**UNIVERSITE CADI AYYAD**

**ECOLE SUPERIEURE DE**

**TECHNOLOGIE SAFI**

Département : technique de management.

Filière : Génie informatique.

Option : Administration des systèmes réseaux.

**Rapport de**

**Projet de fin d’étude**

**Conception et réalisation d’une plateforme de loi optique en ligne avec Java**

* **Travail Elaboré par** : NACIRI TAOUFIK MOHAMMED

TOUAILAB ILYASE

ZAAZAA OUSSAMA

* **Encadré par**  : M. BAKKAS JAMAL
* **Année universitaire** : 2016/2017

# 

# Remerciements

Au terme de ce travail, nous saisissons cette occasion pour exprimer nos vifs remerciements à toute l’équipe Pédagogique de l’école supérieure de technologie de SAFI et les Intervenants professionnels responsables de la formation génie informatique et toute personne ayant contribué, de près ou de loin, à la réalisation de ce travail. Nous exprimons l’expression de notre haute reconnaissance à notre encadrant le professeur BAKAS Jamal, qui nous a encadrés avec patience durant la réalisation de ce travail de fin d'études et qui n’a épargné aucun effort pour mettre à notre disposition la documentation et les explications nécessaires, ses conseils nous ont été bien utiles, notamment pour la rédaction de ce rapport. Nous exprimons également notre gratitude aux membres du jury, qui nous ont honorés en acceptant de juger ce modeste travail. Enfin nous tenons à remercier l’ensemble du corps enseignant de la Filière génie informatique à l’ESTS.

MERCI

Liste des figures

[Figure 1 : organigramme et schéma de navigation 13](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994855)

[Figure 2 : Représentation d’un système optique 15](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994856)

[Figure 3 : différents types de lentilles et leurs symboles 16](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994857)

[Figure 4 : lentille divergente : objet virtuelle, image réel 16](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994858)

[Figure 5 : lentille divergente : objet réel, image virtuelle 16](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994859)

[Figure 6 : lentille convergente : objet virtuelle, image réel 16](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994860)

[Figure 7 : lentille convergente : objet réel, image réel 16](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994861)

[Figure 8: lentille convergente : objet réel, image virtuelle 17](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994862)

[Figure 9 : construction de l’image d’un objet réel par un miroir concave 17](#_Toc483994864)

[Figure 10 : miroir convexe 18](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994865)

[Figure 11 : scene builder 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994866)

[Figure 12 : quelque bouton fourni par javafx 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 13 :Changer la couleur des lignes 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 14 :Fichier XML qui récupère l’état 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 15 : Page index 23](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994863)

[Figure 16 : Zone pour saisir les informations d’accés au site 23](#_Toc483994864)

[Figure 17: page d’acceuil du professeur 24](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994865)

[Figure 18 : page de gestion des cours 24](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994866)

[Figure 19 : page de gestion des étudiants 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 20 : tableau contenant la liste des étudiants **Erreur ! Signet non défini.**](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994863)

[Figure 21 : page d’acceuil 26](#_Toc483994864)

[Figure 22 : page des cours 18](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994865)

[Figure 23 :page conduisant vers la simulation 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994866)

[Figure 24 :simulation en java 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 25 :Chois du type de simulation 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

[Figure 26 :Chois du type de la taille 21](file:///F:\p\UNIVERSITE%20CADI%20AYYAD.docx#_Toc483994867)

Sommaire

[Remerciements 2](#_Toc483997586)

[Liste des figures 3](#_Toc483997587)

[Introduction 6](#_Toc483997588)

[**Chapitre 1 : Présentation du projet** 7](#_Toc483997589)

[1.Présentation 8](#_Toc483997590)

[2.L’objectif 8](#_Toc483997591)

[3.Synthèse 8](#_Toc483997592)

[a. Point à éviter 8](#_Toc483997593)

[b. Point à retenir 8](#_Toc483997594)

[4.Cahier des charges 9](#_Toc483997595)

[a. Contraintes esthétiques 9](#_Toc483997596)

[b. Contraintes ergonomiques 9](#_Toc483997597)

[c. Contraintes fonctionnelles 10](#_Toc483997598)

[5.La simulation 10](#_Toc483997599)

[6.Conclusion 10](#_Toc483997600)

[**Chapitre 2 : Conception** 11](#_Toc483997601)

[1.Introduction : 12](#_Toc483997602)

[2.Conception du Site Web : 12](#_Toc483997603)

[a. Synopsis du Site web 12](#_Toc483997604)

[b. Etude des cas d’utilisation 14](#_Toc483997605)

[i. Identification des acteurs 14](#_Toc483997606)

[ii. Identification des cas d'utilisation 14](#_Toc483997607)

[3.Conception de l’application 15](#_Toc483997608)

[a. L’objectif 15](#_Toc483997609)

[b. Les lois optiques 15](#_Toc483997610)

[4.conclusion 18](#_Toc483997611)

[**Chapitre 3 : Réalisation** 19](#_Toc483997612)

[1.Introduction : 20](#_Toc483997613)

[2.L’environnement de développement : 20](#_Toc483997614)

[a. Environnement Logiciel : 20](#_Toc483997615)

[3.Langage untilisée : 20](#_Toc483997616)

[a.HTML/CSS Le HTML 20](#_Toc483997617)

[b. JSP 20](#_Toc483997618)

[c.Java fx 20](#_Toc483997619)

[d.JDom 20](#_Toc483997619)

[4.Les principales interfaces graphiques : 23](#_Toc483997620)

[a.Le login 23](#_Toc483997621)

[b.Coté professeur 24](#_Toc483997622)

[c.Coté étudiant](#_Toc483997623) 26

[conclusion 29](#_Toc483997624)

[Bibliographie & webographie 31](#_Toc483997625)

# Introduction

Ce projet entre dans le cadre de la réalisation d’un projet de fin d’études en vue de l’obtention du diplôme universitaire de technologie, filière génie informatique au sein de l’école Supérieur de technologie de Safi.

C’est ainsi que nous avons eu l’occasion de préparer notre projet de fin d’étude intitulé «**Conception et réalisation d’une plateforme de loi optique en ligne avec Java** » proposé par notre encadrant M. BAKKAS JAMAL.

