



ATHSUE

Athsue

PPE 2

Présentation

Contexte

GSB

- Le laboratoire Galaxy Swiss Bourdin (GSB) est issu de la fusion entre le géant américain Galaxy (spécialisé dans le secteur des maladies virales dont le SIDA et les hépatites) et le conglomérat européen Swiss Bourdin (travaillant sur des médicaments plus conventionnels), lui-même déjà union de trois petits laboratoires. L'entreprise appartient à l'industrie pharmaceutique dans laquelle les regroupements de laboratoires sont fréquents. Elle constitue le lieu d'échange entre laboratoires et praticiens.

M2L

- La M2L est une structure financée par le Conseil Régional de la Lorraine dont l'administration est délégué au Comité Régional Olympique et Sportif de la Lorraine. (CROSL). La M2L « Maisons de ligues de la Lorraine » a pour mission de fournir des espaces et des services aux différentes ligues sportives régionales et d'autres structures hébergées.

Mise en situation

CONTEXTE Problématique M2L

PROBLEMATIQUE 1 :

- La Maison de Ligue de la Lorraines nous contact pour savoir s'il est possible de faire une présentation de la M2L mais aussi permettre aux clubs sportifs de s'inscrire et d'organiser des entraînements et / ou match.

« Bonjour, nous sommes La Maison de Ligues de la Lorraines, un établissement du Conseil Régional. Nous vous contactons car nous souhaitons présenter la m2l mais aussi permettre aux clubs sportifs de s'inscrire et d'organiser des entraînements et / ou match. »

PROBLEMATIQUE 2 :

- La Maison de Ligue de la Lorraines nous contact car ils ont un problème au niveau de l'accès du planning lorsque les adhérents ne peuvent y pas accéder en dehors de leur domicile.

« Bonjour , nous sommes La Maison des Ligues, un établissement du Conseil Régional. Nous vous contactons car certains clubs sportifs ainsi que les adhérent de ces clubs n'ont pas une facilité à accéder de base aux informations sur leurs plannings

CONTEXTE PROBLÉMATIQUE M2L

Problématique n°1 :

- Présenter la M2L et permettre aux club sportifs de s'inscrire et organiser leur entraînements ou match contre d'autres équipes .

Problématique n°2 :

- Les clubs sportifs ainsi que leurs abonnées ne peuvent pas accéder aux plannings d'entraînements et match réservé préalablement par leurs club sportifs respectifs.

SOLUTION PROBLÉMATIQUE M2L

SOLUTION A LA PROBEMATIQUE 1

- Site web présentant la M2L afin de permettre aux clubs sportifs de s'inscrire et organiser leurs entraînements ou matchs contre d'autres équipes .

SOLUTION A LA PROBEMATIQUE 2

- Application mobile permettant aux clubs sportifs et leurs adhérent de consulter leurs plannings d'entraînements et matchs.

Conception des projects

COMPOSITION SITE WEB SOLUTION 1

- Le site web sera composé de la manière suivante :
 - Front-End : React JS
 - Back-End : Node JS
 - BDD : API REST

Composition application solution 2

- L'application mobile sera composé de la manière suivante:
 - Front-End : Dart Flutter
 - Back-End : Node.js
 - BDD : API REST

GANTT

Création d'un site web

10 déc. 2021

Tâches

2

Nom	Date de début	Date de fin	Coût
Début	31/12/2021	31/12/2021	0
PLANNING	31/12/2021	07/01/2022	114.0
Evaluation des besoins	31/12/2021	04/01/2022	57.0
Accord de projet	05/01/2022	07/01/2022	57.0
Développement	10/01/2022	03/02/2022	606.0
Création API Rest en NodJs	10/01/2022	18/01/2022	98.0
Code page d'accueil React Js	19/01/2022	21/01/2022	57.0
Code page inscription club NodeJs	19/01/2022	24/01/2022	56.0
Code page inscription adhérents NodaJs	19/01/2022	24/01/2022	56.0
Code messagerie NodeJs	19/01/2022	26/01/2022	90.0
Code gestion match NodeJs	19/01/2022	27/01/2022	98.0
Code planning NodeJs	28/01/2022	03/02/2022	95.0
Code gestion entrainement NodeJs	28/01/2022	02/02/2022	56.0
Lancement	03/02/2022	08/02/2022	56.0
Examen final du site	03/02/2022	08/02/2022	56.0

