



Track 1 : Amphi

- 08:00 Ouverture
- 09:00 **Keynote d'ouverture** Nicolas Le Pochat
- 09:15 Pause
- 09:25 **Short track** J'ai plié mon smartphone et il ne s'est pas cassé... mais ma web app si :((Yohan Lasorsa)
- 09:50 Pause
- 10:00 **Conférence** Comment Jooq m'a réconcilié avec SQL dans une application Java (Xavier Calland)
- 10:45 Pause
- 11:15 **Conférence** Angular 14: The next generation of Angular Application (Nicolas Frizzarin)
- 12:00 **Repas**
- 13:30 **Conférence** Appwrite est-il prêt à éteindre Firebase ? (Alves Mickaël / Lucas Audart)
- 14:15 Pause
- 14:25 **Short track** Ansible: un bac à sable en local avec docker (Emile Chornton)
- 14:50 Pause
- 15:20 **Conférence** Rust, ou le langage pour les 10 ans à venir (Adrien Gras)
- 16:05 Pause
- 16:15 **Conférence** Railway Programming: la voie vers un code plus honnête (Romain Berthon / Sylvain Coudert)
- 17:00 Pause
- 17:10 **Keynote de fin** Adrien Gras

Track 2 : Salle TD



- Short track** Méthodologie DevOps : Évolutivité et agilité d'une solution grand public (Camille Darnet)
- Conférence** Automatisation des test (Fernanda Campillo / Frédéric Baralla)
- Conférence** Une touche de chaos pour plus d'efficacité (Yohan Lasorsa)
- Conférence** Développeur, ta prod tu respecteras ! (Emile Chornton / Nicolas Silvain)
- Short track** Le métier de dev expliqué à tes grands-parents... (Sylvain Coudert)
- Conférence** Devenir un pro de l'UX en moins d'une heure (Nicolas Frizzarin)
- Short track** Design To Cost : Partir du coût et du design vers la solution technique (Nicolas Le Pochat)
- Short track** J'ai pactisé avec le diable : j'ai fait carrière dans une ESN (Jérémy Colombet)



Conférences =)



Track 1 : Amphi

Ansible: un bac à sable en local avec Docker

[Short track] Emile Chomton

Ansible, célèbre outil de gestion de la configuration est utilisé un peu partout.

Dans une optique de formation et de démonstration, nous avons besoin d'environnements bac à sable et nous sommes rapidement confrontés à un problème: comment démontrer, tester ou faire tester Ansible facilement ?

Nous avons besoin d'une solution simple, facile à mettre en place, transportable et utilisable par tous, avec un minimum de connaissance...

Il est fortement déconseillé de cibler son poste avec ces premiers playbooks Ansible. On pourrait utiliser une VM sur notre poste ou hébergée dans un cluster quelconque, mais cela peut vite devenir lourd ou cher.

Dans ce quickie, sera donc présentée une solution alternative: utiliser docker ! Docker nous servira à installer et faire tourner ansible mais sera aussi utile pour simuler un petit parc de d'hôtes ansible.

Nous verrons donc les avantages, comment le mettre en place et bien-sûr, les limites à ce système

Comment Joq m'a réconcilié avec SQL dans une application Java

[Conférence] Xavier Calland

Pendant nos études, on nous apprend à faire du SQL (modéliser, requêter, etc), cela reste souvent une approche théorique dans le sens où on accède à la BDD directement depuis un client SQL.

Mais qu'en est-il dans la vraie vie, une fois que les accès se font pour/par une application. Est-ce que SQL est toujours aussi sympa et répond aux attentes des développeurs ?

JDBC est efficace pour le bas niveau, mais c'est léger pour la maintenabilité, on renomme un champ en base et c'est le drame.

Les ORM nous éloignent de SQL et s'accompagnent, en partie, d'une perte de contrôle.

Mais alors, est-il possible de faire du SQL dans une application Java avec des objectifs de qualité et maintenabilité habituels dans le développement logiciel ? En résumé, peut-on faire du SQL robuste ?

Oui, c'est possible, on peut garantir, au build, que les requêtes SQL écrites dans une application sont 100% fonctionnelles. Fini les surprises au runtime : pas de typos dans les champs, pas de problèmes de type de données, ...

Avec comme contexte une application Java, je retrace mon périple à la recherche de l'approche ultime (c'est-à-dire qui répond à mes besoins) pour faire du SQL comme on peut l'attendre en 2022.

On part de JDBC pour arriver à JOOQ après un passage par les ORM, avec chaque étape des découvertes et des déconvenues...





Angular 14: The next generation of Angular Application

[Conférence] Nicolas Frizzarin

Angular 14 est sorti en juin 2022, et cette version annonce que les modules Angular deviennent optionnels.