Ce projet est un apport très bénéfique quant au perfectionnement nos connaissances dans le domaine informatique et pour avoir l’opportunité d’appliquer nos connaissances théoriques acquises tout au long de notre cursus universitaire dans le cadre pratique.

# Chapitre 1 : Présentation du projet

1. Présentation

Le but de notre projet est de construire un site Internet qui met à la disposition des étudiants et des professeurs par le biais de l’Internet désormais incontournable un site web facilitant la gestion et l’accès aux cours ainsi que les simulations concernant les lois optique. Grace à cette plateforme en ligne, seuls les étudiants d’un professeur peuvent accéder aux cours et aux simulations qu’il met en ligne via un login et un mot de passe fournit par le professeur. Ce dernier peut ajouter et supprimer des cours, des simulations et des étudiants.

Notre site web doit être un site facile à mettre à jour sans avoir de grandes connaissances en programmation Web.

1. L’objectif

L’objectif du projet consiste à développer un site qui permettra de réaliser les opérations suivantes :

* Ajout et suppression des cours.
* Ajout et suppression des étudiants.
* Ajout et suppression des simulations.
* Le login et le mot de passe sont fournit par le professeur.
* Seuls les étudiants d’un professeur peuvent accéder aux données mise en ligne par ce dernier.

1. Synthèse

Après discussion, nous avons pu dégager certains points à éviter et d’autre qu’il vaut mieux utiliser pour notre projet et ce pour un résultat plus pertinent.

* 1. Point à éviter

La première chose à éviter est la création d’espace vide dans la page. La deuxième est de créer un équilibre entre le contenu textuel et les images. La dernière chose à éviter est l’utilisation de fichier de grande taille.

* 1. Point à retenir

Ce que nous avons pu relever de plus important est l’utilisation de formes rectangulaires et d’une structure de tableau dans tous les sites sujets de notre analyse. De plus tous ces sites ont utilisé la couleur bleue même si elle n’existe pas dans le logo.

1. Cahier des charges

A la fin de ce chapitre, nous allons énumérer les contraintes ergonomiques, fonctionnelle et esthétiques :

* 1. Contraintes esthétiques

Les principales contraintes esthétiques sont :

* La simplicité de l'interface.
* La convivialité (facilité d'utilisation de l'application).
* L'utilisation des effets des liens (changement de la couleur du lien, changement d'animation d'une interface a une autre, apparition d'un mot indicatif....)
* Quels que soient les goûts en matière artistique, le texte doit toujours rester lisible par rapport au fond d'écran.
* Écrire des phrases courtes à la construction simple.
* La lecture de texte sur écran se révèle vite fatigante. Des lignes de texte de 60 à 90 caractère maximum semblent être un bon compromis surtout si nous savons que la tendance est aux résolutions d'écran de 800 x 600 et 1024 x 768 ...
* Une navigation aisée avec des liens simples et personnalisés.
  1. Contraintes ergonomiques
* S'assurer de l'harmonie graphique de chaque page.
* Privilégier la clarté et la lisibilité.
* L'utilisation d'un nombre limité de couleurs met en valeur la force d'une charte graphique.
* Prévoir une possibilité de retour à la page d'accueil à chaque page. Un utilisateur perdu est un utilisateur déçu.
* Utilisation des couleurs significatives et qui reflètent l'aspect fonctionnel de l'application.

c. Contraintes fonctionnelles

* Nous ne pouvons pas passer d’un service à un autre que par le menu situé sur la plage d’accueil.
* La règle des trois clics. Toute information doit être accessible en 3 clics de souris maximum à partir de la page d'accueil.
* La règle des trois écrans. La longueur d'une page ne doit pas dépasser 3 à 5 écrans (grand maximum) pour limiter le défilement vertical.
* Le premier écran d'un page doit rassembler le maximum de l'information et inviter à en poursuivre la découverte. Un pourcentage important des utilisateurs n’utilise pas ou peu le défilement vertical.
* Le défilement horizontal de la page est à éviter (sinon à proscrire) quelle que soit la résolution d'écran de l’utilisateur.
* Un écran ne doit être rempli qu'à 50% de texte et de graphiques. Le reste est consacré au fond d'écran.

1. La simulation

Nous réaliserons un exemple d’une application accessible en ligne qui va permettre de réaliser la simulation de 4 lois optiques :

* *lentille convergente*
* *lentille divergente*
* *miroir concave*
* *miroir convexe*

1. Conclusion

A travers le cahier des charges qu’on a cité ci-dessus, on comprend bien que l’interface web que je vais réaliser est un site web dynamique mais pour l’implémenter j’ai besoins d’une étude conceptuelle de ce site. Cette étape va donner une vision globale sur la plateforme et les tâches qu’on doit réaliser afin d’atteindre les objectifs.

# Chapitre 2 : Conception

1. Introduction :

J’expose, dans ce chapitre, la solution conceptuelle que j’ai proposée et cette conception du système à réaliser qui a pour but de rendre flexible la tâche de la gestion. En d’autre terme, ce chapitre devrait répondre à la question : *comment faire* ? La structure de ce chapitre dépend de la nature de ce projet.

Une interface est, par définition, une jonction entre deux éléments d’un système informatique. Depuis l’arrivée des techniques du multimédia, le design d’interface est devenu une discipline à part entière qui requiert un large éventail de compétences : développeurs, designers, typographes, graphistes, cogniticiens, traducteurs, rédacteurs... Le design d’une interface utilisateur et son illustration graphique demandent une compréhension des principes de manipulation et des pratiques de l’utilisateur final. Un design d’interface réussi est ergonomique : il doit guider et informer l’utilisateur.

En pratique l’interface est constituée d’une suite d’objets visuels aux fonctions souvent multiples qui permettent à l’utilisateur d’interagir avec le programme multimédia. L’objectif premier d’un design d’interface est de créer chez l’utilisateur un modèle mental approprié et cohérent du contenu du programme et de son fonctionnement.

1. Conception du Site Web :
   1. Synopsis du Site web
      1. *Objectifs du site*
2. Faciliter l’accès aux ressources et services
3. Améliorer la qualité d’apprentissage en utilisant les nouvelles technologies multimedia
   * 1. *Cibles du site*
4. Notre application s'adresse principalement à des professeurs désirant Améliorer la qualité d’apprentissage et à n'importe quel dapprenants du professeur.
   * 1. *Charte graphique*
5. Les caractéristiques : site web en langues française.

