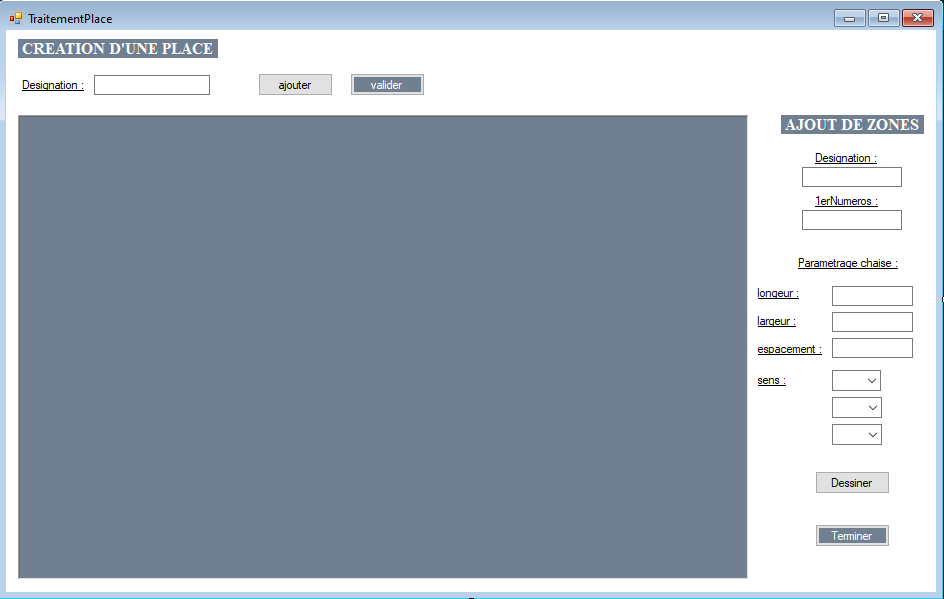
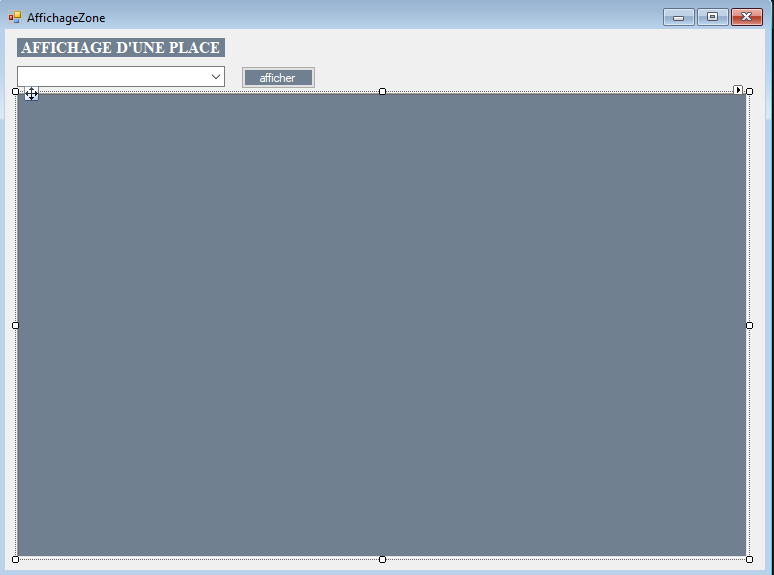
RESERVATION

# Affichage :

1. Création d’une place et ses zones



1. Affichage des zones



# METIER :

Classe Generale : créer la fonction public static string PointsToString(Points[] points) et public static Points[] StringToPoint(string points)

Classe Emplacement : string idEmplacement,string designation

Public void createZone(string designation,Point[] points,int preNumero,int hauteurPlace,int largPlace, int espacementHor,int espacementVer,string sens)

Créer la zone avec les parametres et les ids avec General.getidsequence

Appel de la fonction createPlacesDefaut(hauteurPlace,largPlace) de la zone avec les infos ainsi obtenue pour recuperer les places

