

Livrable 4 – Pationator

Présenté à

M. Jonathan Gaudreault

Génie logiciel orienté objet (GLO-2004)

Réalisé par

Olivier Beauséjour,

Scott Chalmers,

Jonathan Mathieu et

Derek Pouliot

Université Laval

Automne 2020

14 décembre 2020

Table des matières

| | | |
|----|--|---|
| 1. | Énoncé de vision | 3 |
| 2. | Saisie d'écran de la version finale de l'application | 4 |
| 3. | Modèle du domaine | 5 |
| 4. | Diagramme des cas d'utilisation..... | 6 |
| 5. | Diagramme de classes de conception..... | 7 |
| 6. | Conclusion | 8 |
| 7. | Contribution de chacun des membres de l'équipe..... | 9 |

1. Énoncé de vision

Pationator est un logiciel de conception de plans de terrasses rectangulaires en bois. À l'aide d'une interface simple et intuitive, l'application permet à l'utilisateur de concevoir des terrasses respectant les normes du Conseil canadien du bois et de les visualiser sous trois perspectives différentes. Afin de concevoir son plan de terrasse, l'utilisateur a la possibilité de spécifier différentes caractéristiques, telles que la taille du patio, les types de matériaux de construction utilisés et leurs prix associés, ou encore, l'espacement entre les différentes pièces constitutives. L'application utilise ces différentes propriétés entrées par l'utilisateur, pouvant être modifiées par la suite en temps réel, pour générer une terrasse. Ainsi, l'usager est directement informé de la validité de son plan selon les normes canadiennes.

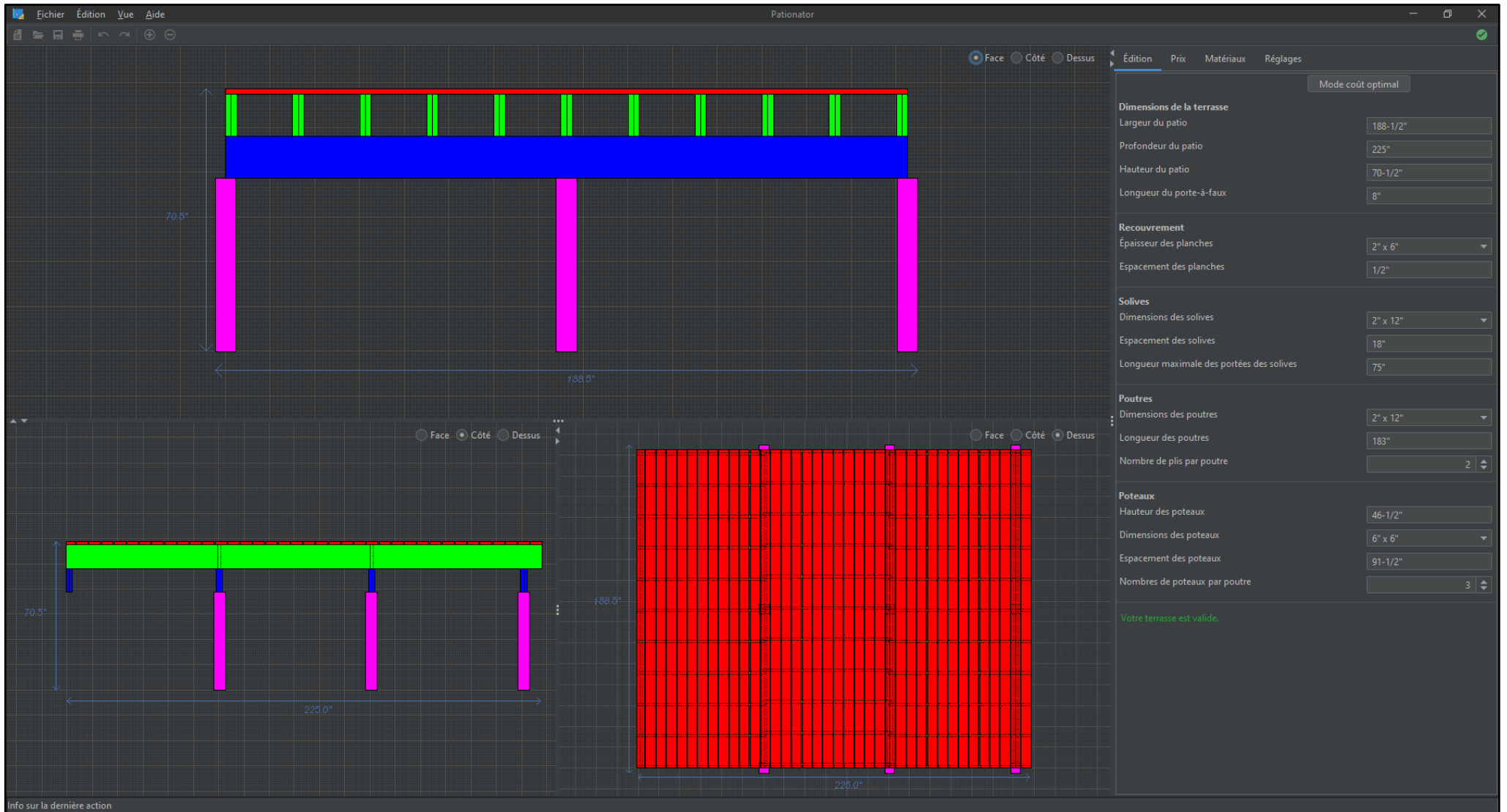
Au sein même de l'interface, l'utilisateur peut également visualiser les dimensions de chacune des pièces utilisées. Afin de rendre le tout non-ambigu et intuitif pour l'utilisateur, celles-ci sont affichées à la fois en taille réelle et en taille théorique.

