
Équipe 203

PolyDraw
Plan de projet

Version 1.4

Historique des révisions

Date	Version	Description	Auteur
2020-01-30	1.0	Version initiale du plan de projet	Pascal Alexandre-Morel Allan Beddouk Martin Pouliot Philippe Côté
2020-02-07	1.1	Ajout des numéros aux tableaux	Cédric Tessier
2020-02-07	1.2	Corrections orthographiques	Martin Pouliot
2020-02-07	1.3	Corrections orthographiques, formatage et ajout des signatures	Philippe Côté-Morneault
2020-04-09	1.4	Correctifs pour la remise finale	Martin Pouliot

Table des matières

1. Introduction	4
2. Énoncé des travaux	4
2.1. Solution proposée	4
2.2. Hypothèses et contraintes	4
2.2.1 Ressources humaines	4
2.2.2 Échéancier	5
2.2.3 Équipement et connexion	5
2.3. Biens livrables du projet	5
3. Gestion et suivi de l'avancement	5
3.1. Gestion des exigences	5
3.2. Contrôle de la qualité	5
3.3. Gestion de risque	6
3.4. Gestion de configuration	9
4. Échéancier du projet	9
5. Équipe de développement	23
6. Entente contractuelle proposée	24
6.1 Définitions	24
6.2 Objet de l'entente	24
6.3 Durée de l'entente	24
6.4 Rémunération	24
6.5 Modalités de paiement	24
6.6 Modification des exigences	24
6.7 Propriété intellectuelle	25
6.8 Résiliation du contrat	25
6.9 Responsabilité post-livraison	25
6.10 Signatures	25
7. Annexes	26

Plan de projet

1. Introduction

Ce document a pour but de donner une bonne perspective de l'organisation de l'équipe pour mener à bien le projet. Étant le troisième projet intégrateur du baccalauréat en génie logiciel, chaque membre de l'équipe a eu différentes expériences de travaux en équipe (notamment avec les projets de première et de deuxième année). Il est donc important qu'on présente une organisation et une manière de travailler uniforme et efficace. C'est donc tout l'intérêt du plan de projet.

Tout d'abord, on retrouve l'énoncé des travaux qui décrit la solution que nous proposons avec notamment la présentation de nos choix stratégiques, accompagnés des hypothèses et contraintes qui en découlent.

Ensuite, une partie portant sur la gestion et le suivi de l'avancement du projet. Cette dernière présente l'organisation de l'équipe relative aux exigences, aux risques et à la qualité du projet.

Les deux parties suivantes traitent respectivement de l'échéancier du projet et de la présentation des membres de l'équipe. Ces deux parties permettent notamment d'introduire chacun des membres de l'équipe. D'une part avec l'échéancier où nous verrons dans quelle partie du projet chacun aime travailler et d'autre part avec la présentation des membres qui donne une bonne idée des compétences et points forts de chacun.

Enfin, la dernière partie traite de l'entente contractuelle découlant directement de l'appel d'offres soumises pour ce projet. Nous avons ainsi décidé de subdiviser cette partie avec des sous-parties afin de mieux comprendre le lien direct avec l'appel d'offres.

2. Énoncé des travaux

2.1. Solution proposée

Pour résoudre ce projet, nous avons choisi de développer l'application Android avec Kotlin qui est un langage plus facile à prendre en main. De plus, nous avons tous eu de l'expérience avec Java et avons choisi d'étendre notre connaissance dans un nouveau langage.

Côté serveur, nous avons choisi de le développer en Go et de l'héberger à Polytechnique Montréal. Pour les mêmes raisons que pour l'application Android, nous avons choisi ce langage pour en apprendre un nouveau, mais aussi pour ces performances.

Pour les deux clients (lourd et léger), le logiciel offrira le clavardage entre utilisateurs. Seulement le mode intégré sera disponible pour le client léger. En ce qui concerne le client lourd, il sera possible de discuter en mode fenêtré ou en mode intégré. Plusieurs canaux de discussion peuvent être créés par les utilisateurs. En plus du système de clavardage, le logiciel permettra de jouer au jeu *Fais-moi un dessin*. Il est possible de jouer à différents modes de jeux afin de rendre la partie le plus divertissante possible. On peut jouer en équipe, en solo ou encore contre des joueurs virtuels. Un logiciel de base Poly Paint est fourni avec le client lourd. Ceci permet de commencer rapidement le développement de l'application.