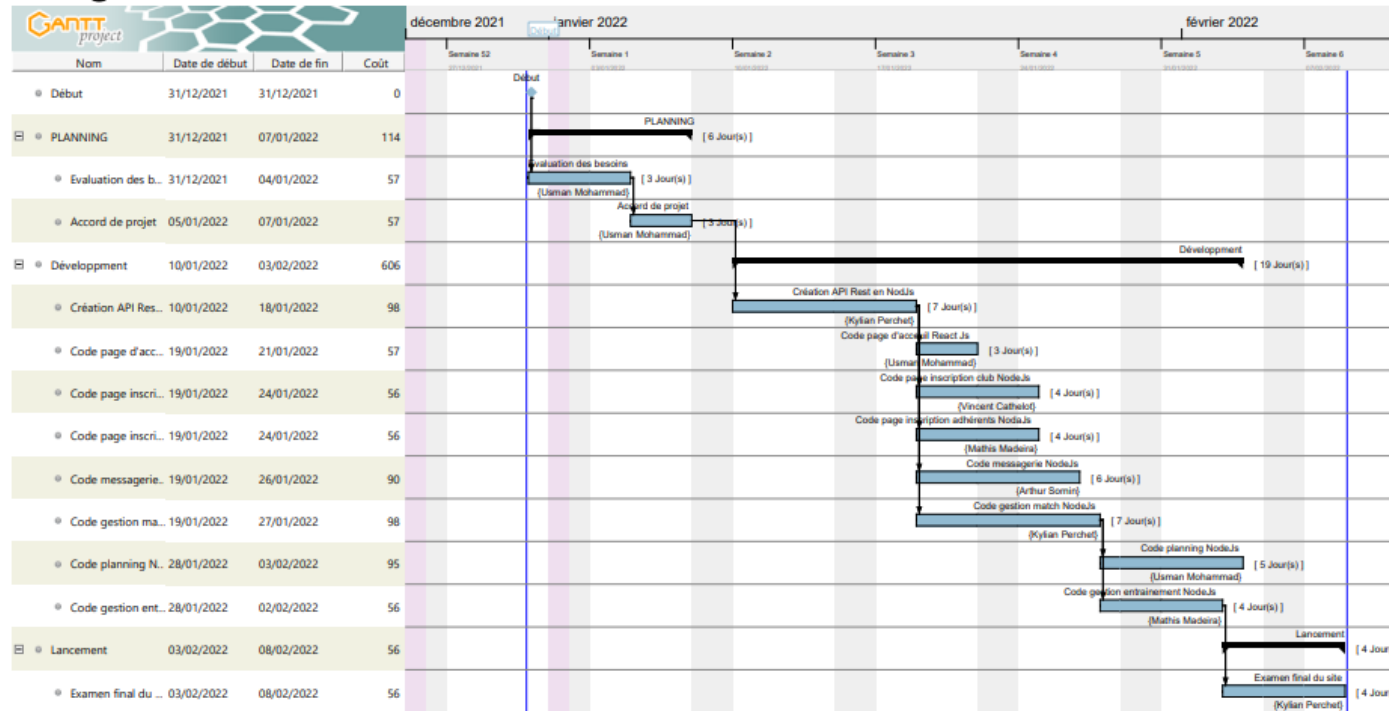
Gantt PPE 3

Création d'un site web

10 déc. 2021

Diagramme de Gantt

4



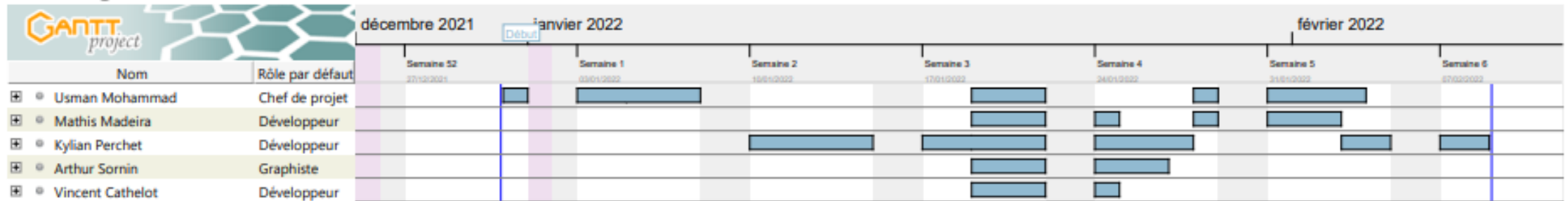
Gantt PPE 3

Création d'un site web

10 déc. 2021

Diagramme des Ressources

5



Gantt PPE 3

Création d'une application

10 déc. 2021

Tâches

2

Nom	Date de début	Date de fin	Coût
Debut	31/12/2021	20/01/2022	0
PLANNING	31/12/2021	07/01/2022	114.0
Evaluation des besoins	31/12/2021	04/01/2022	57.0
Accord de projet	05/01/2022	07/01/2022	57.0
Développement	10/01/2022	17/01/2022	297.0
Code page accueil Dart Flutter	10/01/2022	13/01/2022	56.0
Code page connexion adhérent NodeJs	10/01/2022	13/01/2022	56.0
Code messagerie NodeJs	10/01/2022	17/01/2022	90.0
Code planning NodeJs	10/01/2022	14/01/2022	95.0
Lancement	17/01/2022	20/01/2022	56.0
Examen final du site	17/01/2022	20/01/2022	56.0

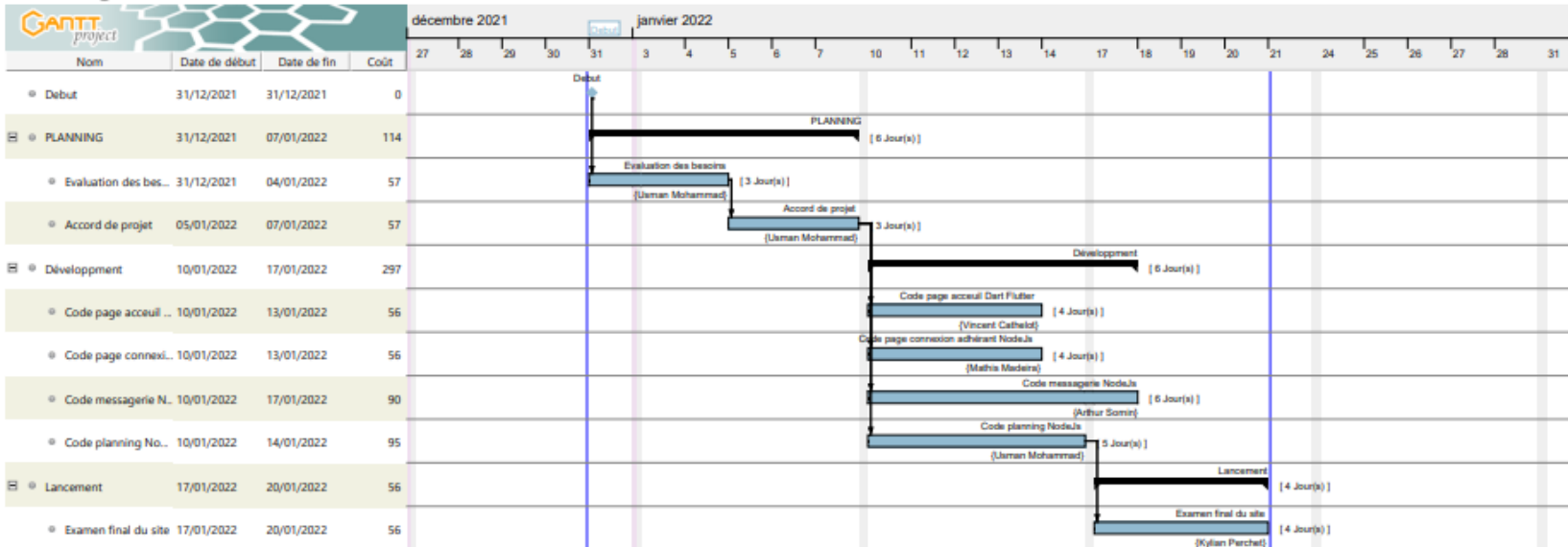
Gantt ppe4

Création d'une application

10 déc. 2021

Diagramme de Gantt

4



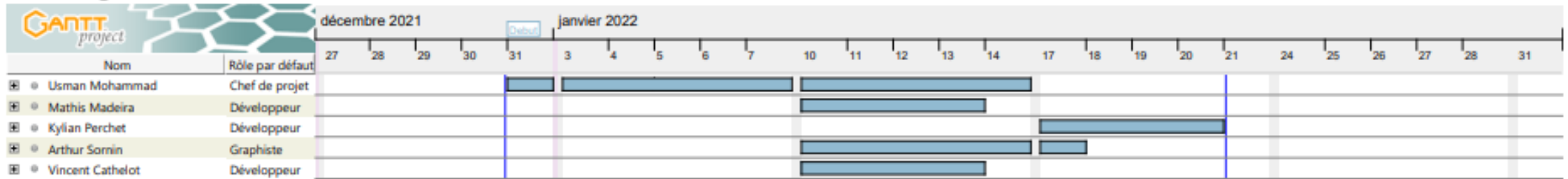
Gantt ppe4

Création d'une application

10 déc. 2021

Diagramme des Ressources

5



Gantt ppe4

Devis

Devis Problème n°1



128 All. des Champs Elysées,
91042 Évry-Courcouronnes

01.20.03.40.05

Ath.sue@outlook.fr

DEVIS

N° du devis : 1 000

Date : 03/11/2021

Date d'expiration:

N° Client: 1 125

M2L

13. rue Jean Moulin

54510 TOMBLAINE

03.83.18.87.40

Site web pour la M2L.

