

GLO-2005: Projet de cours

Description

Pour ce projet, vous devrez réaliser une application web suivant une architecture à trois niveaux et utilisant les technologies vues en classe (HTML, CSS, JavaScript, Python, Flask, et MySQL). Ce projet est à réaliser en équipes de 2 à 4 personnes (idéalement 3), et compte pour 25% de votre semestre. De plus, l'équipe produisant le meilleur projet sera considérée pour le Prix Pierre-Ardouin (voir plus bas).

Vous êtes libre des aspects créatifs de votre application, incluant le sujet, les données à gérer, et l'apparence du site. Des exemples (non-exhaustifs) d'applications possibles incluent :

- Un magasin en-ligne, incluant un catalogue de produits, des comptes pour les clients, et des opérations de magasinage. Des exemples réels incluent Amazon.ca et eBay.ca.
- Un répertoire d'information partagé, incluant des pages documentant chaque instance, des pages personnelles pour les contributeurs, et des opérations d'édition. Des exemples réels incluent IMDb et Wikipédia.
- Une communauté virtuelle, incluant des profils individuels pour les membres, un historique des activités, et des opérations de contacts interpersonnels. Des exemples réels incluent Facebook et LinkedIn.
- Un service en-ligne, incluant un répertoire d'information, des comptes utilisateurs, et des opérations de requêtes et de suivi. Des exemples réels incluent l'Université Laval et la Bibliothèque de Québec.

Ces projets peuvent être ludiques et utiliser des données fictives mais réalistes (voir plus bas). Alternativement, si un projet pertinent réel est disponible à travers une compagnie ou un groupe de recherche, vous pouvez l'utiliser. Des projets réels en collaboration avec des OBNL locales sont disponibles à travers le service Accès Savoir de l'Université Laval. En cas de doutes sur la validité du projet que vous désirez faire, consultez le professeur afin d'éviter les mauvaises surprises.

Réalisation et évaluation

La réalisation d'un tel système devra passer par la série d'étapes suivante :

1. Énonciation du problème et des exigences. (2 point)

Décrivez quelle application vous avez choisi d'implémenter, et dans quel contexte elle doit opérer. Déterminez quelles seront les exigences qui existent sur cette application, et comment elles impacteront le design et les fonctionnalités du système. Comme piste de réflexion, vous pouvez vous demander quels seront les différents

groupes d'utilisateurs de votre application, quels seront leurs besoins pour lesquels ils l'utiliseront, et donc les fonctionnalités exigées de votre système pour répondre à ces besoins.

En conséquence de cette réflexion, définissez les responsabilités de chacun des trois niveaux du système, le niveau client, serveur d'application, et base de données. En d'autres mots, déterminez pour chaque fonctionnalité quelle portion de traitement doit être réalisé par chacun de niveaux et quelles informations doivent être transférées d'un niveau à l'autre.

2. Modélisation des données. (3 points)

Créez le modèle entité-relation du système. Celui-ci devra découler logiquement de votre application, de ses contraintes, et de ses fonctionnalités. Puis, créez le modèle relationnel du système (schéma seulement). Assurez-vous qu'il soit bien cohérent avec le modèle entité-relation.

3. Création de la base de données. (3 points)

Créez la base de données dans MySQL et peuplez-la de données (voir plus bas). Les relations doivent être cohérentes avec la modélisation que vous avez faite. Toutes les contraintes d'intégrité nécessaires doivent être présentes. Expliquez également d'où proviennent vos données (comment elles ont été générées ou obtenues).

4. Création des requêtes et des routines. (3 points)

Créez les requêtes nécessaires pour l'utilisation de votre système. Ne vous limitez pas à des requêtes SELECT simplistes; créez des requêtes avancées qui implémentent un comportement sophistiqué dans votre système (ex. : des requêtes imbriquées, des agrégations, des jointures explicites, etc.) De plus, vous devez utiliser des routines (gâchettes, fonctions, ou procédures) dans votre implémentation. Ces routines doivent servir à automatiser un traitement complexe et utile des données; elles ne peuvent pas simplement dupliquer des contraintes d'intégrités ou encapsuler une seule requête. N'oubliez pas de commenter votre code et de gérer les erreurs.

5. Indexation et optimisation du système. (1 points)

Définissez des index sur les données pour optimiser l'exécution des requêtes. Assurez-vous que votre système fonctionne de manière optimale; vos choix d'index doivent donc découler de l'utilisation prévue de votre système par les utilisateurs et de la charge de travail anticipée.