Inserer la zone dans la base avec General.insert

Classe Zone : string idZone ,string idEmplacement,string designation,string points,int preNumero,int espacementhor,int espacementver,string sens

Public place[] createPlacesDefaut(int hauteurPlace,int largPlace)

Récupérer le tableau des points de la zone avec General.StringToPoints(this.points)

Créer un graphicsPath gp contenant les points de la zone

Créer un rectangle rect avec la fonction GetBounds du graphicsPath

Creation de la classe sens avec this.sens

J’appelle la fonction remplirChaisesVerticale ou horizontale selon les cas le point d’origine du placement des chaises selon 4 cas :

GestionComparaison gestX,gestY ;

Haut-gauche : gestX(0,rect.widht,largeurPlace+this.espacementhor)

gestY(0,rect.height,hauteurPlace+this.espacementver)

Haut-droite : gestX(rect.widht,0,largeurPlace+this.espacementhor)

gestY(0, rect.height,hauteurPlace+this.espacementver)

Bas-gauche : gestX(0,rect.widht,largeurPlace+this.espacementhor)

gestY(rect.height,0,hauteurPlace+this.espacementver)

Bas-droit : gestX(rect.widht,0,largeurPlace+this.espacementhor)

gestY(rect.height,0,hauteurPlace+this.espacementver)

Place[] rep=remplirChaiseVerticale(gestX,gestY,hauteurPlace,largeurPlace,gp)

Return rep ;

haut-droit,bas-gauche,bas-droite

Ensuite appeler la fonction remplirChaisesVerticale ou remplirChaisesHorizontale selon le troisième sens choisis

Public Place[] remplirChaisesVerticale(GestionComparaison gestX,GestionComparaison gestY,hauteurPlace,largeurPlace,GraphicsPath zone){

Int num=this.preNum ;

Liste de place ;

While(gestX.notFinished()){

While(gestY.notFinished){

Place place=new Place(hauteurPlace,largeurPlace,gestX. nbActu,gestY. nbActu)

Si place.ifIn(zone);

Place=new Place(this.idZone,num,hauteurPlace,largeurPlace, gestX. nbActu,gestY. nbActu

Num++;

Ajouter la place dans la liste

gestY.move()

{

getY.reinitialiser();

gestX.Move();

}

}

Classe Place : string idPlace,string idZone,int numero ,int hauteurPlace,int largPlace,int x,int y

Public Point[] getPoints(){

Point[0]=(this.x,this.y)

Point[1]=(this.x+this.LargPlace,this.y)

Point[2]=(this.x+this.LargPlace,this.y+hauteurPlace)

Point[3]=(this.x,this.y+hauteurPlace)

}

Public bool ifIn(GraphicsPath gp){

Recuperer les points

Rep=true

For(int i=0;i<points.Length;i++){

Si gp.isVisible(points[i])==false , rep=false et break ;

}

}

Classe concert : string idConcert,string idEmplacement,string designation,dateTime dateAnnonce,dateTime dateRepresentation

Classe simulation: string idSimulation,string idConcert

Classe infoSimulation : string idZone,decimal porcentageRes

Classe Ensemble (herite de Emplacement):Zone[] zone,Place[] places

Classe GestionComparaison(int nbInitial,int nbFinal,int mouvement,string type,int nbActu)

Constructeur :( int nbInitial,int nbFinal,int mouvement)

Si initial<final type=asc sinon desc

Public void Move()

If(type==asc){

nbActu +=mouvement ;

}else{

nbActu-=mouvement ;

Return true ;

}

Public bool notFinished()

If(type==asc){

If(nbActu >final){

Return false;

}

}else{

If(nbActu <final){

Return false;

}

}

Return true ;

Public void reinitialiser(){

This.nbActu=this.nbInitial ;

}

Classe Sens : string cote,string posVerticale,string sensHorVer

Constructeur:(string sens)

Split le sens par ‘–‘ pour recuperer les trois attribut

# BASE :

## Tables :

Emplacement(idEmplement,designation)

Zone(idZone,idEmplacement,designation,points,preNumero,espacementhor,espacementver,sens)

Place(idPlace, idZone, numero , hauteurPlace, largPlace, x, y)