Description	Unité	Montant HT	Montant TTC
Planning	1	114 €	136.80 €
<u>Front-End React</u>	1	98 €	117.60 €
<u>Back-End Node.js</u>	1	508 €	609.60 €
Lancement	1	56 €	67.20€

Total HT : 776 €

TVA 20%

Total TTC: 931.20 €

Devis

Problème n°2



128 All. des Champs Elysées,
91042 Évry-Courcouronnes

01.20.03.40.05

Ath.sue@outlook.fr

DEVIS

N° du devis : 1 000

Date : 03/11/2021

Date d'expiration:

N° Client: 1 125

M2L

13. rue Jean Moulin

54510 TOMBLAINE

03.83.18.87.40

Application mobile pour la
M2L

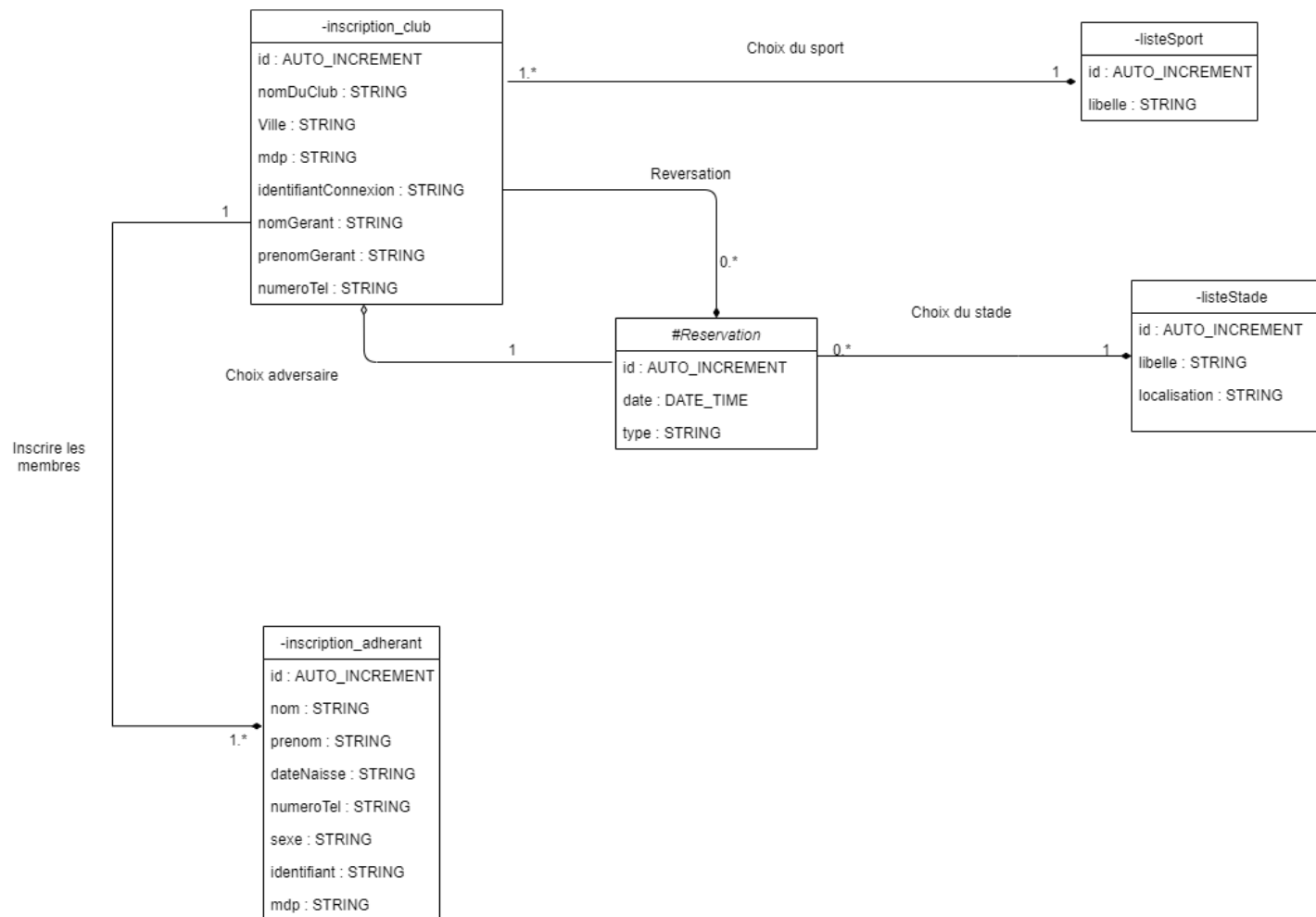
Description	Unité	Montant HT	Montant TTC
Planning	1	114 €	136.80 €
<u>Front-End</u> Dart Flutter	1	56 €	67.20 €
<u>Back-End</u> Node.js	1	241 €	289.20 €
Lancement	1	56 €	67.20 €