De plus, dans un souci monétaire, le logiciel offre une fonctionnalité de génération automatique d'un plan de patio au coût optimal selon certaines spécifications fournies par l'usager. Ce dernier obtient alors directement un plan au coût minimal possible.

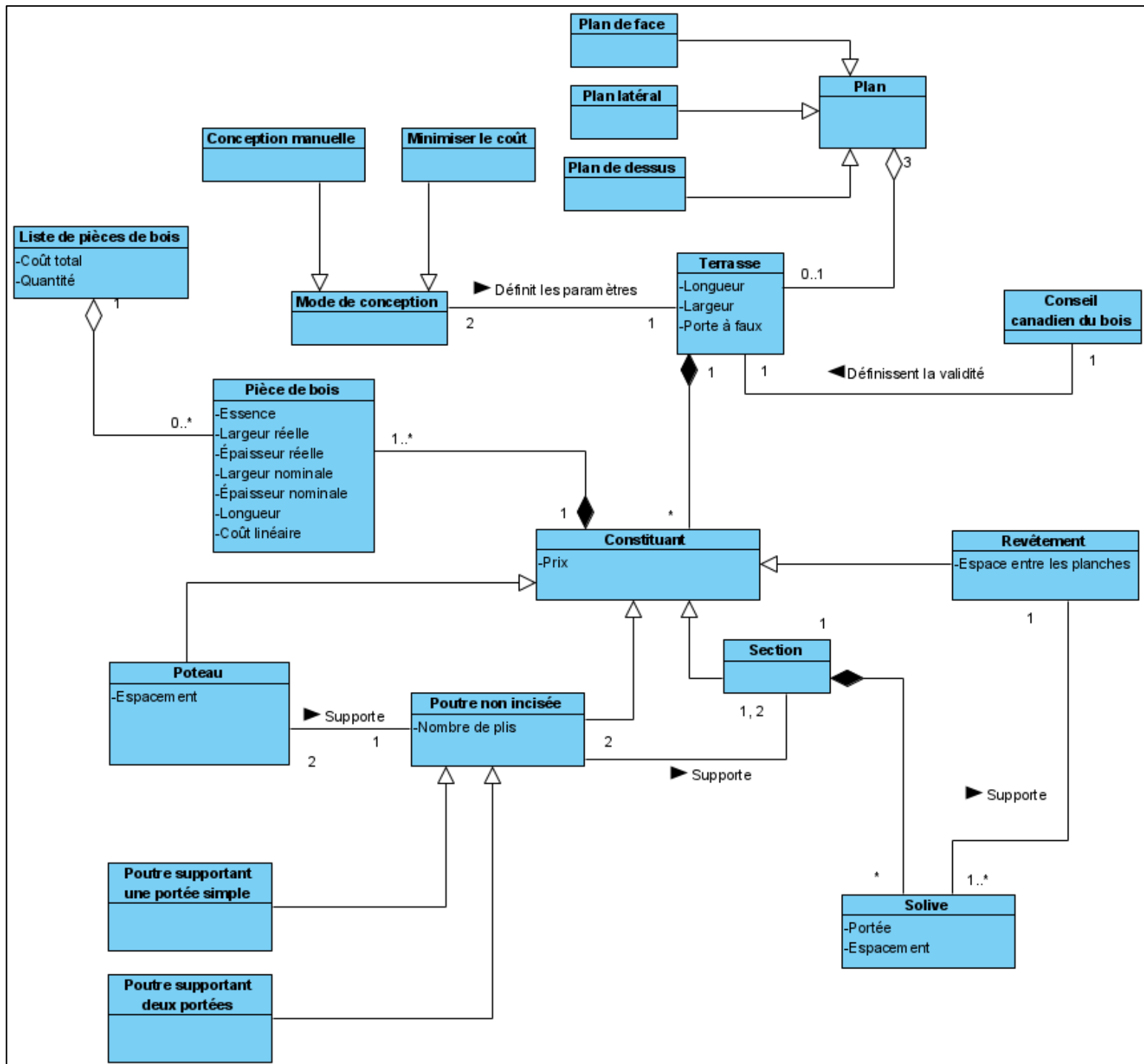
Lorsque l'utilisateur le désire, il est possible de sauvegarder le plan sous forme d'image pour les différentes vues et d'obtenir une liste de toutes les pièces nécessaires à la fabrication accompagnée de leurs prix. Le plan peut également être exporté en format .STL pour permettre la visualisation de la terrasse dans diverses applications, ou en faire une impression 3D.