Il existe une interface d'administration pour gérer les comptes et modifier les jeux si jamais on le désire.

2.2. Hypothèses et contraintes

2.2.1 Ressources humaines

L'équipe de développement est formée de 6 étudiants qui peuvent fournir un minimum de 15h de travail hebdomadaire. Ces étudiants sont à l'aise à programmer, mais pas nécessairement avec les langages de programmation Kotlin, Go et C#. Une période d'adaptation sera nécessaire.

2.2.2 Échéancier

L'équipe de travail a reçu la proposition de projet le 8 janvier 2020. La réponse à l'appel d'offres doit être soumise le 7 février. Le produit final doit être livré le 13 avril 2020. L'équipe fera un suivi hebdomadaire chaque mercredi afin de faire un retour sur l'avancement du projet, des obstacles rencontrés et de discuter de la planification du travail.

2.2.3 Équipement et connexion

L'équipe émet l'hypothèse que l'équipement sur lequel la solution logiciel s'exécute est fonctionnel et n'est pas brisé ou endommagé. La connexion avec le service PolyDraw doit être faite sur une bonne connexion ayant au moins 5 Mbps et une latence de moins de 40ms. Dans le cas où la connexion ne respecterait pas les critères, le logiciel pourra en avertir l'utilisateur.

2.3. Biens livrables du projet

7 février 2020: Réponse à l'appel d'offres soit le plan de projet, le SRS, la liste d'exigences, le document d'architecture logicielle, le protocole de communication, ainsi que les prototypes de communication serveur client lourd et serveur client léger.

13 avril 2020: Mise à jour des artefacts remis lors de la remise précédente, le plan de tests, les résultats de tests, ainsi que le code source et exécutable du produit final.

3. Gestion et suivi de l'avancement

3.1. Gestion des exigences

Les exigences sont répertoriées dans le fichier « Listes des exigences 203 ». Tout changement apporté aux exigences préalablement approuvées doit être approuvé de nouveau par les membres de l'équipe et évalué par ceux-ci. Les exigences doivent être validées de nouveau par le client. Ceci a pour but de s'assurer de ne pas perdre de points inutilement dans le cas où un changement serait nécessaire.

Avant de proposer une exigence, les membres de l'équipe de développement doivent s'assurer que cette exigence ne change pas la charge de travail associée au pointage des exigences préalablement approuvées. Les exigences étant déjà approuvées, proposer une nouvelle exigence doit se faire en cas d'extrême nécessité. Par exemple, une limitation technologique, un départ d'un des membres de l'équipe ou encore une force majeure.

Les exigences ne sont pas supposées changer au courant du projet. Dans le cas où de nouvelles exigences seraient imposées par le client, les exigences actuelles seront revues avant de faire place aux nouvelles exigences. L'échéancier devra être refait et sera proposé dans une nouvelle version de ce document.

Dans le cas d'une modification apportée aux exigences souhaitables, une rencontre d'équipe doit avoir lieu. Lors de cette rencontre, il faut établir les points suivants. Comment est-ce que ce changement va affecter l'échéancier ? Quelle partie du code cela affecte-t-il ? Est-ce que les fonctionnalités existantes sont affectées ? Est-ce que ce changement rend des parties incompatibles entre elles ? De plus, cette fonctionnalité doit être documentée dans la documentation comme le SRS et si elle a un impact sur l'architecture et le protocole de communication elle doit s'y retrouver. Toute planification déjà effectuée dans Redmine doit être revue et dans certains cas, des tâches devront être annulées ou fermées pour faire place à celles de la nouvelle fonctionnalité.

3.2. Contrôle de la qualité

Le contrôle de la qualité est assuré par une personne responsable de l'artefact. L'artefact comporte autant le produit final que les documents. Cette personne est responsable de s'assurer que l'artefact est recevable et respecte les requis imposés ou déterminés au début du projet. Cette personne est affectée à une tâche pour l'artefact dans le logiciel de gestion de projets. Les autres membres de l'équipe ouvrent des sous-tâches pour travailler sur les fonctionnalités et sections de l'artefact. Une fois que toutes les sous-tâches sont terminées, la personne nommée « Project Manager » effectue l'assurance qualité sur l'artefact. Au besoin, elle suscite l'aide d'un autre membre de l'équipe pour l'aider à faire l'assurance qualité.