Total HT : 467 €

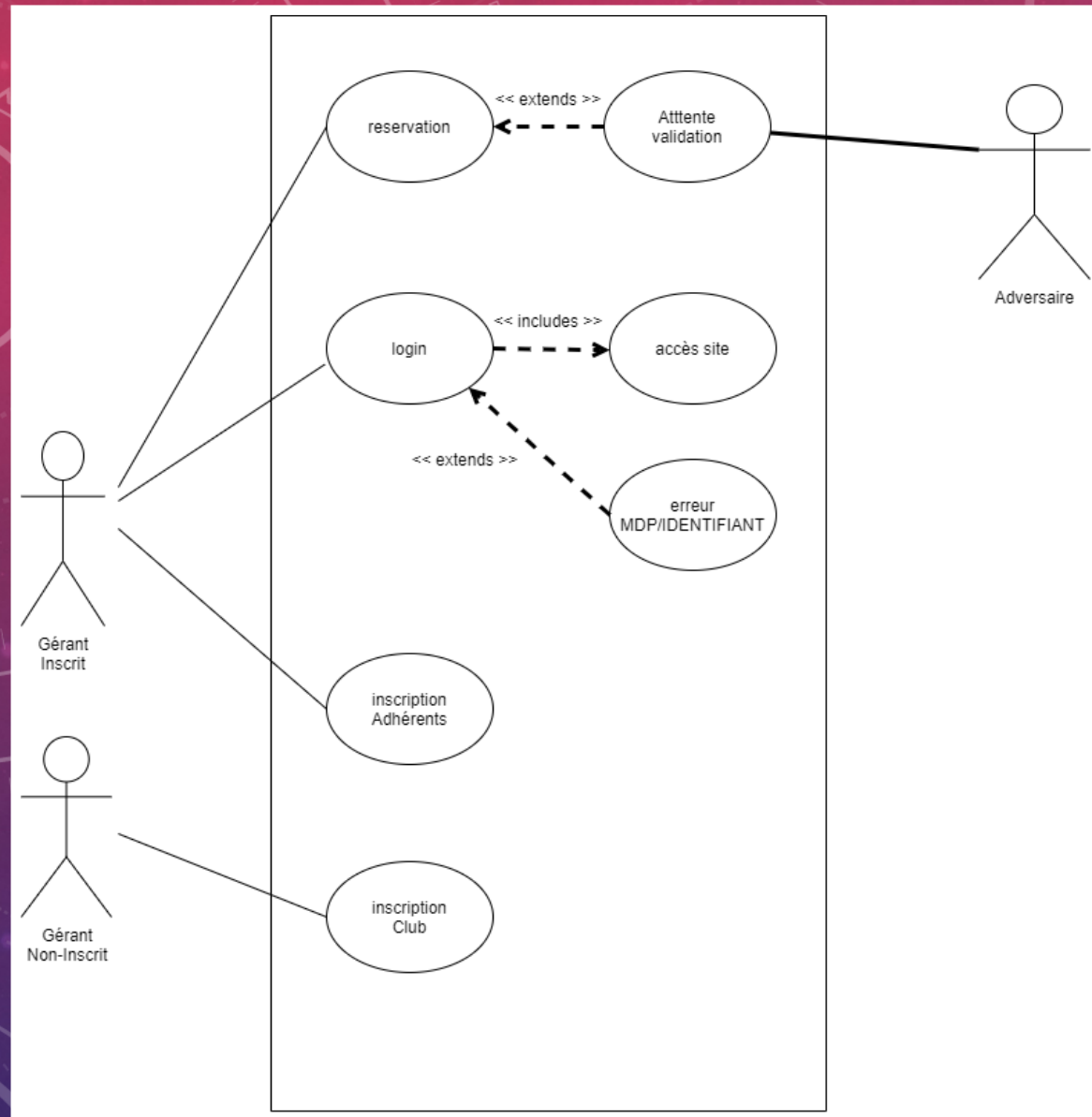
TVA 20%

Total TTC: 560.40 €

PRESENTATION DE LA MODELISATION



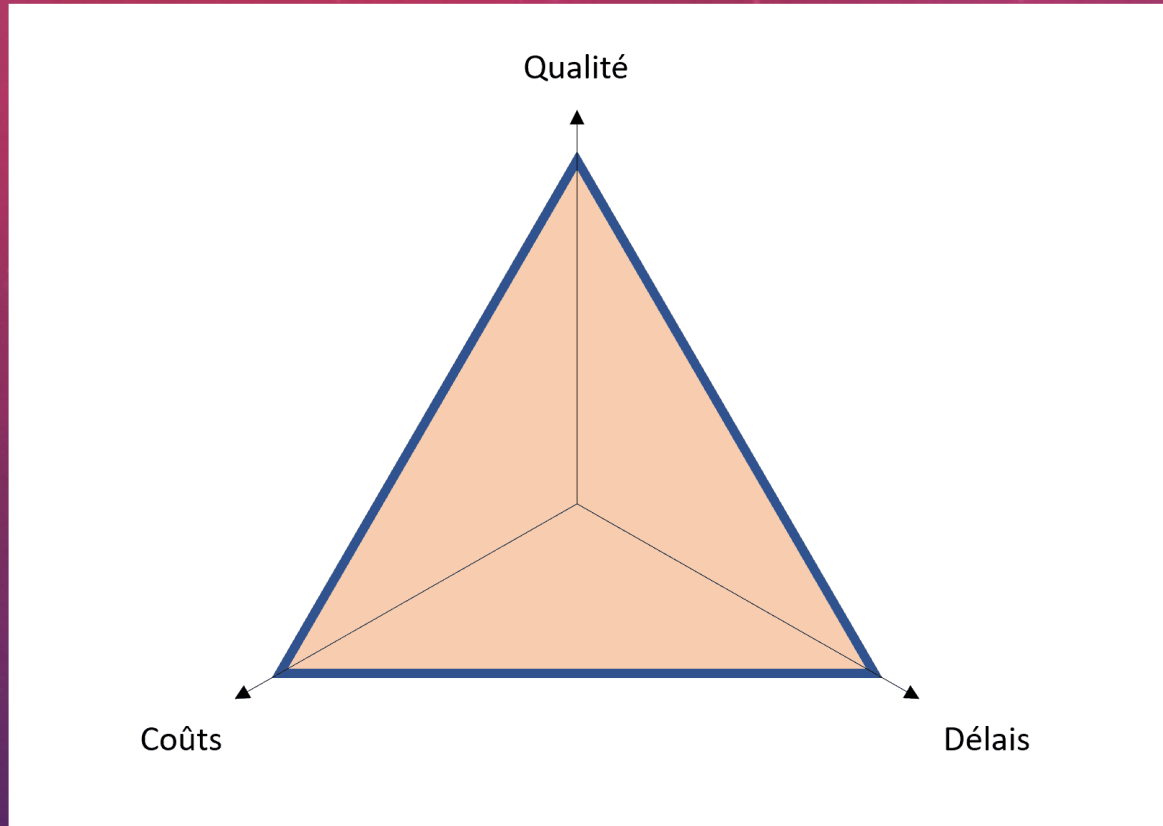
UML 2



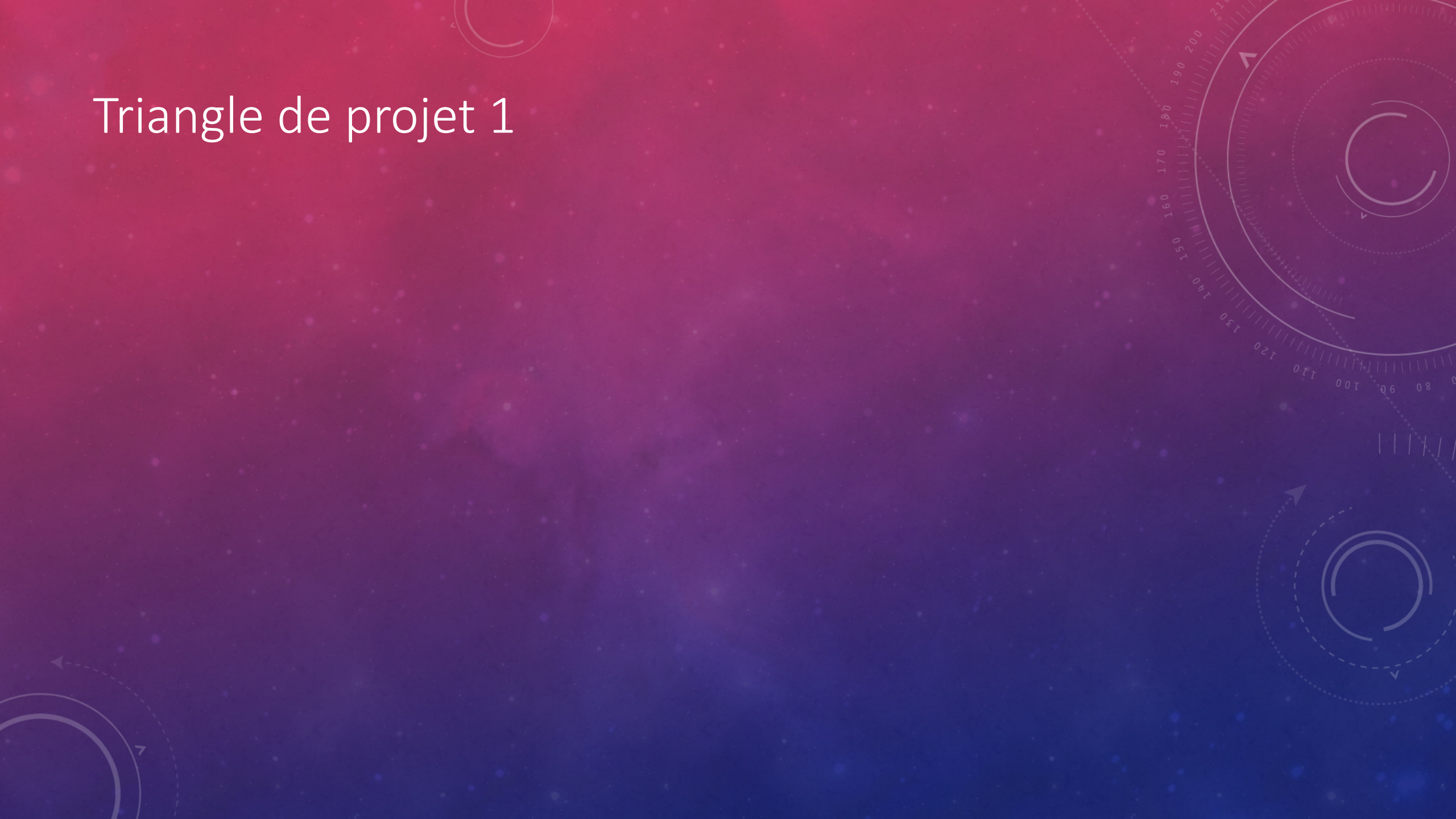
Use case

TRIANGLE DE PROJET

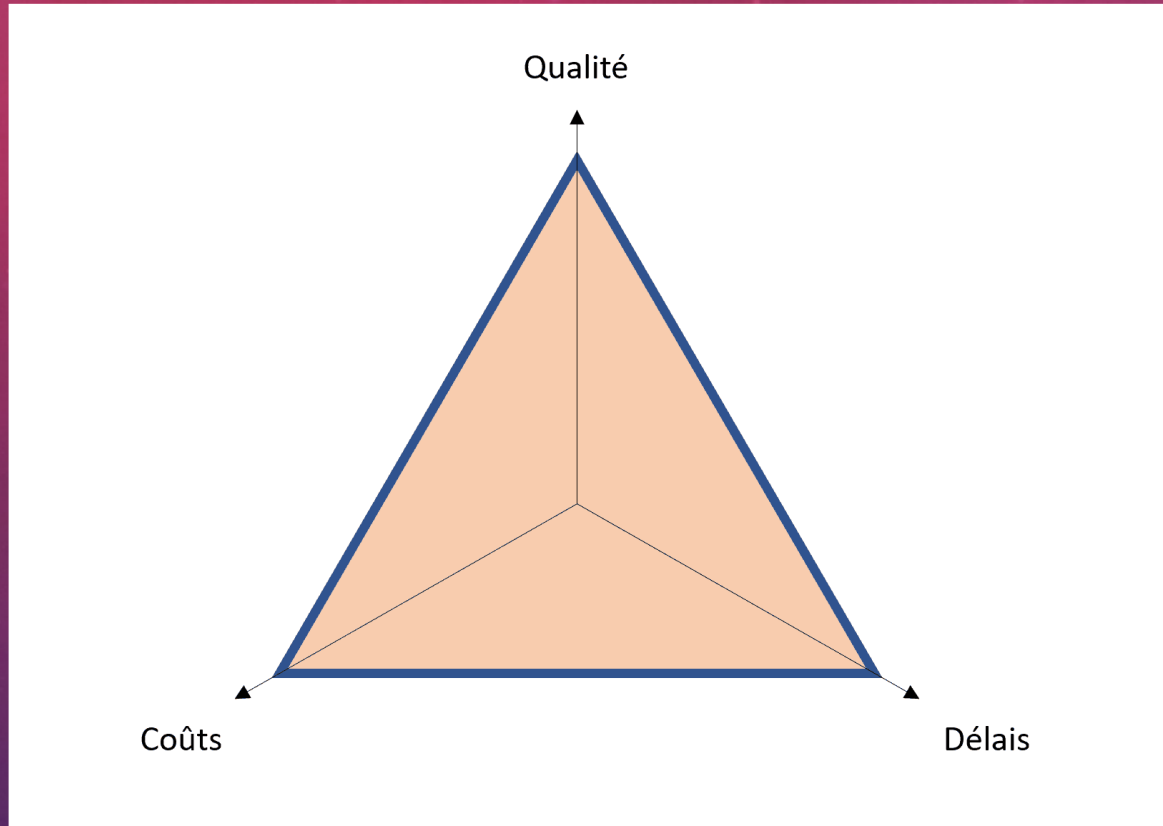
Triangle de projet 1



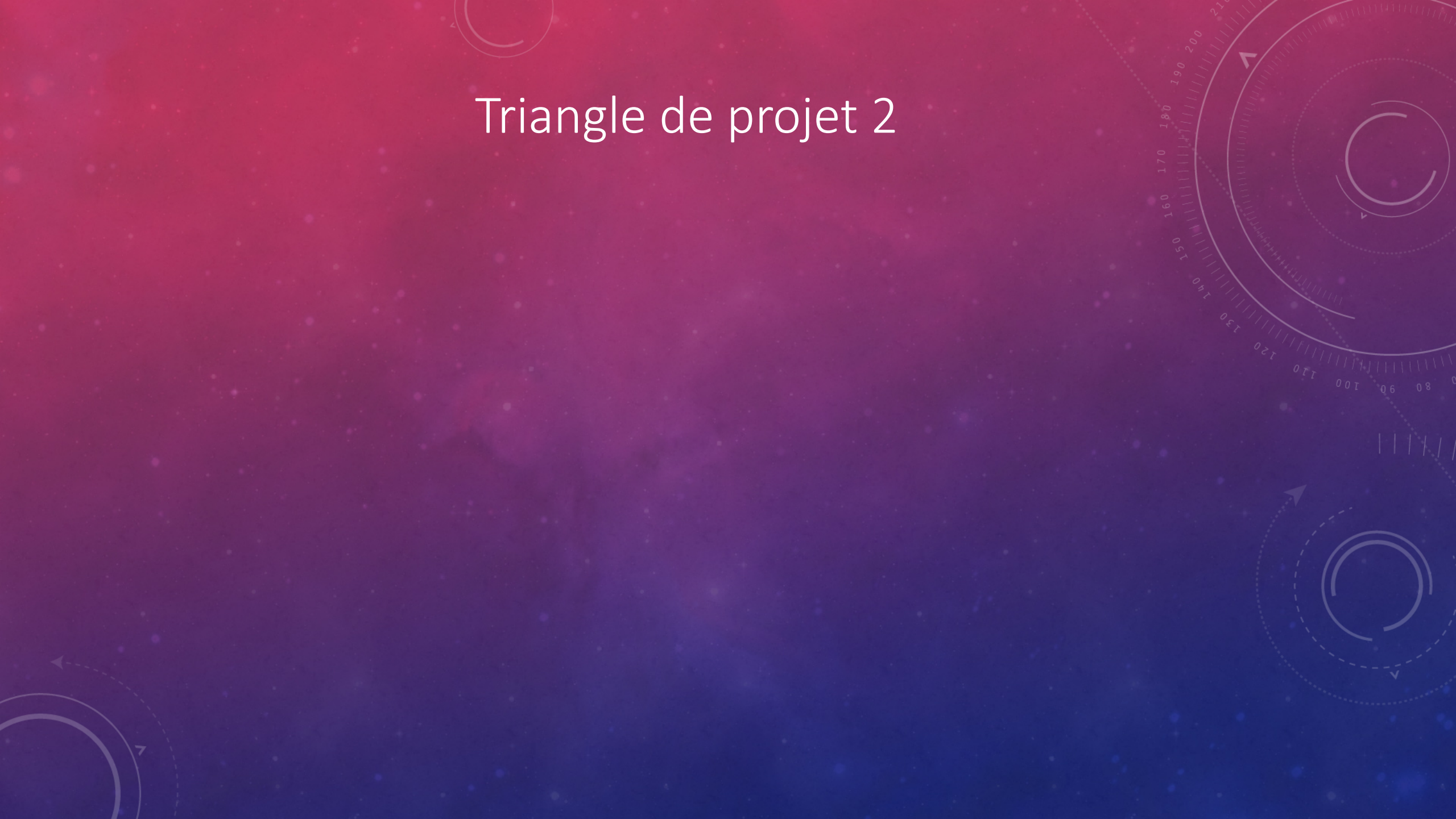
Triangle de projet 1



Triangle de projet 2



Triangle de projet 2



Vielles Technologique

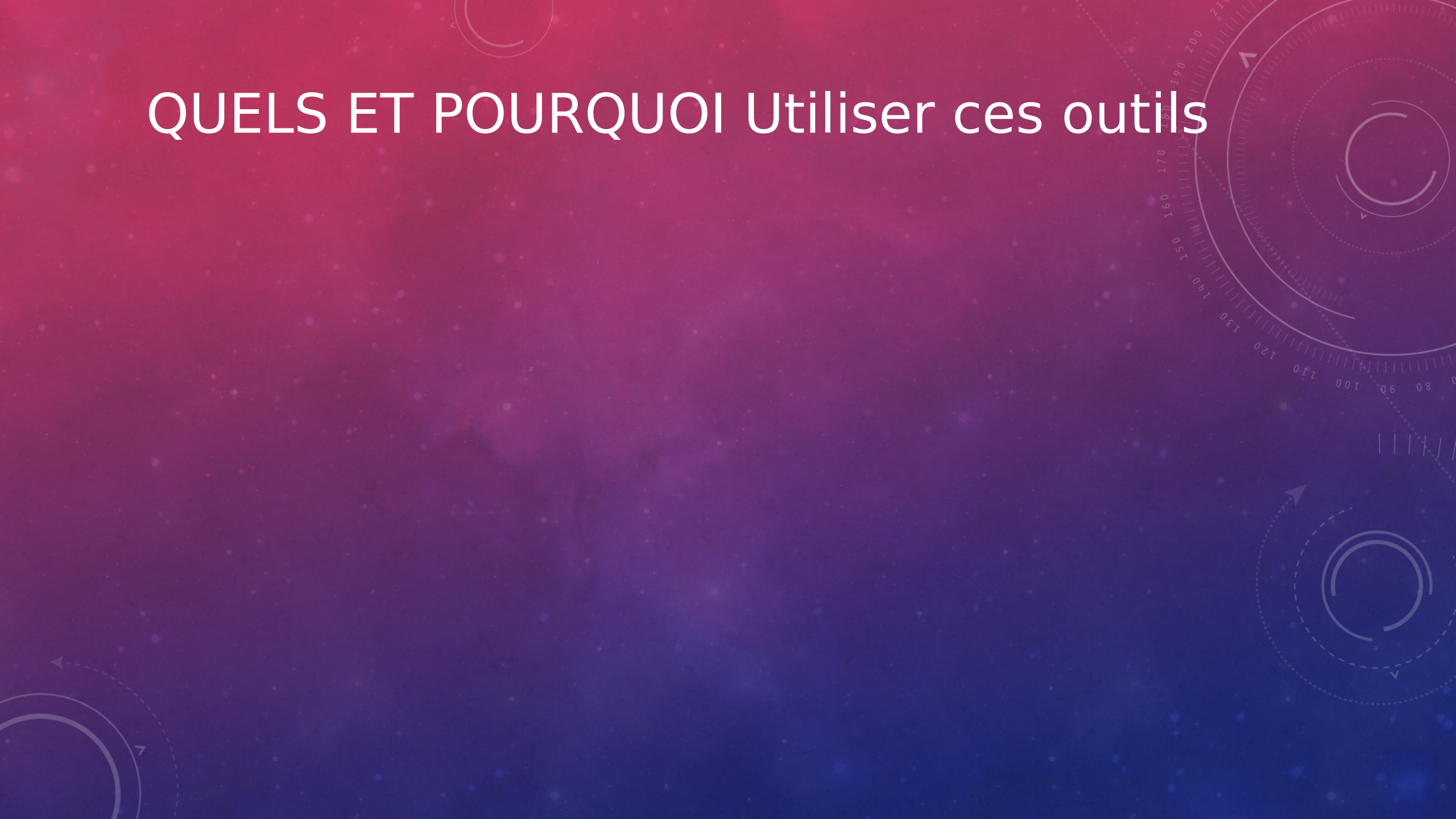
Veilles technologique

- Les différents outils de développement utilisé:
 - - REACT
 - - DART FLUTTER
 - - NODE.JS
 - - API REST

Langage utilisé pour le projet

- REACT JS : JavaScript / JSX / CSS
- DART-FLUTTER : Dart / C++
- NODE.JS : JavaScript
- API REST : Json

QUELS ET POURQUOI Utiliser ces outils



Pourquoi ces outils de développement

- REACT JS : modifie les données sans avoir à recharger la page, d'être rapide, évolutif et simple.
- DART FLUTTER : créer des applications multiplateformes extrêmement performantes et dynamiques
- NODE.JS : performant au niveau de la rapidité d'exécution
- API REST : permet de récupérer les données sur plusieurs langage

React JS

- React est une bibliothèque JavaScript open-source qui est utilisée pour construire des interfaces utilisateur spécifiquement pour des applications d'une seule page. Elle est utilisée pour gérer la couche d'affichage des applications web et mobiles. React nous permet également de créer des composants d'interface utilisateur réutilisables. React a été créé par Jordan Walke, un ingénieur logiciel travaillant pour Facebook. React a été déployé pour la première fois sur le flux d'informations de Facebook en 2011 et sur Instagram.com en 2012.