6. Normalisation des relations. (1 points)

Déterminez la forme normale de chacune des relations de votre système. Pour celles qui ne sont pas FNBC, vous devez soit les normaliser, soit justifier votre décision de les laisser dans une forme normale plus faible.

7. Sécurité de la BD. (1 points)

Identifiez les risques potentiels de perte ou de vol des données dans votre système, et corrigez-les afin de vous assurer que votre système soit sécurisé. En particulier, le chiffrement des mots de passe des utilisateurs (s'il y en a dans votre système) est obligatoire.

8. Implémentation de la logique d'affaire. (2 points)

Créez les fonctions Python et Flask pour implémenter les fonctionnalités de ce niveau. Afin d'assurer une bonne performance, ce niveau devrait valider les informations provenant et allant aux deux autres niveaux, et minimiser les communications inutiles entre les niveaux. Ce niveau devrait également attraper des cas d'erreurs inattendus pour éviter l'échec du système (par exemple avec l'utilisation de `try-except`). N'oubliez pas de commenter votre code.

9. Implémentation de l'interface utilisateur. (2 points)

Créez le site web pour votre application. Ce site web doit permettre à l'utilisateur de demander et retrouver facilement les informations pertinentes à votre application et être organisé de manière logique. Il devrait également valider les informations entrées par l'utilisateur pour éviter les erreurs de saisies ou les attaques délibérées. L'apparence du site est libre à vous (et n'est pas évaluée); cependant, cette apparence doit être cohérente d'une page à l'autre à l'intérieur du site, et donc utiliser des CSS. Vous devez également implémenter des fonctionnalités utiles au traitement de données à ce niveau (ex.: validation des entrées) en utilisant JavaScript. N'oubliez pas de commenter votre code et de gérer les erreurs.

10. Tests du système. (2 points)

Assurez-vous que le système fonctionne bien dans son ensemble, que les trois niveaux sont bien intégrés et communiquent bien ensemble lors de l'utilisation normale du système. Assurez-vous que les cas d'erreurs sont bien gérés (par exemple, que ce passe-t-il si un utilisateur entre une chaîne de caractère dans un champ de la page web qui devrait normalement recevoir un nombre?).

11. Accessibilité du système. (1 points)

Une diversité d'utilisateurs seront appelés à utiliser votre système, incluant certains avec des besoins différents de la moyenne. On peut penser ici à des contraintes techniques (ex. : connexion internet lente ou peu fiable), des conditions médicales (ex. : daltonisme, troubles de lecture), ou des situations personnelles particulières (ex. : sexe non-binaire à entrer dans une variable binaire du profil utilisateur) qui rendent l'accès et l'utilisation de votre système plus compliqués, voire impossible. Identifiez les limites d'accessibilité de votre système, et corrigez-les afin de vous assurer qu'un maximum d'utilisateurs puissent bénéficier de votre système.

12. Gestion de l'équipe et organisation du travail. (1 point)

Comme il s'agit d'un projet en équipe d'une certaine complexité, pour le réaliser avec succès vous devrez vous assurer de maintenir une bonne coordination et

communication au sein de l'équipe. Tout au cours du projet, vous devrez diviser les tâches entre les membres de l'équipe de manière équitable et efficace. Vous devrez organiser des rencontres régulières afin que chaque membre de l'équipe soit au courant du progrès global du projet. Une attention particulière devra être portée à vous assurer que les interactions entre les différentes parties du projet, développées par différentes personnes, soient bien planifiées et implémentées.

13. Vidéo démo. (3 points)

En plus d'un rapport technique présentant les points plus haut, vous devez produire une vidéo qui présente votre projet, tant ses aspects techniques que son bon fonctionnement. Ces éléments seront détaillés plus bas.

Données

Dans le cadre de ce projet, il est acceptable d'utiliser des données fictives pour peupler votre base de données. Un script simple peut facilement générer et emmagasiner dans la BD une bonne quantité de données avec des valeurs aléatoires. Les valeurs ont besoin d'être réaliste, sans pour autant nécessiter d'être réelles et exactes. Par exemple, pour un catalogue en-ligne, les prix des produits doivent couvrir une gamme de valeurs allant de quelques dollars à quelques milliers de dollars, mais que la valeur exacte d'une voiture affichée soit 35 000\$ ou 3.50\$ ne fait aucune différence. De même, les messages échangés entre des utilisateurs sur un réseau social doivent varier de quelques mots à quelques paragraphes de longueur, mais que le contenu des messages soit une vraie correspondance ou un *lorem ipsum* n'importe pas.