Le gestionnaire de projet « Project Manager » est aussi responsable de s'assurer que les sous-tâches respectent l'échéancier. Au besoin, il rencontre les membres responsables des sous-tâches afin de planifier l'échéancier de l'artefact à nouveau. Si celui-ci prend trop de retard (plus de 4 jours), une rencontre d'équipe a lieu pour refaire l'échéancier. Vers la fin du jalon le « Project Manager » rédige un plan de tests avec les différents cas de tests à tester. Ces cas de tests sont des cas de tests en boîte noire. Ces cas de tests proviennent des exigences fonctionnelles et non fonctionnelles. Ceux-ci sont gardés dans un dossier partagés entre les membres de l'équipe. Ainsi les développeurs peuvent aller vérifier que l'artefact produit est conforme à ce qui est attendu. Ils doivent ensuite marquer qu'ils ont vérifié que l'exigence a été testée. Ces plans de tests seront résumés dans l'artefact final plan de test.

Dans le cas où l'artefact n'est pas jugé satisfaisant, le gestionnaire de projet est responsable d'ouvrir des tâches de types anomalies liées à l'artefact. Une fois cette première passe faite, le gestionnaire de projet demande aux autres personnes de l'équipe de vérifier si la qualité convient. Dans le cas où des anomalies sont détectées, des tâches de types anomalies doivent être ouvertes et assignées dans le logiciel de gestion de projet. Une fois que ces problèmes sont corrigés, le gestionnaire de projet vérifie que cette anomalie n'est plus présente dans l'artefact. Le cycle se répète dans le cas d'un problème subséquent.

Afin de s'assurer que le code possède une qualité suffisante, des revues de codes sont prévues en fin de jalon. De plus, les projets des différentes plateformes utilisent des linters afin d'assurer une consistance dans le code ainsi qu'une qualité.

3.3. Gestion de risque

La description des risques suit la convention suivante :

- Ampleur : sur une échelle de 1 à 10, 10 étant le risque le plus élevé. Cette analyse est basée sur la probabilité d'occurrence du risque, ainsi que ses impacts.
- Description : une description textuelle du risque ainsi que les problèmes attendus.
- Impact : échelle définissant la portée du risque
 - C – critique (affecte le projet en entier)
 - E – élevé (affecte les fonctionnalités principales du système)
 - M – moyen (devrait être maîtrisable en appliquant une stratégie d'atténuation adéquate)
 - F – faible (l'acceptation du risque est une stratégie envisageable)
- Facteurs : aspects (**métriques**) du système pouvant être compromis.
- Stratégie de gestion : mesures à prendre afin de gérer le risque.

Tableau 3.1 - Problème entrepôt de fichiers				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
4	Suppression accidentelle, virus, erreurs de manipulation de git	M	Productivité (heures dans Redmine)	Chaque personne possède localement une copie du code. Dans le cas d'une suppression accidentelle ou d'un bogue avec l'entrepôt de fichier, on peut créer un nouveau projet et envoyer le projet facilement sur ce nouvel entrepôt. De plus, l'entrepôt est en réplication sur l'entrepôt fourni par Polytechnique Montréal.

Tableau 3.2 - Départ d'un membre				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
6	Un membre de l'équipe quitte l'équipe pour une raison personnelle ou autre.	E	Productivité (heures dans Redmine)	Dans le cas d'un départ d'un membre de l'équipe, celui-ci ne pourra pas être remplacé. Il faut donc refaire l'échéancier et voir s'il est possible de remplir toutes les exigences avec un membre en moins. Si ce n'est pas possible, il faut revoir les exigences et en couper au besoin.

Tableau 3.3 - Bris matériel et pertes de données				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
4	Un bris matériel sur un ordinateur d'un membre d'équipe cause une perte de données sur des éléments du projet	M	Productivité (heures dans Redmine)	Afin d'éviter les pertes de données, tous les documents sont rédigés dans Google Drive. Les fichiers se retrouvent dans des entrepôts Git qui sont répliqués.