*iiii. Organigramme et schéma de navigation*

Figure 1 : organigramme et schéma de navigation

* 1. Etudes des cas d’utilisation

Les cas d’utilisation décrient un ensemble d’actions réalisées par le système, en réponse à une action d’un acteur.

1. *Identification des acteurs*

Le formateur et l’apprenant sont les acteurs qui interagissent avec notre système.

***Professeur*** : assure les cours dans le salon de formation.

***Etudiant*** : bénéficie d’un accès aux cours disponible dans le site web.

1. *Identification des cas d'utilisation*

Nous décrivons pour chaque acteur les cas d’utilisation. On distingue les cas d'utilisation suivants :

**Formateur :**

* S’inscrire au site de formation
* S’authentifier
* Gérer son profil (mettre à jour ses informations personnels et la liste de ses formations)
* Ajouter une formation après la vérification de la disponibilité du salon de formation
* Consulter la liste des apprenants par Formation
* Uploader un cours
* Uploader et télécharger des documents.

**Apprenant :**

S’authentifier

Gérer son profil

Consulter la liste des formations Apprenant

Réclamer un problème

Télécharger des documents

avoir accès aux cours

avoir accès aux simulations.

1. Conception de l’application
2. L’objectif

Notre projet de fin d’études se concentre sur par la réalisation de l’application de simulation de lois optiques ont cherche à faciliter la compréhension du fonctionnement des systèmes optique en proposant plusieurs services de simulation de différent type de lentille et de miroir.

1. Les lois optiques

Dans l’application nous allons simuler la propagation de la lumière à travers un système optique. Les systèmes optiques qui seront abordé sont :

* Lentille divergente
* Lentille convergente
* Miroir convexe
* Miroir concave

Dans cette partie nous verrons des notions générale et quelque caractéristique de chaque système ainsi que les fonctions utiliser pour en faire la simulation.

* + 1. ***Système optique***

Un système optique (S) est un ensemble de milieux transparents séparés par des dioptres ou des miroirs.

Le système optique (S) sera représenté par un espace coloré, avec sa face d’entrée dans lequel pénètre un rayon incident et sa face de sortie de laquelle sort un rayon émergent. Le système est centré sur un axe appelé axe optique. La lumière se propage conventionnellement de la gauche vers la droite.

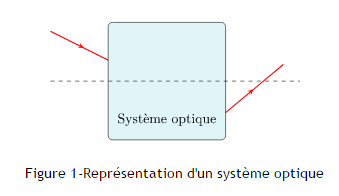
**

Figure 2 : Représentation d’un système optique

* + 1. ***Lentille:***
       1. Définition d’une lentille

Une lentille est un milieu transparent limité par deux dioptres, les deux peuvent être sphériques ou l’un est sphérique et l’autre est plan (on les nomme souvent lentilles sphériques).

* + - 1. types de lentilles

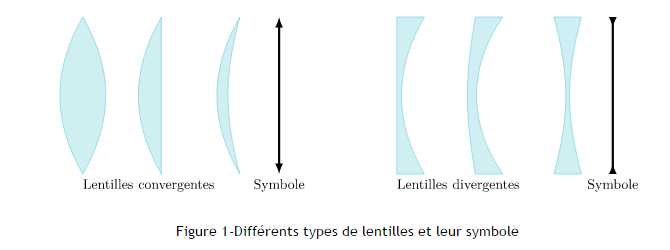
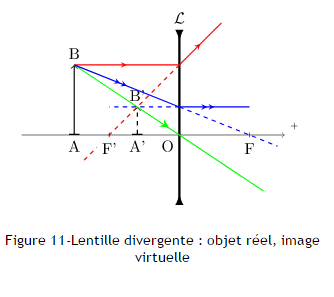
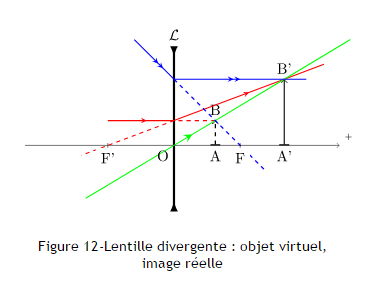
On distingue deux types de lentilles, celles à bords minces et celles à bords épais. Les premières sont convergentes, les secondes sont divergentes.

Figure 3 : différents types de lentilles et leurs symboles

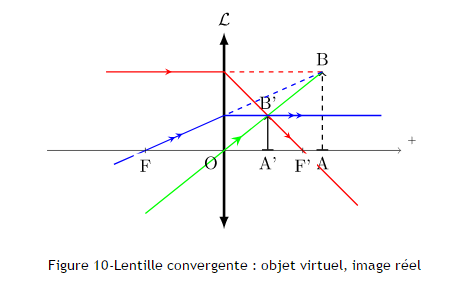
* + - 1. *lentille divergeante*

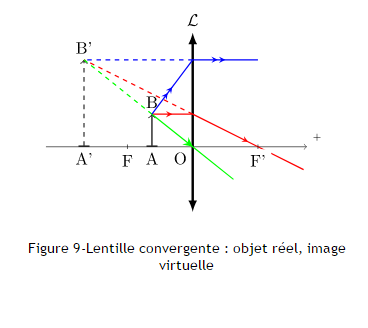
**

* + - 1. *lentille.convergeante*

Figure  : lentille divergente : objet réel, image virtuelle

Figure  : lentille divergente : objet virtuelle, image réel

**

**

16

Figure : lentille convergente : objet réel, image réel

Figure  : lentille convergente : objet virtuelle, image réel

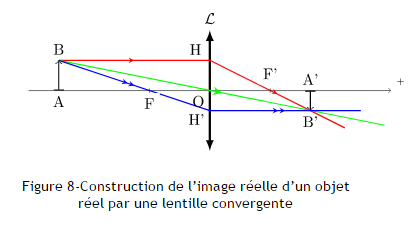
**

Figure : lentille convergente : objet réel, image virtuelle

* + 1. ***Les miroires***
       1. Définition d’une miroire

Un **miroir** est un objet possédant une surface suffisamment polie pour qu'une image s'y forme par réflexion et qui est conçu à cet effet.