Votre BD devrait avoir au moins 6 relations différentes, dont au moins 3 doivent avoir un degré de 5 attributs ou plus, et au moins 2 doivent avoir une cardinalité d'au moins 100 tuples.

Rapport technique

Le rapport technique devra documenter clairement tout le travail que vous avez fait à chaque étape de la réalisation du projet, tant les aspects techniques du système que les réflexions ayant mené à sa création. Il y a un rapport à remettre par équipe, en utilisant le site web du cours.

Le rapport doit être rédigé dans le même style que ce présent document (police Times New Roman 12pt, interligne 1.15, espace de 6pt entre les paragraphes, marges de 2,54cm) et avoir une limite de 12 pages. Cette limite s'applique au contenu technique, incluant les tableaux et figures. Cette limite exclue la page titre, la table des matières, et les annexes. Cette limite sera strictement appliquée, et des points seront retirés pour les débordements.

Un aspect important du rapport est de justifier vos décisions de design du système. En effet, chaque décision devrait découler logiquement des conséquences des étapes précédentes. Par exemple, le modèle entité-relation découle de l'énonciation et de la spécification du problème, le modèle relationnel découle de la spécification et du modèle entité-relation, l'implémentation des requêtes découle de la spécification du problème et du modèle relationnel, et l'optimisation découle des requêtes et du problème. Les incohérences entre ces étapes seront pénalisées.

Une annexe est un ajout au rapport, dont le contenu peut être utile en référence mais n'est pas strictement nécessaire pour la compréhension du rapport. En d'autres mots, votre rapport doit nécessairement présenter le contenu essentiel de votre projet, alors que le contenu offrant des détails supplémentaires intéressants mais non-essentiel et dont l'inclusion dans le rapport pourrait le rendre trop lourd, compliqué, ou long, peut aller en annexe. Par exemple, si vous avez un tableau de résultats détaillés de plusieurs pages, vous pouvez présenter un résumé ou des lignes importantes pour votre discussion dans le rapport et inclure le tableau complet en annexe. À l'inverse, une annexe n'est pas une manière de contourner la limite de pages en y déplaçant le matériel qui devrait être dans votre rapport. Une bonne règle pour décider est de vous poser la question, « est-ce que mon rapport serait complet si j'éliminais complètement ce contenu? » Si oui, c'est du matériel pour l'annexe, et si non c'est du matériel qui va dans votre rapport.

Les explications et justifications sont à prioriser sur des extraits de code ou de pseudo-code. Ce contenu donne une bien meilleure idée de ce que vous avez réalisé et pourquoi. Des exemples de code ou de pseudo-code sont utiles pour illustrer une explication d'un algorithme complexe. Par contre le code doit être écrit et placé dans un cadre numéroté comme une figure, et limité aux lignes importantes pour l'explication. Un copier-coller de votre code en entier ne fait pas office d'explication et ne sera pas évalué. Une saisie d'écran du code est contre-indiquée: ça ne donne jamais un résultat visuellement plaisant et c'est souvent difficile à lire, et ce sera pénalisé.

Soumission du code

Pour la remise de ce projet, vous devez fournir une archives zip comprenant tout le code (SQL, Python, HTML, JavaScript, CSS) que vous avez écrit, et uniquement le code que vous avez écrit (excluant donc complètement les bibliothèques externes). Ce code sera donc non-fonctionnel, mais permettra une évaluation plus approfondie de votre travail. N'oubliez pas le code SQL de création des relations et des requêtes! Le code sera utilisé en complément avec votre rapport pour évaluer l'implémentation des fonctionnalités de votre système.

Démo vidéo

Vous devez réaliser une vidéo démo de votre projet, dans laquelle vous présentez votre projet et démontrez l'utilisation de votre système. Cette vidéo doit être de 15 minutes. Il n'est pas nécessaire que tous les membres de l'équipes soient entendus dans la vidéo. Vous pouvez la

téléverser sur le site de remise de MonPortail avec votre rapport, ou sur un site d'hébergement vidéo de votre choix en fournissant un lien avec votre rapport (n'oubliez pas d'ajuster les droits d'accès pour que les correcteurs puissent la visionner!)