Whaoouuu, jusque maintenant les modules étaient le concept de base d'une application Angular. Toute l'architecture d'une application Angular gravitait autour de ce concept. En résumé, les modules étaient une pièce centrale de nos applications.

Depuis la sortie de la version 14 d'Angular, ce sont les composants qui occupent ce rôle. Ces composants spécifiques sont appelés: "Standalone Components"

Cependant cette version apporte bien d'autres nouveautés, toutes très intéressantes comme:

- la composition par injection
- les formulaires typés
- une nouvelle façon de gérer les titres de nos SPA

Comment ces nouveautés vont changer notre façon d'architecturer nos applications ? Comment les intégrer dans nos applications existantes ? Les modules vont-ils devenir dépréciés ?

L'objectif de ce talk est d'expliquer tous les concepts de cette nouvelle version et de donner des usages sur quand et comment les utiliser et les intégrer dans des applications existantes.



Appwrite est-il prêt à éteindre Firebase ?

[Conférence] Alves Mickaël / Lucas Audart

Est-ce que Firebase vous dit quelque chose ? Vous en avez sûrement entendu parlé et vous l'avez peut-être déjà utilisé, et pour le coup, c'est normal ! Voilà maintenant plusieurs années que de nombreux développeurs l'utilisent pour faciliter la création de back-end scalable et performant.

Mais avez-vous déjà entendu parler d'Appwrite ?

Peu importe votre réponse, venez découvrir avec nous le duel entre Appwrite la jeune solution open-source et Firebase la flamme de Google.

J'ai plié mon smartphone et il ne s'est pas cassé... mais ma web app si :(

[Short track] Yohan Lasorsa

Vous avez sûrement dû les voir passer, les smartphones à écrans pliables comme le Samsung Galaxy Fold ou avec deux écrans comme le Surface Duo sont déjà sur le marché, il ne reste "plus" qu'à repenser et mettre vos applications à jour ! Mais comment ? Par où commencer ? Heureusement, le web évolue en permanence et a déjà commencé à s'adapter à ces nouveaux périphériques grâce à de nouvelles APIs. Dans ce talk on va explorer pourquoi et surtout comment exploiter plusieurs écrans dans vos web apps, pour créer des nouvelles expériences via des fonctionnalités encore expérimentales. On va aussi regarder comment adapter des applications existantes pour ces nouveaux appareils, tout en essayant d'éviter au maximum d'avoir affaire avec les bizarreries des nouvelles primitives CSS et JavaScript.

Rust, ou le langage pour les 10 ans à venir

[Conférence] Adrien Gras

Depuis sa version 1.0 sorti il y a maintenant 7 ans, Rust est un langage qui ne cesse de croître en popularité et en utilisation. Backend, web, réseau, informatique embarquée, crypto ; il est pratiquement possible de tout faire avec ce langage aux performances égalant celles du C et à la sécurité renforcée.

Je vous invite avec moi pour un voyage où nous irons découvrir ce langage nouvelle génération, ses utilisations possibles, ses points forts et points faibles, et quelques histoires concrètes de son adoption par de grandes figures du numérique.



Railway Programming: la voie vers un code plus honnête

[Conférence] Romain Berthon / Sylvain Coudert

Le code auquel nous sommes pour la plupart confrontés comporte beaucoup d'informations implicites (erreurs, exceptions, nullables, ...). Celles-ci se montrent problématiques car elles nécessitent des connaissances poussées de la codebase et génèrent une charge cognitive supplémentaire. Malheureusement, tout ceci augmente la probabilité que les développeurs fassent des erreurs. Au cours de ce talk, exemples à l'appui, nous vous montrerons comment rendre plus explicites ces comportements cachés à l'aide de votre système de types. Nous mettrons ensuite en lumière les problématiques qui en découlent et comment les résoudre.



Track 2 : TD



Méthodologie DevOps : Évolutivité et agilité d'une solution grand public



[Short track] Camille Darnet

Nous évoquerons les différentes phases d'évolution et d'extension de nos portails de services en ligne depuis plus de 10 ans. Ces phases répondent à nos exigences de qualité de services envers les usagers mais aussi aux problématiques d'infrastructures, techniques, de sécurité...Le tout est orchestré de manière agile en se basant sur des principes d'architectures, qui nous ont permis d'avoir une solution à jour et pérenne dans un contexte spécifique.

Automatisation des test

[Conférence] Fernanda Campillo / Frédéric Baralla

Comment assurer la qualité des livraisons via l'automatisation des Tests de non régression ? Comment détecter rapidement les anomalies et libérer les testeurs des exécutions répétitives ?

L'objectif est de partager la problématique sur des tests répétitifs à faire lors de chaque livraison d'une nouvelle version du code, puis de partager des solutions et des retours d'expérience.