* + - 1. *Miroire concave*

Pour établir l’image d’un objet, nous nous servons de la construction la plus classique de l’image d’un objet AB situé au delà du centre du miroir concave . Mais on notera que ceci est valable quelle que soit la position de l’objet et quelle que soit la nature du miroir.

Pour effectuer cette construction, on peut tracer quatre rayons dont les directions de propagation sont connues :

* Le rayon qui arrive parallèle sur le miroir est réfléchi en passant par le foyer ;
* Le rayon qui passe par le foyer se réfléchit dans le miroir en étant parallèle à l’axe optique ;
* Le rayon qui passe par C et qui se réfléchit dans le miroir n’est pas dévié (il a la direction de la normale).
* Le rayon qui frappe le miroir en son sommet est réfléchi avec un angle de réflexion identique à son angle d’incidence.

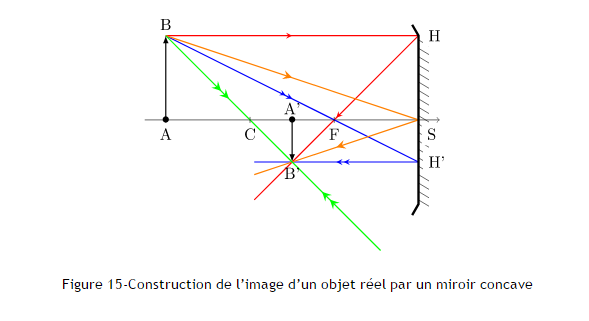
On obtient alors la construction suivante :

Figure 9 : construction de l’image d’un objet réel par un miroir concave

* + - 1. *Miroire convexe*

Pour établir l’image d’un objet, nous nous servons de la construction la plus classique de l’image d’un objet AB situé au delà du centre du miroir convexe . Mais on notera que ceci est valable quelle que soit la position de l’objet et quelle que soit la nature du miroir.

Pour effectuer cette construction, on peut tracer quatre rayons dont les directions de propagation sont connues :

* Le rayon qui arrive parallèle sur le miroir est réfléchi en passant par le foyer ;
* Le rayon qui passe par le foyer se réfléchit dans le miroir en étant parallèle à l’axe optique ;
* Le rayon qui passe par C et qui se réfléchit dans le miroir n’est pas dévié (il a la direction de la normale).
* Le rayon qui frappe le miroir en son sommet est réfléchi avec un angle de réflexion identique à son angle d’incidence.

On obtient alors les constructions suivantes :

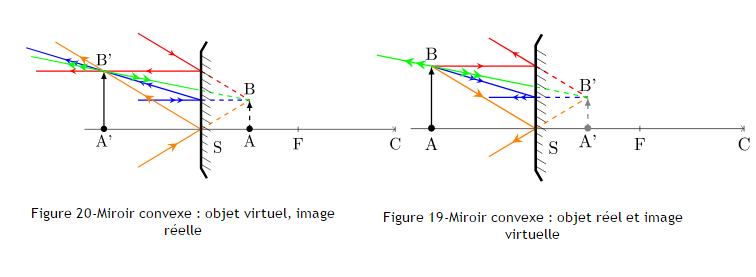


Figure 10 : miroir convexe

1. Conclusion

La phase conceptuelle est une étape fondamentale pour la réalisation de n’importe quel projet. Par la suite, je dois chercher les moyens et les outils possibles pour développer l’application, ce que je vais présenter dans la chapitre suivant.

# Chapitre 3 : Réalisation

1. Introduction :

Ce chapitre a pour objectif majeur de présenter le *produit finale.* C’est la phase de réalisation de ce site web dynamique qui utilise des technologies spécifiques. Ce chapitre est composé de deux parties : la première partie présente l’environnement de développement alors que la seconde partie concerne les principales interfaces graphiques.

1. L’environnement de développement :
   1. **Environnement Logiciel :**

Lors du développement de ce application, j’ai utilisé, les outils logiciels suivants:

* Serveur tomcat Apache.
* Netbeans
* Sublim text
* Scene builder

Je vais présenter ces différents logiciels dans la section suivante :

1. Langages utilisée :
2. **HTML/CSS**

Le HTML (« HyperText Mark-Up Language ») est un langage dit de « balisage » ou de « structuration » permettant la conception de pages web avec des balises de formatage. Les balises permettent d'indiquer la façon dont doivent être présentés le document et les liens qu'il établit avec d'autres documents.

Le CSS (« Cascading Style Sheets » : feuilles de style en cascade) est un langage informatique complétant le HTML. Alors que le HTML structure la page Web, le CSS va la mettre en forme en y apportant du style.

1. **JSP**

Le JavaServer Pages ou **JSP** est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique.

1. **Java fx**

**JavaFX** est une technologie créée par [Sun Microsystems](https://fr.wikipedia.org/wiki/Sun_Microsystems) qui appartient désormais à [Oracle](https://fr.wikipedia.org/wiki/Oracle_(entreprise)), à la suite du rachat de [Sun Microsystems](https://fr.wikipedia.org/wiki/Sun_Microsystems) par [Oracle](https://fr.wikipedia.org/wiki/Oracle_(entreprise)) le 20 avril 2009.Avec l'apparition de Java 8 en mars 2014, JavaFX devient la bibliothèque de création d'interface graphique officielle du langage Java, pour toutes les sortes d'application (applications mobiles, applications sur poste de travail, applications Web), le développement de son prédécesseur [Swing](https://fr.wikipedia.org/wiki/Swing_(Java)) étant abandonné (sauf pour les corrections de bug).

JavaFX est désormais une pure API Java (le langage de script spécifique qui a été un temps associé à JavaFX est maintenant abandonné).

JavaFX contient des outils très divers, notamment pour les médias audio et vidéo, le graphisme 2D et 3D, la programmation. Le SDK de JavaFX étant désormais intégré au JDK standard [Java SE](https://fr.wikipedia.org/wiki/Java_SE), il n'y a pas besoin de réaliser d'installation spécifique pour JavaFX. Voici ci-dessous un exemple de bouton fournit par javafx.