Pour l'aspect de présentation du projet, vous pouvez passer, point par point, à travers la liste d'items de réalisation du projet. Pour l'aspect utilisation du système, vous devez démontrer que toutes les fonctionnalités de votre système fonctionnent bien tant pour une utilisation normale qu'une utilisation anormale (i.e. erreurs d'entrées accidentelles ou délibérées).

Il existe une variété d'outils gratuits qui permettent d'enregistrer une saisie de votre écran durant l'utilisation de votre ordinateur avec un flux d'entrée audio et/ou vidéo. Vous pouvez utiliser celui de votre choix. Si vous désirez une suggestion, j'utilise personnellement l'outil OBS Studio.

Bonus et Pénalités

Le rapport technique et la vidéo doivent être d'une qualité digne des ingénieur.e.s que vous devenez. Je réserve donc le droit de retirer jusqu'à 20% des points en pénalité pour un rapport et/ou une vidéo de mauvaise qualité. Ceci inclut particulièrement les rapports présentant des fautes d'orthographe et de grammaire, des figures mal préparées (ce qui inclus, mais n'est pas limité à, des saisies d'écran mal cadrées, des photos de contenu texte peu lisible, des graphiques avec des filigranes, des figures dessinées à la main), des irrégularités de mise en page (variations de polices, tailles de caractère, marges, etc.), les textes incohérents, les rapports soumis en document Word (ou tout autre format que PDF). Pour la vidéo, je pénaliserais un langage inapproprié ou un enregistrement de mauvaise qualité (incluant de vous filmer avec votre cellulaire tenu à la main). Dans les deux cas je pénaliserais le non-respect des longueurs maximales.

Pour une remise en retard sans justification acceptable et préalablement approuvée, une pénalité de 3% par jour sera appliquée, jusqu'à un maximum d'une semaine. Un projet remis avec plus d'une semaine de retard recevra la note de 0.

À l'inverse, je peux donner des points bonus à ma discrétion pour récompenser des étudiants qui intègrent des fonctionnalités supplémentaires dans leur système (ex. : envoi de courriel de confirmation, utilisation de vrais services en ligne, importation de données de vraies BD organisationnelles, etc.). Ces fonctionnalités ne sont pas demandées dans le projet et ne sont pas nécessaires, mais l'effort supplémentaire pour ajouter une touche de réalisme au projet sera récompensé.

Évaluation par les pairs

En plus du rapport d'équipe, chaque étudiant doit remettre une évaluation de son travail et du travail de ses co-équipiers à travers la section formative du site web du cours. Cette évaluation est soumise séparément du rapport, de manière confidentielle, en utilisant la soumission d'évaluations formatives sur le site web du cours.

Cette évaluation prend la forme d'un simple document texte donnant, pour chaque membre de l'équipe (incluant soi-même), les trois lignes suivantes : le nom du membre de l'équipe, l'évaluation, et une brève justification de l'évaluation.

L'évaluation peut prendre une des quatre valeurs suivantes : « satisfaisante », « insatisfaisante », « minimale », ou « nulle ». Comme règle du pouce, un coéquipier contribuant sa juste part du travail (environ un tiers du travail pour une équipe de 3) a une contribution « satisfaisante », un coéquipier contribuant bien moins que sa juste part a une contribution « insatisfaisante », un coéquipier qui traîne et ne travaille que le strict minimum lorsque ses coéquipiers le force a une contribution « minimale », et un coéquipier qui ignore ses collègues et disparaît de l'équipe a une contribution « nulle ».

Un étudiant recevant une évaluation « insatisfaisante » de la majorité de ses co-équipiers recevra une pénalité de 25% de la note, un étudiant recevant des évaluations « insatisfaisante » et « minimale » recevra une pénalité de 34% de la note, et un étudiant recevant l'évaluation « minimale » de la majorité de ses co-équipiers recevra une pénalité de 50% de la note. Un étudiant recevant l'évaluation « nulle » recevra la note de 0 et ses coéquipiers doivent exclure son nom du rapport, afin de refléter le fait qu'il n'y a pas participé du tout au projet.

L'enseignant se réserve le droit de substituer son propre jugement à celui des pairs advenant que leurs évaluations ne semblent pas conformes aux faits.

