

M1 Informatique – UE Projet

Carnet de bord : les coulisses de la recherche documentaire

Les éléments que vous indiquez dans ce carnet donneront lieu à une notation

Noms, prénoms et spécialité :

Fazazi Zeid
Dibassi Brahim
Luo Yukai

Sujet :

Interface Web pour Autobill

Consigne :

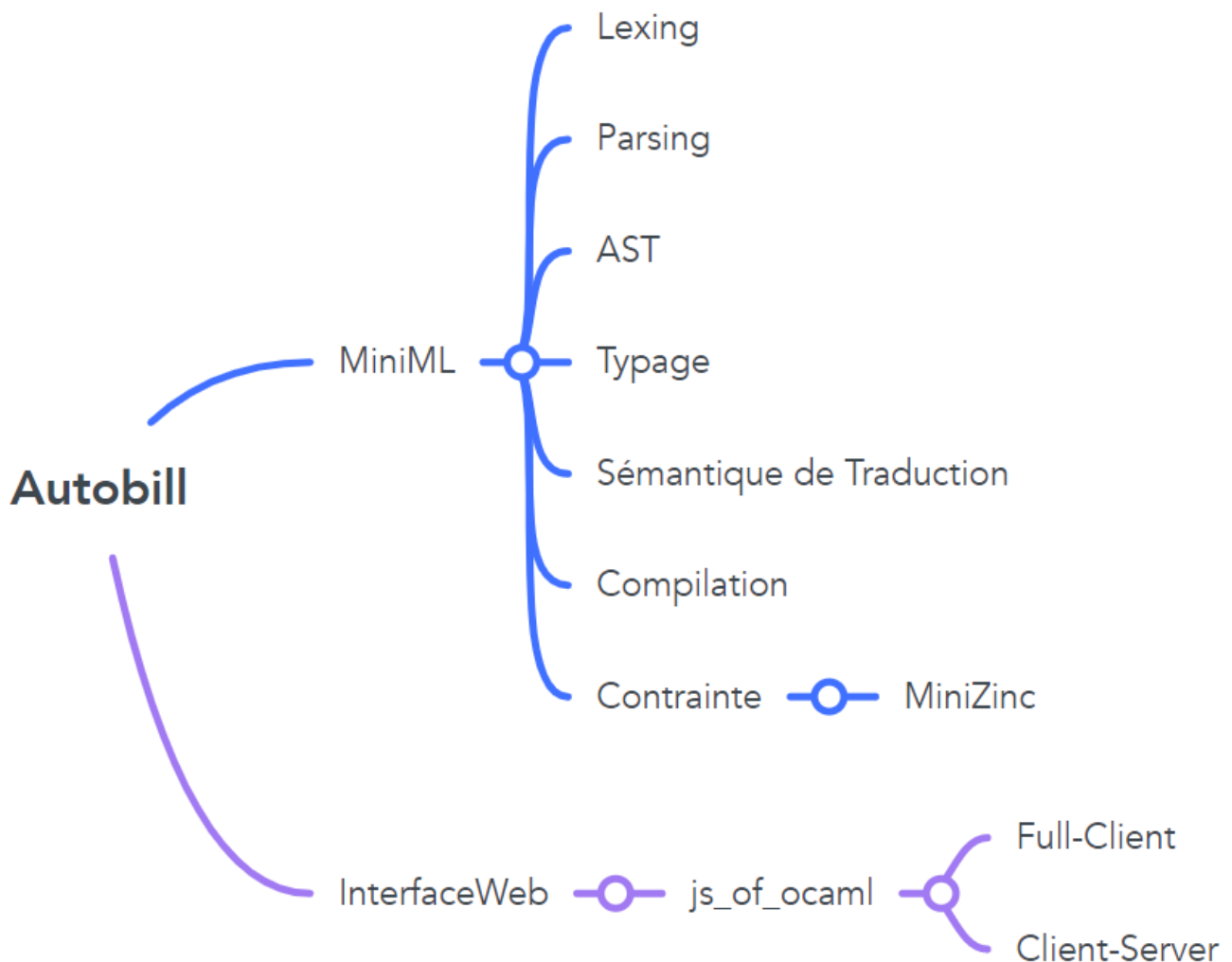
1. **Introduction (5-10 lignes max)** : Décrivez rapidement votre sujet de recherche, ses différents aspects et enjeux, ainsi que l'angle sous lequel vous avez décidé de le traiter.
2. **Les mots clés retenus** : Listez les mots-clés que vous avez utilisés pour votre recherche bibliographique. Organisez-les sous forme de carte heuristique.
3. **Descriptif de la recherche documentaire (10-15 lignes)** : Décrivez votre utilisation des différents outils de recherche (moteurs de recherche, base de donnée, catalogues, recherche par rebond etc.). Comparez ensuite les outils entre eux. A quelles sources vous ont-ils permis d'accéder ? Quelles sont leurs spécificités ? Quel est leur niveau de spécialisation ?
4. **Bibliographie produite dans le cadre du projet** : Utilisez la norme ACM.
5. **Evaluation des sources (5 lignes minimum par source)** : Choisissez 3 sources parmi votre bibliographie, décrivez la manière dont vous les avez trouvées et faites-en une évaluation critique en utilisant les critères vus sur les supports de TDs.

Introduction :

Notre recherche documentaire portera sur des sujets qui englobent notre projet PSTL : ce dernier consiste en la création d'une interface Web prenant la forme d'un environnement de développement, dans lequel on peut soumettre du code "MiniML" à une librairie nommée Autobill. Cette dernière va expliciter la quantité de ressources à allouer au programme soumis sous la forme dans une équation qui, une fois passé à un solveur, sera résolu.