Donc JavaFX est au même titre que Swing une bibliothèque offrant des solutions pour créer des interfaces graphiques. Les seules différences sont que JavaFX est plus performant, plus configurable, plus génial en fait. Voici quelque nouveauté :

* Java fx nous donne la possibilité d’intégrer du code css
* Un fichier xml qui gère l’interface graphique.
* l’outil scène builder pour faciliter la construction de l’interface graphique.
* Le modèle MVC qui facilite le processus de construction de l’application.

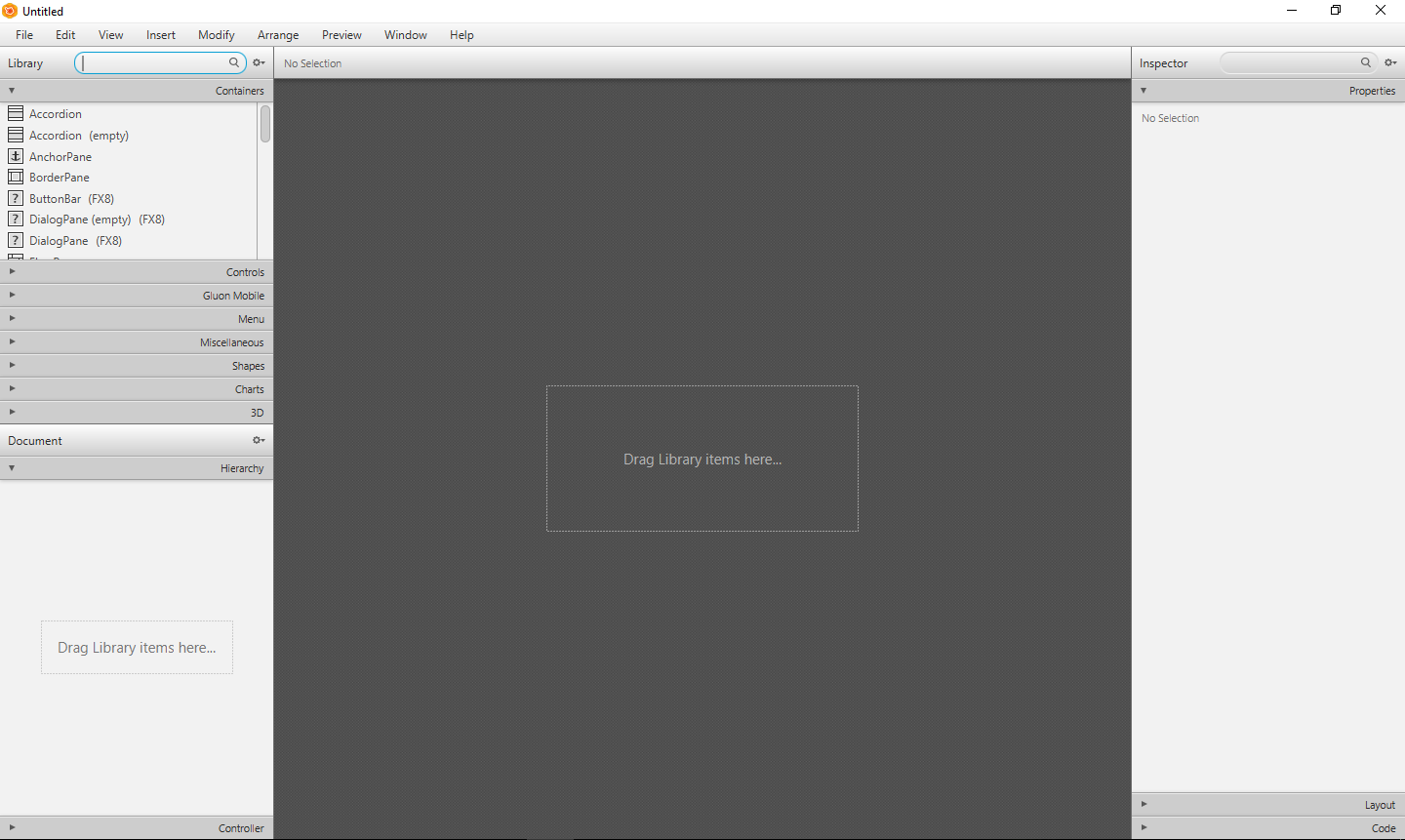


Figure 9 : scene builder

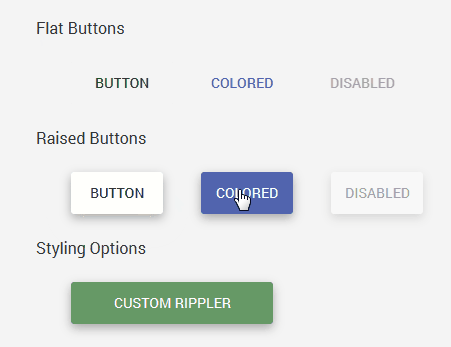


Figure 12 : quelque bouton fourni par javafx

1. **JDom :**

 Est une [bibliothèque](https://fr.wikipedia.org/wiki/Biblioth%C3%A8que_logicielle) [open source](https://fr.wikipedia.org/wiki/Open_source) pour manipulation des fichiers [XML](https://fr.wikipedia.org/wiki/XML) en Java.

L’utilisation de cette bibliothèque consiste sur la sauvegarde de l’état des objets dans fichier XML.