Soulignons que la participation au projet se doit d'être technique, et ne peut pas être limitée à la rédaction du rapport. L'évaluation des pairs se concentre sur les aspects techniques du travail réalisé, et un étudiant qui n'a fait aucune contribution technique a une participation nulle et une note de 0, même s'il a contribué à la rédaction du rapport ou l'enregistrement de la vidéo.

Cette évaluation doit être remise individuellement par chacun des membres de l'équipe. Une personne ne remettant pas son évaluation sera présumée de se donner une auto-évaluation de « minimale » et une évaluation de « satisfaisante » à tous ses coéquipiers.

Plagiat

Le plagiat est une offense académique sérieuse. Tout étudiant qui tente de soumettre un travail qui n'est pas le sien sera pénalisé. Ceci inclut de copier le travail ou rapport d'un autre étudiant du cours, d'un étudiant d'un autre cours, d'un étudiant du cours d'une année précédente (incluant

soi-même en cas de reprise du cours), ou un travail trouvé ailleurs sur internet. Un étudiant coupable de plagiat recevra automatiquement la note de zéro pour le projet entier et s'exposera à d'autres sanctions telles que décidées par l'Université.

Conseils

- Il est fortement conseillé que tous les co-équipiers travaillent sur tous les aspects du système. Ceci vous donnera la meilleure expérience d'apprentissage, et vous préparera au maximum aux autres évaluations (le questionnaire SQL et les examens).
- La description du rapport à soumettre inclut les points alloués à chaque élément du travail. Tenez-en compte pour planifier la quantité d'effort à mettre pour chaque élément du projet. Par exemple, il n'est pas normal de passer plus de temps à développer un aspect valant 1 point qu'un valant 3 points.
- La réalisation de ce projet peut facilement prendre une envergure démesurée et consommer une majorité de votre temps cette session. Ne perdez donc pas de vue que le projet vaut 25% de votre note du cours, et ne négligez pas les autres aspects du cours (ni vos autres cours). Fixez-vous des objectifs réalistes et réalisables avec un effort raisonnable. Une fois de plus, la liste de points alloués à chaque aspect du travail peut vous aider dans cette planification de votre projet.

Ressources externes

Vous connaissez peut-être une ressource externe (une plateforme de développement JavaScript ou un modèle standard CSS ou un serveur d'application C++, etc.) que vous aimeriez utiliser dans votre projet. Ceci est acceptable, étant donné certaines conditions. Premièrement, gardez en tête le règlement sur le plagiat : toute tentative de remettre un travail que vous n'avez pas fait comme si c'était le vôtre est une infraction académique majeure. En d'autres mots, votre rapport doit indiquer clairement quelle partie du projet vous avez réalisé et quelle partie a été empruntée d'une autre source, et la division entre les deux ne doit laisser aucune place à l'ambiguïté. Deuxièmement, les points du projet sont donnés pour le travail que vous avez réalisé, et non pour l'intégration du travail de tiers. En d'autres mots, si vous utilisez une ressource externe pour réaliser un des objectifs évalués du projet sans contribution de votre part, vous aurez une note de zéro sur cet objectif, comme si vous ne l'aviez pas fait du tout. De même, aucuns points bonus ne sont donnés pour l'intégration de ressources externes au projet.

Le code démo du chapitre du cours sur l'architecture trois-niveaux est disponible pour téléchargement à travers le site web du cours. Vous pouvez l'utiliser comme point de départ pour votre projet, au même titre qu'une autre ressource externe.

Il n'est pas permis d'utiliser un ORM ou un autre logiciel de création de BD. Quoique ces logiciels sont utilisés et même recommandés en pratique, l'objectif du cours est l'apprentissage

du SQL et de l'implémentation de bases de données, et l'utilisation d'un logiciel qui réalise automatiquement ceci pour vous va donc à l'encontre des objectifs du cours.

Vous pouvez vouloir ajouter des fonctionnalités réalistes à votre site en appelant une API d'un service externe (ex. : compléter le site de votre boutique en ligne en appelant PayPal pour faire des paiements). Cette touche de réalisme n'est pas obligatoire, et n'est pas comptabilisée dans les critères de correction. Vous pouvez donc remplacer ces APIs par des étapes-bidons dans votre code pour les simuler (ex. : une fonction PayPal dans votre code qui accepte toutes les transactions sans contacter la vraie API). Par contre, l'utilisation des vraies API peut valoir des points bonus, et sera un critère considéré dans la sélection d'un projet gagnant du Prix Pierre-Ardouin (voir plus bas).