Ainsi, notre recherche viendra soutenir le développement de notre solution afin de nous aider à déterminer les technologies, architectures et outils à employer, les avantages et contreparties de certains choix de conception ainsi que la méthodologie à suivre pour la documentation de notre projet.

Mot clé retenu :



Descriptif de la recherche documentaire:

On va distinguer deux sources majeures d'information qui nous seront utiles:

- Les sources issues d'Internet, notamment pour des documentations et des livres en ligne
- Les sources universitaires, comme des articles ou des références bibliographiques de certaines de nos cours.

Chaque source intervient dans des points clés du projet. Les sources d'Internet sont variées et faciles d'accès et informe sur la manière d'employer les outils et technologies utilisés dans ce projet via les documentations étoffées et les exemples qui les accompagnent. On les trouve facilement depuis n'importe quel moteur de recherche.

Quant aux sources universitaires, elles servent d'appui pour notre développement et notre méthodologie. On distingue les sources universitaires référencées / issues des cours de notre Master, cours ayant grandement contribué à et les sources universitaires "libres" issues de notre propre initiative ou des suggestions de nos tuteurs académiques. Dans les deux cas, on utilise la base de données Primo BSU pour la recherche de ces sources, celle-ci redirigent souvent vers des banques d'articles comme [l'ACM](#) (Association for Computing Machinery), le [Web of Science](#) grâce aux accès octroyés par Sorbonne Université ou des plateformes reconnues comme [HAL](#) (Hyper Article en Ligne) ou le Cambridge Core

Bibliographie produite dans le cadre du projet

Will Kurt. 2018. Get Programming with Haskell. Simon and Schuster. Chapitre 5,7

Pierce, Benjamin C. Types and Programming Languages. MIT Press, 2002, [Web](#)

[1] Levy, Paul Blain. "Call-by-Push-Value: A Subsuming Paradigm." Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999. 228–243. [Web](#).

Winskel, Glynn. The Formal Semantics of Programming Languages : an Introduction. Cambridge (Mass.) London: MIT Press, 1993. [Web](#).

[2] *Compilers : Principles, Techniques, and Tools*. 2nd ed. Boston (Mass.) San Francisco (Calif.) New York [etc: Pearson Addison Wesley, 2007, [Web](#)

Minsky, Anil Madhavapeddy, and Jason Hickey. 2013. *Real World OCaml*. O'Reilly Media. [Web](#)

Martin Avanzini and Ugo Dal Lago. 2017. Automating sized-type inference for complexity analysis. *Proceedings of the ACM on Programming Languages* 1, 1-29, [Web](#)

Hoffmann, Jan, and Steffen Jost. "Two Decades of Automatic Amortized Resource Analysis." Mathematical structures in computer science 32.6 (2022): 729–759. [Web](#).

[3] Dominic Orchard, Vilem-Benjamin Liepelt, and Harley Eades III. 2019. Quantitative program reasoning with graded modal types. *Proceedings of the ACM on Programming Languages* 3, ICFP (2019), [Web](#)

Xavier Leroy. 2022 OCaml library. Ocaml Lazy Doc. Retrieved February 20, 2023 from <https://v2.ocaml.org/api/index.html>

Haskell - Wikibooks, open books for an open world. Doc Haskell. Retrieved February 17, 2023 from <https://en.wikibooks.org/wiki/Haskell>

Evaluation des sources

[1]

Il s'agit ici d'un compte rendu d'une présentation de Levy Paul Blain, à propos du paradigme Call-by-Push-Value lors de la quatrième conférence internationale centrée sur le lambda Calculs typé et ses applications en 1999.

Levy, Paul Blain, professeur et expert reconnu dans le domaine de la sémantique des langages de programmation y présente ses travaux basés sur des preuves et expérimentations solides à ses pairs.

Cependant il est important de noter qu'il s'agit ici d'une des premières publications sur Call-By-Push Value.

Sous les conseils de nos professeurs, nous avons dû recouper cette dernière avec d'autres sources mises à jour et plus simples d'accès du même auteur.

[2]

Ce livre fait partie des références que l'on peut retrouver à la base de nombreux cours de nos modules, dans le Master d'Informatique, traitant de sujets autour de la compilation et de l'analyse statique.

C'est un ouvrage de référence en informatique dans le domaine de la compilation, écrit par Alfred V. Aho, Ravi Sethi et Jeffrey D. Ullman, des informaticiens-chercheurs issus des Laboratoires Bell, en 1986. Il est édité par Addison-Wesley, un éditeur historique de littérature informatique américain.

Il traite de tous les aspects scientifiques et techniques de la construction de compilateurs. Son ancienneté et son statut de référence dans le milieu fait qu'il est cité dans des nombreux ouvrages scientifiques encore aujourd'hui et sert à la transmission de connaissances pour des élèves du cycle Master.

Il est cependant très fourni et aborde un grand nombre de sujets qui dépassent la portée de notre projet, il en reste tout de même pertinent pour l'usage qu'il est en fait dans les supports de cours.

[3]

L'article "Quantitative Program Reasoning with Graded Modal Types" publié par Dominic Orchard, Vilem-Benjamin Liepelt et Harley Eades III dans "Proceedings of the ACM on Programming Languages 3, ICFP" en 2019. Les auteurs de l'article, Dominic Orchard, Vilem-Benjamin Liepelt et Harley Eades III, sont des chercheurs en informatique spécialisés dans la sémantique des langages de programmation et la vérification de programmes. L'article présente une nouvelle méthode de raisonnement quantitatif pour la vérification des programmes informatiques, utilisant des types modaux gradués. Il peut beaucoup nous aider en typage et traduction de notre projet. Après notre analyse, nous pensons que cette ressource est assez fiable.