Ainsi, *Pationator* est un outil qui simplifiera grandement le travail des designers ou toute personne intéressée par la conception de terrasses par sa facilité d'utilisation et sa quantité importante de fonctionnalités.

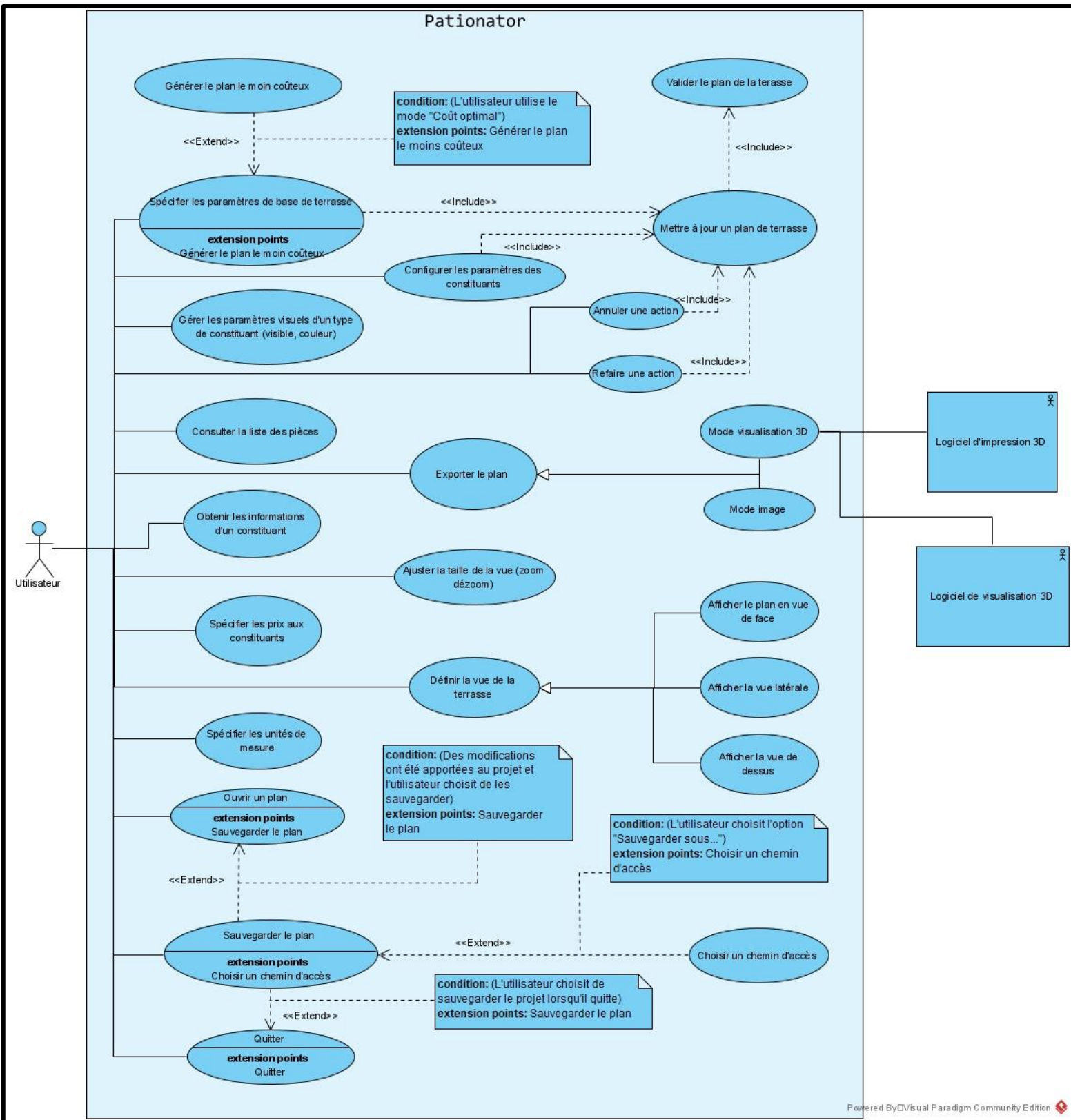
2. Saisie d'écran de la version finale de l'application