Tableau 3.4 - Qualité des artefacts				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
8	Un artefact non fonctionnel est livré	C	Qualité (Anomalies dans Redmine)	Afin d'éviter que des artefacts non fonctionnels soient livrés, ceux-ci passent une série de tests et sont compilés automatiquement sur Gitlab. Ceci permet d'assurer une constante vigilance sur la qualité. De plus, les artefacts doivent suivre le processus de la section 3.2 afin de minimiser ce risque.

Tableau 3.5 - Changements des exigences ou ajout de fonctionnalité par le client				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
3	Le client demande une nouvelle fonctionnalité pour une prochaine version qui n'était pas dans la liste initiale.	F	Réorganisation de l'échéancier	Ce cas n'est pas supposé arriver vu la nature du projet. Dans le cas où ceci devrait arriver, le client sera rencontré et des exigences devront être retirées pour faire place à la nouvelle exigence. Le processus de la section 3.1 sera utilisé afin de gérer la nouvelle exigence.

Tableau 3.6 - Limitations technologiques d'une plateforme				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
6	La plateforme Android ou Windows ne s'attend pas à nos attentes et on rencontre une limitation technique qui nous empêche de faire la fonctionnalité escomptée	E	Retard sur l'échéancier	Dans ce cas, il faut contourner le problème. La fonctionnalité peut probablement être implémentée d'une autre façon qui ne rencontre pas la limitation technologique. Dans le cas où aucune solution ne peut être trouvée, on peut regarder pour remplacer cette fonctionnalité si possible.

Tableau 3.7 - Problèmes de performances de l'application				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
6	La plateforme ne répond pas dans le temps défini dans le SRS. Temps d'ouverture d'une fenêtre par exemple, temps de réponse du serveur, etc.	M	Temps passés sur les anomalies	Utiliser les différents profileurs inclus dans les IDEs. Dans le cas où le réseau est en cause, isoler le problème et déplacer le serveur si cela peut améliorer le temps de réponse.

Tableau 3.8 - Problèmes de réseau lors de la présentation				
Ampleur	Description	Impact	Facteurs	Stratégie de gestion
8	Une coupure réseau survient lors de la présentation finale. Ceci pourrait donc être perçu comme un bogue.	M	Note affectée	Afin de s'assurer de trouver tous les problèmes associés au réseau, il serait intéressant lors de la présentation d'exécuter une commande qui fait des ping en continu au serveur afin de vérifier si le réseau est problématique. De plus selon le SRS, les clients doivent s'assurer que le serveur est toujours présent sinon ceux-ci vont se déconnecter automatiquement en avertissant l'utilisateur. La commande ping permet de valider le comportement.

3.4. Gestion de configuration

Les artefacts doivent être remis dans le dossier Remise de l'entrepôt de fichier. Une fois que ce dépôt est fait, le processus d'assurance qualité doit être appliqué. Ce processus est décrit dans la section 3.2 du présent document. Lors de changements apportés à l'artefact, celui-ci doit augmenter la version mineure de un. Dans le cas d'une version majeure (prototype, produit final), celle-ci doit être augmentée de un.

Les artefacts doivent être nommés de la façon suivante. Le numéro de la version doit être changé pour celui de l'artefact.

Remise\NomArtefact-v1.0.extension

Dans le cas d'artefacts qui sont compilés, ils doivent être téléchargés à partir de Gitlab. Cela assure qu'ils ont passé les validations automatiques faites par l'intégration continue. Il minimise le risque de livrer un artefact non fonctionnel. Lorsque l'artefact a passé le processus d'assurance qualité, la version doit être assignée à l'aide d'un tag dans l'entrepôt de fichiers.

Dans le cas d'artefact de documentation, ceux-ci ne possèdent pas de solutions de gestion de version. La table «Historique des révisions» doit donc être remplie systématiquement avant la remise d'un artefact dans le dossier «Remise» de l'entrepôt de fichiers. Les anciennes versions des artefacts peuvent être effacées, car Git va garder une copie de l'ancienne version.