Formation des équipes

Ce projet doit obligatoirement être réalisé en équipe. Un projet individuel n'est pas accepté et sera pénalisé.

Vous être responsable de former vos propres équipes. Parlez-vous, faites-vous des amis et travaillez ensemble. Je ne vais en aucun cas assigner arbitrairement un individu à une équipe ni forcer une équipe à accepter un individu orphelin.

Un forum pour la recherche de coéquipiers est disponible sur le site web du cours pour vous faciliter cette tâche. Je vous recommande de l'utiliser pour expliquer vos contraintes personnelles (emplois, enfants, etc.) et vos préférences (travail à distance ou en personne, jour nuit ou fin de semaine, objectif d'obtenir un A ou un C, etc.) et ainsi trouver des partenaires compatibles.

En cas d'abandon d'un ou plusieurs membres d'une équipe, les membres restants peuvent en tout temps ajouter de nouveaux coéquipiers à leur équipe, ou abandonner leur équipe pour en joindre une autre.

Prix Pierre-Ardouin

« Depuis plusieurs années, le Département d'informatique et de génie logiciel a mis en place un concours récompensant l'équipe qui produit le meilleur TP/projet dans le cadre d'un cours. Ces travaux de session ont l'envergure d'un mini-projet qui est admissible par rapport aux normes fixées par le département. À la suite des évaluations des travaux, l'enseignant du cours détermine l'équipe gagnante; chaque membre de l'équipe gagnante reçoit alors un certificat-cadeau d'une valeur de 50\$ ainsi qu'une attestation remise par le département à la fin de l'année universitaire.

De plus, le Département d'informatique et de génie logiciel a mis en place une prix élite, appelée Prix « Pierre Ardouin », qui vise à récompenser le meilleur projet, tous cours participants confondus. Deux principaux critères guident le choix des évaluateurs dans l'identification du lauréat: l'excellence du travail (par rapport à ce qui est demandé dans l'énoncé) et l'aspect créativité/innovation. Il est actuellement prévu un certificat-cadeau de 200\$ pour récompenser chaque membre de l'équipe « élite » gagnante (pour un maximum de 1000\$ pour toute l'équipe). Aussi, le département veille à publier l'information sur le site web.

Notez que depuis l'automne 2019, le concours récompense les meilleurs travaux d'une année (été, automne et hiver) selon que les cours participants sont de 1ère, de 2e année, de 3e/4e année, ou de 2e/3e cycle; ainsi, pour chacune de ces catégories de cours, un prix sera remis aux finalistes ainsi qu'aux lauréats du prix Pierre Ardouin (donc, potentiellement 4 équipes lauréates par année).

À la fin du mois de mai de chaque année universitaire, le département organise une cérémonie pour honorer les finalistes et les lauréats du prix « Pierre Ardouin » de l'année qui vient de s'écouler (été, automne et hiver), et pour leur remettre une attestation. Les attestations peuvent être transmises par courriel également à ceux et celles qui ne seraient pas présents à la cérémonie. »

Pour ce cours, le projet sera choisi parmi ceux ayant répondu à toutes les exigences d'évaluation et ayant reçu une excellente note. Parmi ceux-ci le projet choisi sera celui qui se démarque en allant au-delà des exigences du cours, par exemple en implémentant des fonctionnalités sophistiquées, en appelant les APIs de vrais services en-ligne, en utilisant des données réelles, ou en collaborant avec un partenaire réel. Ce sera un projet qui pourrait, avec un minimum de modifications, devenir un service web réel. Le rapport et la vidéo démo du projet gagnant seront diffusés sur le site du Département.

Si vous voulez que votre projet soit considéré pour le Prix Pierre-Ardouin, vous devez fournir, en même temps que la soumission de votre projet, un bref texte (maximum une page) expliquant en quoi votre projet se démarque et va au-delà des exigences. Profitez-en également pour donner

un nom original à votre projet (s'il gagne, ça paraîtra mieux sur le site web que « projet de GLO-2005 »). Je rappelle que cette participation est entièrement optionnelle et à votre discrétion; elle n'affecte en rien votre note dans le projet ni le cours.

Liste de soumission

- ☐ Rapport de projet
- ☐ Archive du code que vous avez écrit
- ☐ Vidéo démo
- ☐ 1 évaluation des pairs par membre de l'équipe
- ☐ Texte de participation au Prix Pierre-Ardouin (optionnel)