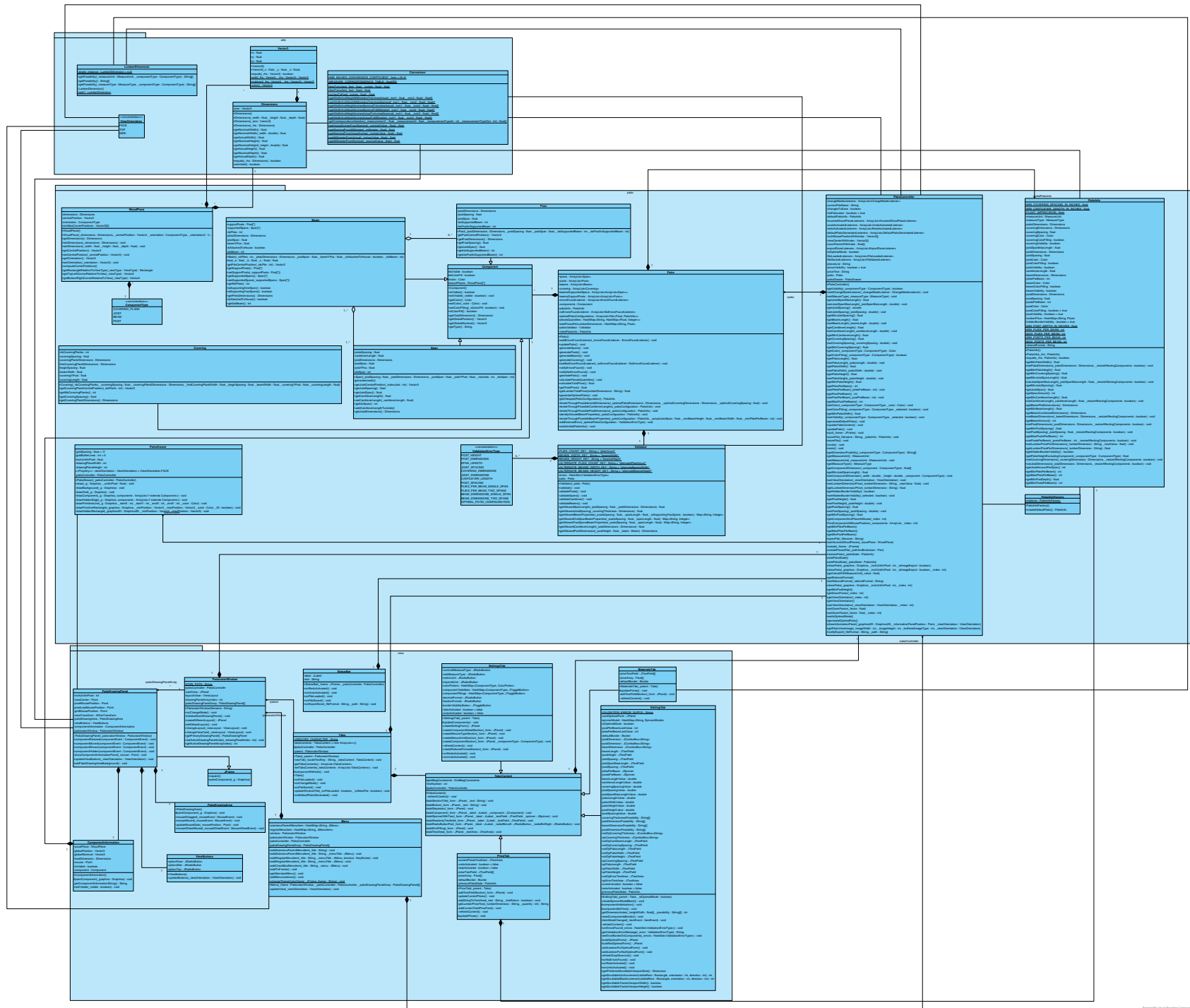
3. Modèle du domaine



4. Diagramme des cas d'utilisation



5. Diagramme de classes de conception



6. Conclusion

En ce qui concerne les points forts, notre application se distingue par son interface très intuitive et personnalisable. Effectivement, l'utilisateur n'a qu'à entrer une minime quantité d'information afin d'obtenir une configuration de terrasse désirée, et l'ensemble de ces propriétés spécifiées se retrouvent dans un seul conteneur. Ainsi dit, dans un contexte professionnel, n'importe quel ingénieur ou constructeur de patio amateur pourrait facilement utiliser l'application.

Nous considérons également que le validateur de patio est de qualité professionnelle. Incontestablement, ce dernier permet non seulement d'identifier précisément les cas d'erreurs qui ne respectent pas le guide du *Conseil canadien du bois*, il indique nettement les propriétés à modifier afin de rendre la terrasse valide. Ainsi, il est simple et facile d'obtenir un patio valide, sans avoir à se casser la tête.

Enfin, notre application assure de maximiser le potentiel d'efficacité de l'utilisateur y faisant usage. Effectivement, notre vue, pouvant être séparée en trois perspectives différentes de façon simultanée, permet de visualiser la terrasse sous tous ses angles, sans avoir à constamment itérer au travers de ces dernières. Il est à noter que les options de perspectives sont personnalisables, où l'utilisateur peut décider la taille et la disposition de ces fenêtres de vues.