React permet aux développeurs de créer de grandes applications web qui peuvent modifier les données, sans avoir à recharger la page. L'objectif principal de React est d'être rapide, évolutif et simple. Il ne fonctionne que sur les interfaces utilisateur de l'application. Cela correspond à la vue du modèle MVC. Il peut être utilisé avec une combinaison d'autres bibliothèques ou frameworks JavaScript, tels que Angular JS dans MVC.

React JS est également appelé simplement React ou React.js.

Comment fonctionne react js

- Dans React, au lieu d'utiliser le JavaScript habituel pour les modèles, il utilise JSX. JSX est un simple JavaScript qui permet de citer du HTML et utilise la syntaxe de ces balises HTML pour afficher des sous-composants. La syntaxe HTML est traitée dans les appels JavaScript de React Framework. On peut aussi écrire en JavaScript ancien pur.

Avantages react js

- **1. Simplicité**

- ReactJS est tout simplement plus simple à saisir tout de suite. L'approche basée sur les composants, le cycle de vie bien défini et l'utilisation de JavaScript simple rendent React très simple à apprendre, à construire un site web professionnel (et des applications mobiles) et à le prendre en charge. React utilise une syntaxe spéciale appelée JSX qui vous permet de mélanger HTML et JavaScript.

- **2. Facile à apprendre**

- Toute personne ayant des connaissances de base en programmation peut facilement comprendre React, tandis que Angular et Ember sont désignés par l'expression "Domain-specific Language", ce qui implique qu'il est difficile de les apprendre. Avec react, il suffit de posséder des connaissances de base en CSS et en HTML.

- **3. Approche native**

- React peut être utilisé pour créer des applications mobiles (React Native). Et React est un fervent partisan de la réutilisation, ce qui signifie que la réutilisation du code est largement prise en charge. Ainsi, nous pouvons en même temps créer des applications iOS, Android et Web.

Inconvéniant react js

- Dépend de Facebook
- En matière de sécurité javascript n'est pas le mieux
- Malgré la légèreté de la syntaxe, React requiert une connaissance approfondie pour intégrer l'interface utilisateur au framework MVC.

Dart flutter

- Flutter est un framework de **développement d'applications multiplateforme**, conçu par Google, dont la première version a été publiée sous forme de projet open source à la fin de l'année 2018. Flutter met à disposition une grande variété de bibliothèques d'éléments d'UI standard pour Android et iOS.
- Flutter est principalement utilisé pour le développement d'applications Android et iOS, sans nécessiter la création d'une base de code propre à chacun de ces deux systèmes si différents l'un de l'autre. De cette manière, les applications de smartphone se comportent, sur les différents appareils, comme de réelles applications natives.

Comment fonctionne Dart flutter

- Le Flutter SDK se base sur le langage de programmation Dart également développé par Google. Il se veut le successeur moderne du langage JavaScript classique et, tout comme ce dernier, il s'exécute directement sur les navigateurs, sous forme d'application web. Les programmes Dart peuvent aussi être exécutés directement sur un serveur.
- Dans le navigateur, ils sont employés avec le **transcompilateur** Dart2js dans JavaScript. Les applications de la nouvelle plateforme Fuchsia de Google sont directement développées avec Dart, dont la structure s'apparente à celle de **langages de programmation orientés objet** tels que Java ou C#.

AVANTAGES Inconvénient Dart flutter

- Chaque SDK et chaque langage de programmation a ses avantages et ses inconvénients. D'une manière générale, on pourrait dire que les **avantages de Flutter le démarquent clairement d'autres systèmes similaires.**

- **Avantages de Flutter**

- Base de code unique pour toutes les plateformes cibles majeures
- Dart, un langage de programmation d'apprentissage facile pour les débutants
 - Concept « Tout est un widget » offrant de nombreuses opportunités
 - Exécution performante d'applications natives sur smartphones
 - Accès à des bibliothèques complètes d'éléments d'IU préconçus
- Implémentation facile de flux de données pour la mise à jour continue des utilisateurs
- Rechargement à chaud (« Hot Reload ») pour l'accélération des tests durant la phase de développement.

- **Inconvénients de Flutter**

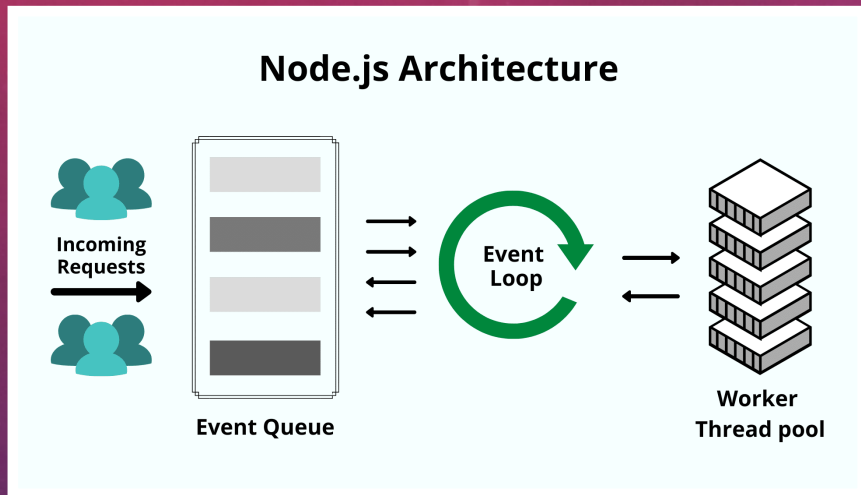
- Code de programmation confus en raison de l'intégration des widgets
- Nécessité de mise à jour des modules Flutter lors de la mise à jour des designs des différents systèmes d'exploitation. Recompilation et réinstallation des modules sur les appareils, car ils sont intégrés au programme de manière fixe.
 - Langage récent et peu répandu ; communauté encore réduite.

Node.js

- Node.js est un environnement d'exécution single-thread, open-source et multi-plateforme permettant de créer des applications rapides et évolutives côté serveur et en réseau. Il fonctionne avec le moteur d'exécution JavaScript V8 et utilise une architecture d'E / S non bloquante et pilotée par les événements, ce qui le rend efficace et adapté aux applications en temps réel.

Comment fonctionne Node.js

NodeJS est un environnement d'exécution permettant d'utiliser le JavaScript côté serveur. Grâce à son fonctionnement non bloquant, il permet de concevoir des applications en réseau performantes, telles qu'un serveur web, une API ou un job CRON.