Une touche de chaos pour plus d'efficacité

[Conférence] Yohan Lasorsa

Quel est le point commun entre un jeune parent et un logiciel open source? A priori pas grand-chose, et pourtant dans les deux cas survivre au milieu du chaos quotidien vous forcera à être plus efficace sur la gestion des tâches.

De l'onboarding au déploiement, les tâches courantes du cycle de dev sont rarement optimisées dans les projets d'entreprise et consomment énormément de temps et d'énergie! Mais alors comment font les projets open source pour délivrer des mises à jour régulières et du code de qualité sur une base de volontariat? Une partie du secret réside dans la nécessité née du chaos.

Avec ma triple casquette de papa, mainteneur open source, et développeur en entreprise, je vais partager mes retours d'expériences, les erreurs que j'ai faites et tout ce que j'ai pu en apprendre au profit de la priorisation, la gestion du temps et l'outillage de différents projets.

Développeur, ta prod tu respecteras !

[Conférence] Emile Chomton / Nicolas Silvain

Faire une application de qualité c'est bien mais pouvoir la maintenir en prod c'est mieux.

Faire des logs pertinents, surveiller sa prod et ses métriques est souvent vu comme une tâche ingrate que personne ne veut faire et donc n'est pas fait correctement.

Avec cette histoire de projet qui pourrait être la vôtre, nous allons vous raconter la vie d'une application, de la préparation à sa mise au monde et du suivi de sa croissance. Nous aborderons ce qu'il faut faire ou éviter, les préparatifs qui ont été nécessaires ainsi que ce qui l'aide à se maintenir en bonne santé.

Voici donc l'histoire (presque inspirée de faits réels) de l'application HAMS-TERGRAM. HamsterGram est une sorte d'instagram réservé aux photos et stories de hamster, développée par une équipe de choc: deux développeurs, Caroline et Cherriff ainsi qu'un ops; Jean-Cloud.





Le métier de dev expliqué à tes grands-parents...

[Short track] Sylvain Coudert

Comment décrire et expliquer votre métier à vos grands-parents ? A vos enfants ? A votre manager ? Comment leur faire envisager (sans aller jusqu'à comprendre) que le métier de dev ne se résume pas à empiler des lignes de codes sur un écran à fond noir ? Comment nous-même apprendre à faire nôtre la définition de notre métier par ses différentes facettes ? D'ingénieur à pompier, en passant par linguiste ou architecte, la créature développante doit régulièrement changer de forme ou de peau. Petit tour d'horizon de celles-ci pour vous aider à optimiser votre élevage et ainsi donner à vos devs l'œil brillant et le poil soyeux !



Devenir un pro de l'UX en moins d'une heure

[Conférence] Nicolas Frizzarin

Lorsque l'on développe une application web, il est primordial de respecter un certain niveau d'exigence sur l'UX. Cette pratique permet de satisfaire l'utilisateur mais également de s'assurer qu'il prenne plaisir à utiliser notre application.

En 2020, Google annonce vouloir intégrer, l'année suivante, dans ces outils de performances et de métriques, trois nouveaux signaux permettant de caractériser et de mesurer cette expérience utilisateur.

Ces signaux sont regroupés sous le nom de Core Web Vitals et sont les suivants:

LCP (Largest Contentful Paint)

CLS (Cumule Layout Shift)

FID (First Input Delay)



Mais que signifie concrètement ces métriques ? Comment les déboguer et les optimiser et ainsi rendre notre application agréable à utiliser ?

A travers ce talk, je répondrai à ces questions et présenterai également un quatrième signal encore en cours d'expérimentation: INP (Interaction to Next Point)

Design To Cost : Partir du coût et du design vers la solution technique

[Short track] Nicolas Le Pochat

L'approche du produit par la notion de coût et d'apport fonctionnel est une méthode inventée dans les années 90 par Renault qui a été répandue et est devenue une référence d'optimisation des projets.

Cette approche est aujourd'hui fortement utile dans le développement d'applications web.

Petit horizon de présentation !

J'ai pactisé avec le diable : je fais carrière en ESN

[Short track] Jérémy Colombet

ESN, ce terme fait frémir n'importe quel développeur junior. "Tu ne veux quand même pas devenir de la chair à commercial ?" me disaient les uns, "Tu n'arriveras pas à faire ce que tu aimes" me disaient les autres... Pourtant, j'ai tenu. 13 ans même. Aujourd'hui si je propose ce talk, c'est pour redorer la toison de sociétés qui sont mal vues, pour transmettre mon retour d'expérience sur comment j'ai su faire carrière en ESN et comment je me suis épanoui dans ce monde étrange.