Toutefois, en ce qui concerne les points négatifs, nous croyons que notre application peut faire preuve de limitations de design. Effectivement, dans un contexte professionnel, notre application ne permettrait pas de générer des terrasses de formes autre que rectangulaire. Il est également à noter que les sections du patio sont strictement de la même longueur, sauf pour la section comprenant le porte-à-faux. Ainsi, cela peut apporter de nombreuses restrictions en ce qui concerne la gamme de terrasses pouvant être générées. De plus, notre application ne permet également pas la génération de garde-corps ou d'escaliers.

D'autre part, nous considérons que notre application fait preuve d'un léger irréalisme en ce qui concerne la génération des composants du patio. Effectivement, il se peut que certaines planches de bois générées comportent des longueurs ne pouvant pas se retrouver en magasin (par exemple, une planche de recouvrement pourrait être longue de 50 mètres, dans une configuration pourtant valide). Une approche plus professionnelle aurait été de segmenter ces planches en dimensions pouvant être vendues en magasin.

Somme toute, nous apprécions grandement le résultat final de notre application *Pationator*, et nous jugeons qu'avec quelques modifications légères, nous pourrions obtenir un résultat encore plus professionnel.

7. Contribution de chacun des membres de l'équipe

Pour cette quatrième partie du projet, afin de bien se partager les tâches à réaliser en équipe, nous avons décidé de séparer le travail à faire, encore une fois, en fonctionnalités, un peu comme dans un processus Agile où on assigne différentes fonctionnalités d'un carnet de produit (*backlog*) selon les forces de chacun des membres de l'équipe. Ainsi, chacun des membres a surtout travaillé sur des fonctionnalités plutôt proches de celles sur lesquelles ils ont travaillé durant le dernier livrable.