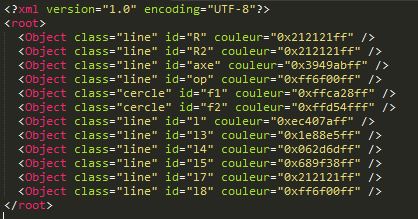
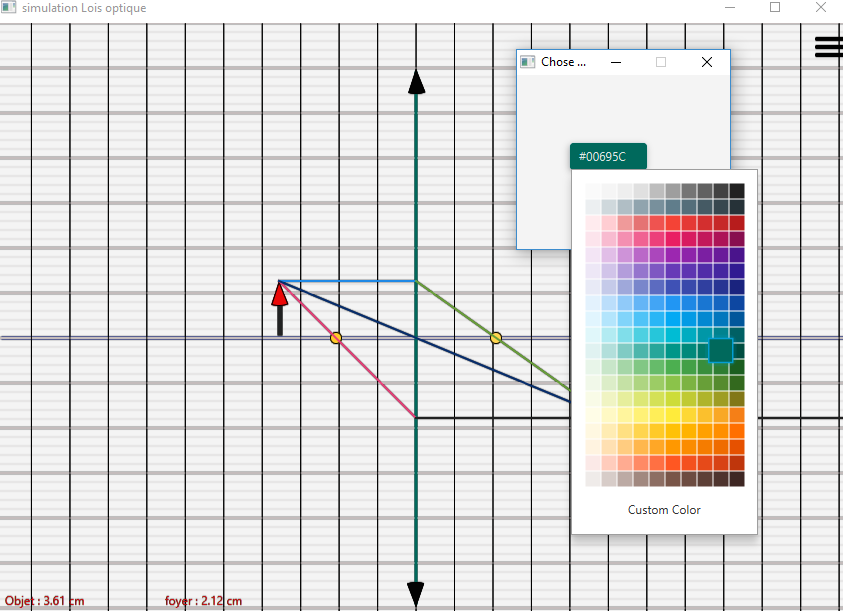


Figure 14 : Fichier XML qui récupère l’état

Figure 13 : changer la couleur des lignes



1. Les principales interfaces graphiques :

Cette partie me permet de vous mettre dans les conditions réelles d’utilisation de l’application.

* 1. Le login

La page de login c’est la première page de notre site web, qui s’affiche aux étudiant et aux professeur :

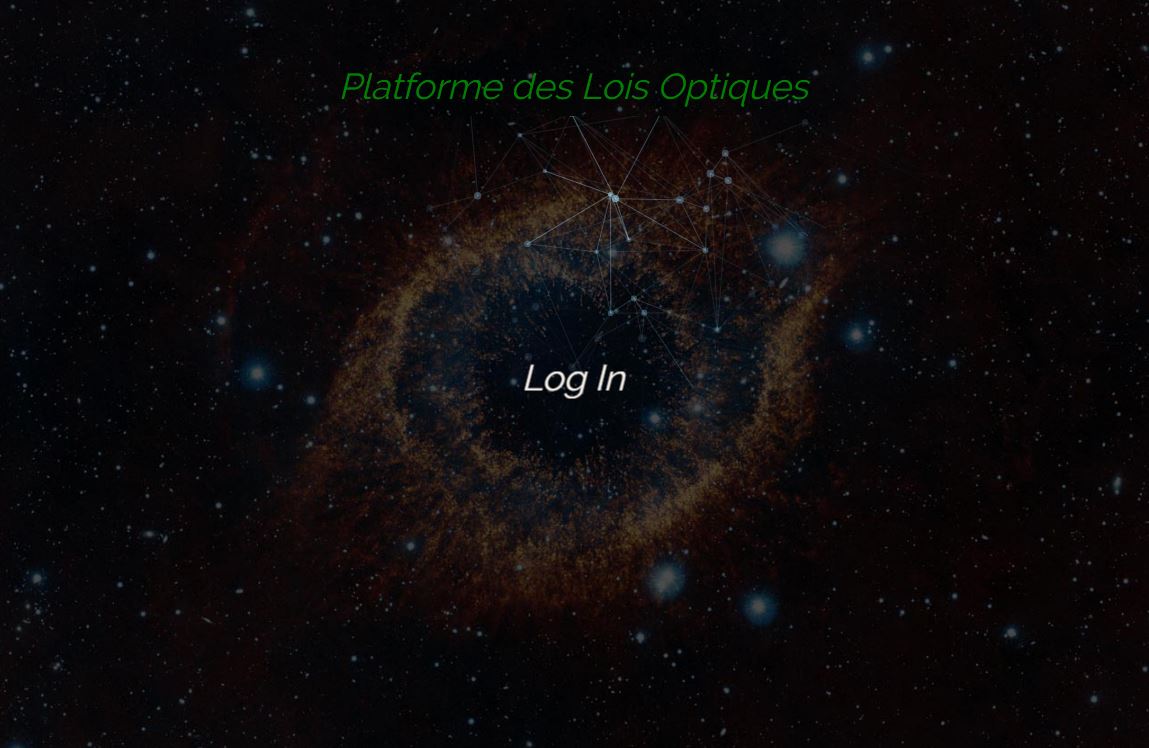


Figure 15 : Page qui nous donne l’accès au site

Apres avoir cliqué sur le bouton login on a les champs suivants pour entrer les informations nécessaire pour accéder aux espace personnel :

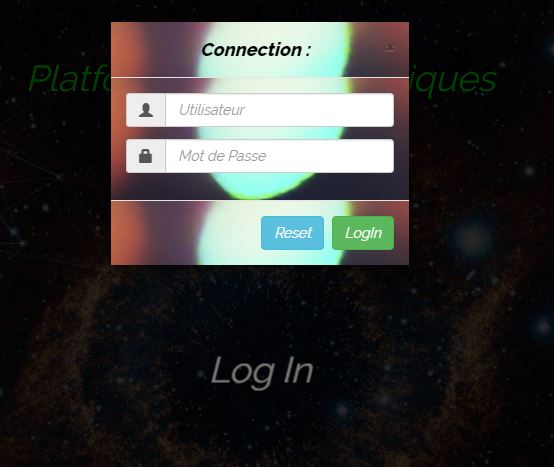


Figure 16 : zone pour saisir les informations d’accès au site

* 1. Coté professeur

Lorsqu’un professeur accède à la plateforme, il est rediriger vers la page suivante :

Il peut choisir d’entrer en tant qu’étudiant pour qu’il soit rediriger vers la page qui s’affiche aux étudiants, ou d’entrer en tant que professeur pour qu’il soit rediriger vers la page de gestion des cours, étudiant ou simulation.