- Node.js utilise une architecture qui permet de gérer plusieurs clients en même temps.
- Dans un modèle requête-réponse multi-thread, plusieurs clients envoient une requête, et le serveur traite chacune d'entre elles avant de renvoyer la réponse. Cependant, plusieurs threads sont utilisés pour traiter les appels simultanés.
 - Chaque fois qu'une requête arrive, Node.js la place dans une file d'attente.
- La boucle d'événements suit les requêtes bloquantes et les place dans la file d'attente une fois que la tâche bloquante est traitée. C'est ainsi qu'elle conserve sa nature non bloquante.
- Puisque Node.js utilise moins de threads, il utilise moins de ressources et de mémoire, ce qui permet une exécution plus rapide des tâches. Ainsi, pour nos besoins, cette architecture single-thread est équivalente à une architecture multi-threads

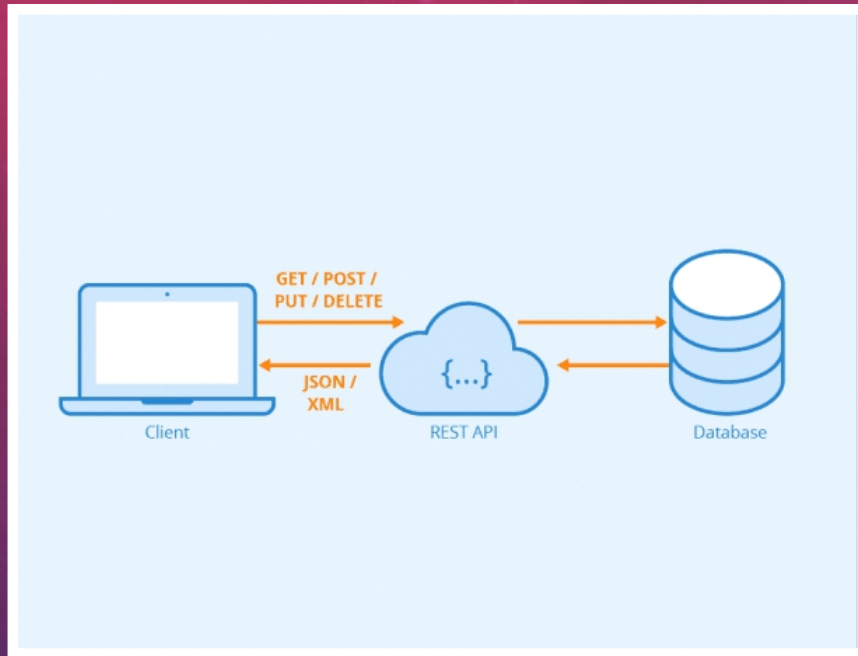
Avantages Node.js

- Node.js a connu une croissance rapide au cours des dernières années. Cela est dû à la vaste liste de fonctionnalités qu'il offre :
- **Facile** – Easy-Node.js est assez facile à prendre en main. C'est un choix incontournable pour les débutants en développement web. Grâce à de nombreux tutoriels et à une vaste communauté, il est très facile de se lancer.
- **Évolutif** – Il offre une grande évolutivité aux applications. Node.js, étant single-thread, est capable de gérer un grand nombre de connexions simultanées avec un débit élevé.
- **Vitesse** – L'exécution non bloquante des threads rend Node.js encore plus rapide et plus efficace.
- **Paquets** – Un vaste ensemble de paquets Node.js open source est disponible et peut simplifier votre travail. Aujourd'hui, il y a plus d'un million de paquets dans l'écosystème NPM.
 - **Backend solide** – Node.js est écrit en C et C++, ce qui le rend rapide et ajoute des fonctionnalités comme le support réseau.
- **Multi-plateforme** – La prise en charge multi-plateforme vous permet de créer des sites web SaaS, des applications de bureau et même des applications mobiles, le tout en utilisant Node.js.
- **Maintenable** – Node.js est un choix facile pour les développeurs, car le frontend et le backend peuvent être gérés avec JavaScript comme un seul langage.

Inconvénient node.js

- Le langage de programmation ne dispose pas d'une bibliothèque bien empilée par rapport aux autres langages de programmation de son genre
- Node.js s'est une mauvaise plate-forme en termes de calculs lourds, car il s'agit d'une opération gourmande en CPU.
- Les développeurs soulignent que l'utilisation de Node.js avec des bases de données relationnelles n'est pas une tâche facile. Les outils sont encore extrêmement sous-développés par rapport à ses concurrents sur le marché.

API REST



- La forme complète de l'API REST est l'interface de programmation d'applications de transfert d'état représentationnelle, plus communément appelée service Web API REST. Cela signifie que lorsqu'une API RESTful est appelée, le serveur transfère a représentation des ressources demandées Etat au système client.
- Par exemple, lorsqu'un développeur demande à l'API Twitter de récupérer l'objet d'un utilisateur (une ressource), l'API renvoie l'état de cet utilisateur, son nom, ses abonnés et les publications partagées sur Twitter. Cela est possible grâce à l'intégration de l'API.
- Cette représentation de l'état peut être au format JSON, XML ou HTML.

Comment fonctionne api rest

- REST détermine la structure d'un API. Les développeurs s'obligent à un certain ensemble de règles lors de la conception d'une API. Par exemple, une règle stipule qu'un lien vers une URL doit renvoyer des informations.
 - Chaque URL est connue sous le nom de demande, tandis que les données renvoyées sont appelées réponse.
- L'API REST décompose une transaction pour générer une séquence de petits composants. Chaque composant aborde un aspect fondamental spécifique d'une transaction. Cette modularité en fait une approche de développement flexible.
- Une API REST exploite les méthodes HTTP décrites par le Protocole RFC 2616. Il utilise les requêtes suivantes:
 - **ÉCONOMISEZ** récupérer des données
 - **PUT** pour modifier l'état des données (comme un objet, un fichier ou un bloc)
 - **POSTEZ** pour créer des données
 - **EFFACER** pour l'éliminer

Avantages API REST

- Un protocole REST permet d'être simplement accédé, aussi bien par un autre élément du SI, que par une application cliente. On parle alors d'architecture WOA (Web Oriented Architecture), qui est un sous-ensemble des SOA. il n'y a pas de « préjugé » côté client.
- Avec son approche orientée ressources (au lieu de traitement), et le fait que les serveurs sont sans état, la tenue de la charge est facilitée par la possibilité de passer à l'échelle une partie du SI qui serait particulièrement sollicitée.
- Le protocole HTTP est un protocole déconnecté, extrêmement maîtrisé par tous les devops, sys admin, développeurs.... Les outils de protection (firewalls), de tenue de la qualité de service, ainsi que les librairies de développement d'API existent et sont omniprésents dans de nombreux environnements: la plupart des organisations utilisent aujourd'hui ces API REST pour permettre l'accès aux données de leurs différents systèmes.



FIN