D'abord, par rapport à Scott Chalmers, il a développé l'export du patio en format STL, ainsi que l'export du patio en format image (JPG et PNG). Par la même occasion, il a ajouté la possibilité d'imprimer les différentes vues. Pour ajouter des fonctionnalités bonus, Scott a ajouté la possibilité d'avoir des vues multiples sur le patio avec différentes dispositions de fenêtre, ainsi que des flèches sur les plans pour indiquer la hauteur, la largeur et la profondeur du patio. Suite aux commentaires de la présentation en classe, il a développé une classe pour prendre en considération les fractions et a adapté la vue pour permettre d'afficher les valeurs en métrique, en impérial avec fraction ou en impérial en décimal. Pour faciliter l'observation du patio, la possibilité de centrer la vue sur le centre du patio fut également ajoutée, que ce soit depuis le menu, ou lors du changement de la vue. Pour conclure, une petite page "À propos" fut ajoutée dans le menu pour présenter Pationator à l'utilisateur. Un package de visualisation 3D était également en cours de développement, basé sur une vidéo explicative¹, lors de la remise, mais l'équipe a préféré ne pas l'inclure dans cette version, car l'affichage du patio n'était pas optimal. Les contrôles de la caméra depuis le clavier ont été implémentés par Scott, ainsi que le dessin des composants du patio dans la bonne position et bonne orientation.

Du côté d'Olivier Beauséjour, ce dernier s'est principalement occupé de la génération optimale de la terrasse. Pour commencer, Olivier s'est chargé de régler les problèmes de génération de patio, suite aux commentaires de la première démonstration. D'emblée, c'est Olivier qui s'est occupé de créer l'algorithme testant l'ensemble des configurations possibles (valides) de terrasses selon les paramètres spécifiés par l'utilisateur, et de sélectionner celle qui est la moins chère en considérant le prix linéaire de chaque pièce de bois. Ainsi dit, c'est Olivier qui s'est occupé de calculer le prix total du patio, et de le séparer en sous-totaux afin de connaître le prix pour chaque type de pièces dans la liste des matériaux. Enfin, c'est Olivier qui a implémenté le comportement de création d'un nouveau projet Pationator.

En ce qui concerne Derek Pouliot, à l'instar du troisième livrable, celui-ci s'est attardé sur différentes tâches restantes du projet. Ainsi, il s'est d'abord occupé du système complet de contrôle d'action : c'est lui qui s'est occupé de la possibilité d'annuler ou de refaire une action dans l'interface. De même, Derek s'est occupé de l'affichage des différentes pièces de bois utilisées pour construire le patio. En complément avec cela, il s'est également occupé de l'affichage du prix total (et des sous-totaux) du patio. Il a ensuite réalisé une fonctionnalité permettant d'exporter ces informations (pièces et prix) sous la forme d'un fichier texte. Finalement, Derek s'est occupé de l'affichage d'un message d'action dans la barre de statut de l'application (ex. : Message spécifiant la réussite d'une exportation de fichier).

¹ <https://www.youtube.com/watch?v=gnT6YC5Nf70&list=PLsRmsZm0xMNogPyRn6gNWq4OM5j22FkAU>

Enfin, Jonathan Mathieu a encore travaillé principalement sur l'affichage de la terrasse. Tout d'abord, il a retravaillé sur le zoom/dézoom pour qu'il s'effectue vers la position de la souris. De plus, il a corrigé certaines imprécisions survenant lors du zoom/dézoom pour qu'il soit très précis. Ces mêmes imprécisions causaient une mauvaise détection de l'emplacement des pièces sur laquelle la souris se retrouvait. Donc, en soi, la correction de ces imprécisions a rendu la détection des pièces et le zoom/dézoom impeccable. De plus, il a limité la vitesse d'appel des fonctions de mises à jour de la souris, car la vue ne se mettait pas à jour assez vite pour ces appels, ce qui causait des bugs visuels déplaisants. Finalement, il a essayé de développer une solution d'affichage 3D dans un autre projet pour tester la faisabilité de l'introduction de ce type de vue dans l'application, mais l'avancement de la solution n'a pas été assez rapide pour l'utiliser fonctionnellement dans l'application.