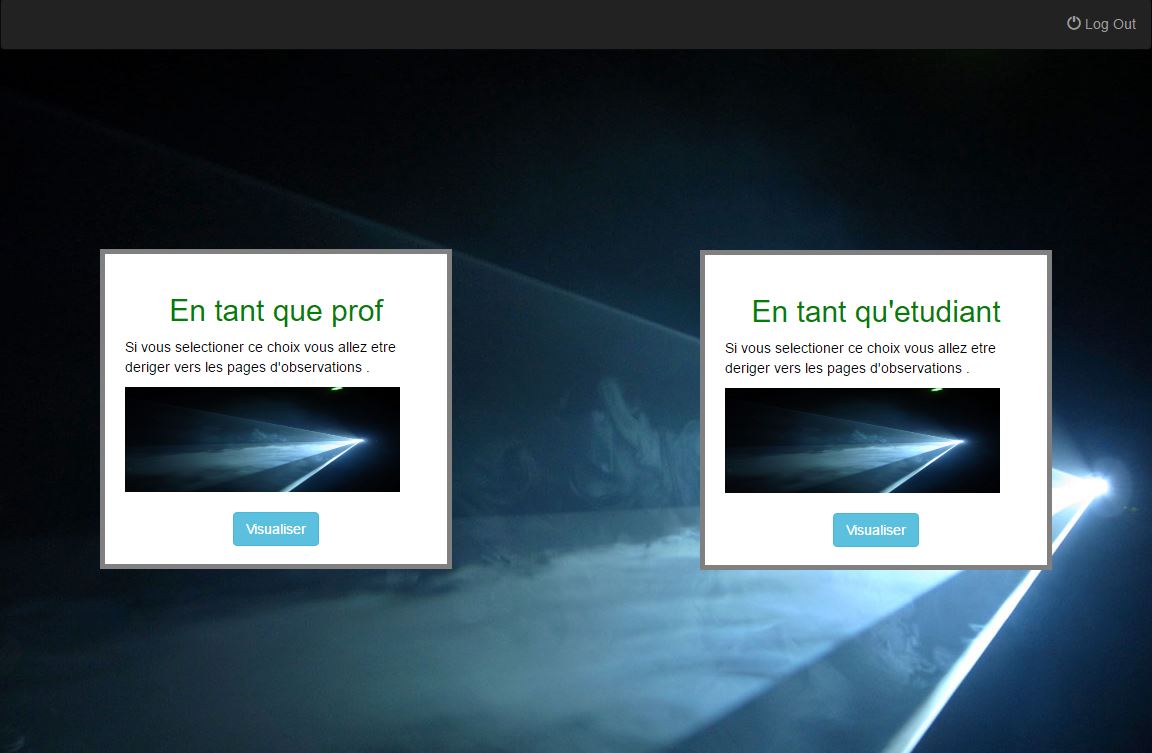


Figure 17 : page d’accueil du professeur

En tant que professeur il ‘est rediriger vers la page suivante :



Figure 18 : page de gestion des cours



Figure 19 : page de gestion des étudiants

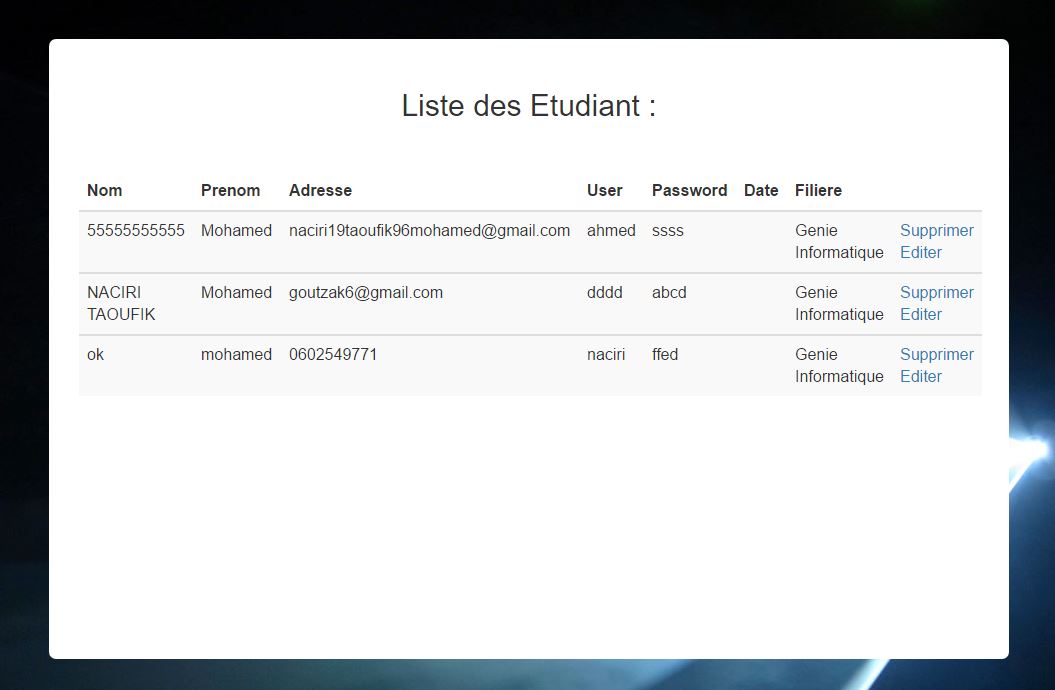


Figure 20 : tableau contenant la liste des étudiants

* 1. Coté étudiant :

L’accueil c’est la première page de notre site web, qui s’affiche aux étudiant après login :

* Accueil
* Cours
* Simulation
* Log out



Figure 21 : page d’accueil

Dans l’onglet cours on trouve tous les cours disponible sur notre site web :

Figure 22 : page des cours

 Dans l’onglet simulation on a une page web qui nous permet d’accéder à la simulation :

Figure 23 : page conduisant vers la simulation

Apres avoir cliqué sur le bouton démarrer la simulation la fenêtre de l’application de la simulation s’ouvre comme suit :

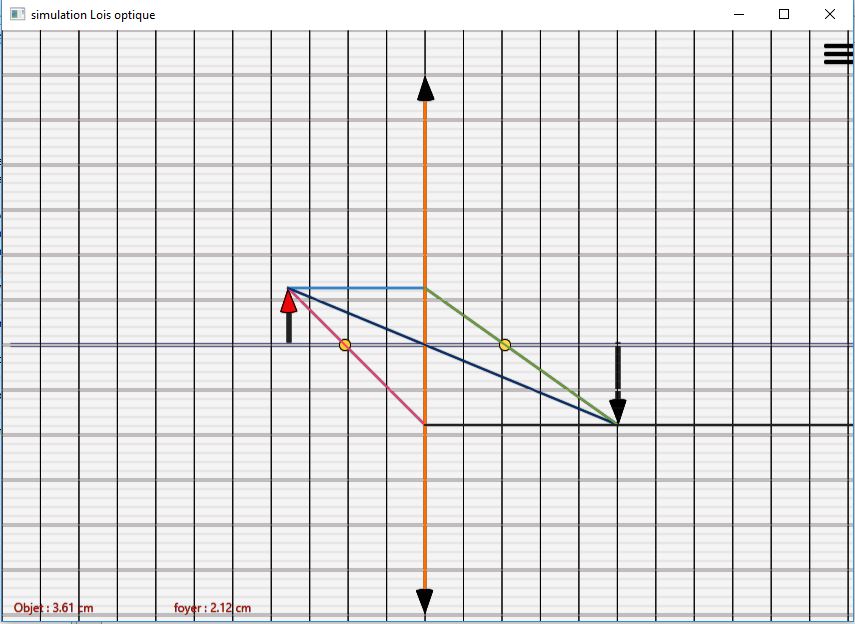


Figure 24 : simulation en java

L’application nous permet de réaliser 4 simulations :

* lentille convergente
* lentille divergente
* miroir concave
* miroir convexe

L’utilisateur peut choisir dans un menu le type de simulation et quelque personnalisation

Lorsqu’on appuis sur optique on a le menu suivant :

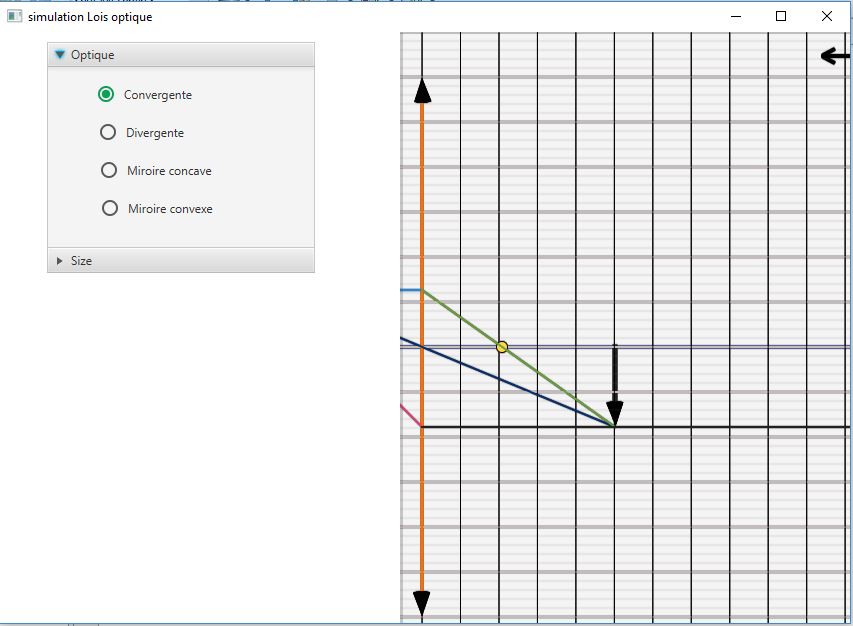
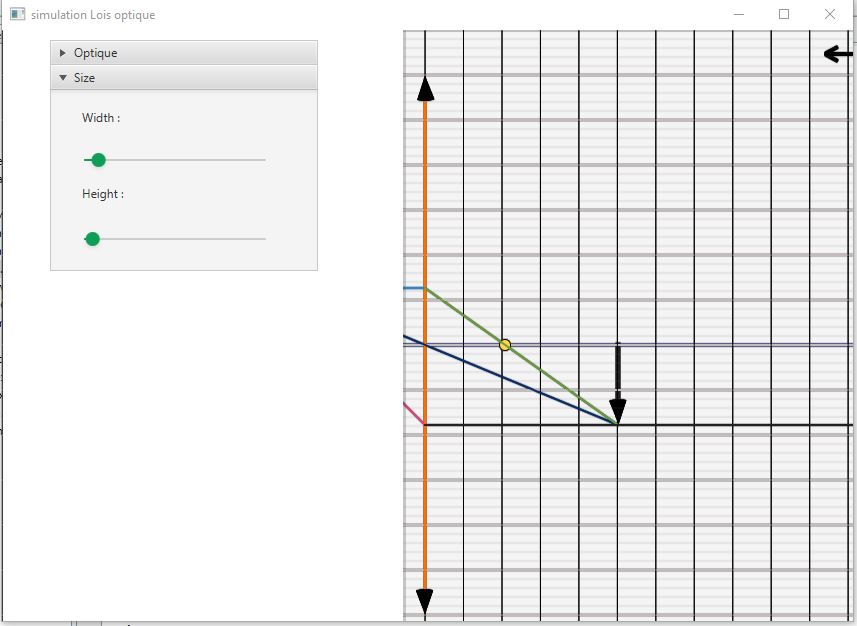


Figure 25 : Chois du type de simulation



Lorsqu’on appuis sur size on a le menu suivant :

Figure 26 : Chois du type de simulation

# Conclusion

Au terme de ce projet de fin d’étude au sein de l’école supérieure de technologie de Safi, nous avons eu la satisfaction d’avoir réalisé un projet.

Pour la réalisation de ce projet, nous avons utilisé une démarche structurée en partant d’un cahier des charges et en respectant les différentes étapes de réalisation et de tests. Nous avons validé ce projet en respectant l’ensemble de ses fonctions.

Au cours de la période de notre projet de fin d’étude, nous avons eu l’opportunité de mettre en application différentes connaissances acquises durant notre formation DUT génie informatique. En plus ce projet nous a permis de développer notre esprit d’équipe sachant bien que ce type de travail provoque des conflits grâce aux différentes pensées.

Nous souhaitons que notre projet trouve les conditions nécessaires pour entrer en vigueur.

Bibliographie & webographie

https://openclassrooms.com

http://w3schools.com

http://tomcat.apache.org/