L'outil Redmine doit être utilisé pour gérer les différents jalons sous la forme de versions. Les demandes doivent donc être ouvertes dans la bonne version. Ceci permet de suivre l'échéancier et d'évaluer si des retards potentiels sur le projet peuvent arriver. Une fois un jalon terminé, le dernier commit doit porter le tag de la version. De plus, chaque changement dans git doit se faire en se basant sur une demande. Il faut utiliser le numéro de la demande comme numéro de branche. Les branches à part celles nécessaires dans le processus, doivent être nommées sous la forme suivante. *f_####*

4. Échéancier du projet

L'échéancier présenté a pour but de guider le projet à travers les différentes étapes importantes du projet. Il a été séparé en différents jalons, afin d'assigner des objectifs clairs et atteignables. Ces objectifs permettent à l'équipe d'évaluer si elle est en retard ou en avance par rapport à la planification. Les dates sont présentées dans le tableau suivant. Ce tableau présente les différentes étapes du projet ainsi que la charge assignée à ces différentes étapes.

Tableau 4.1 - Jalons de l'échéancier				
Jalons	Date début	Date limite	Charge de travail	Description des tâches
Prototype v1.0	2020-01-15	2020-02-01	158h	<ul style="list-style-type: none"> Interface de connexion Interface de clavardage Serveur Partie réseautique
Prototype v1.1	2020-02-02	2020-02-05	86h	<ul style="list-style-type: none"> Tests logiciels sur le prototype
v1.2	2020-02-06	2020-02-14	86h	<ul style="list-style-type: none"> Clavardage <ul style="list-style-type: none"> Salles Historique Profil <ul style="list-style-type: none"> Statistiques Historique
v1.3	2020-02-15	2020-02-29	187h	<ul style="list-style-type: none"> Validation mot Création de parties ou jeux Tutoriel
v1.4	2020-03-01	2020-03-22	389h	<ul style="list-style-type: none"> Mode de jeu Personnalité des joueurs virtuels Effets visuels et sonores
v1.5	2020-03-23	2020-03-28	101h	<ul style="list-style-type: none"> Vue administrateur simple Traduction du jeu Trophées
Débogage et tests finaux v1.6	2020-03-29	2020-04-02	72h	<ul style="list-style-type: none"> Corrections de bogues sur les trois plateformes, serveur, client léger et client lourd

Cette charge de travail a été estimée à partir du meilleur de la connaissance de l'équipe ainsi que de l'expérience des membres. L'effort des membres est mesuré par heures totales pour chaque partie du travail.

Les étapes importantes du projet sont également représentées dans la ligne de temps ci-dessous et dans le diagramme de Gantt en annexe. Ces deux visuels affichent la répartition des tâches dans le temps et l'interdépendance entre les tâches.

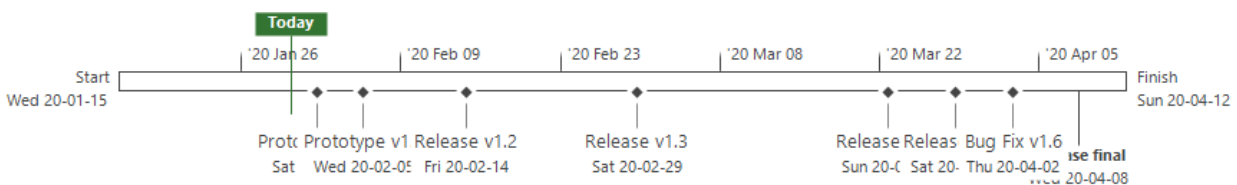


Figure 4.1: Ligne du temps des différents jalons

Les tâches de chaque jalon sont identifiées dans les tableaux suivants. Ces tâches constituent la liste des tâches qui seront assignées lors de la planification hebdomadaire. Ceci permet de diviser le travail et de s'assurer que les délais seront respectés. L'échéancier a été fait pour terminer plus de 10 jours avant la date de remise. Ceci permet d'avoir

une avance sur l'échéancier dans le cas où certains jalons dépasseraient les dates limites. Si l'échéancier n'est pas respecté, il sera mis à jour avec les nouvelles dates afin de s'assurer que le projet est toujours livré dans les temps.

** Certaines tâches ont été biffées. Ces tâches correspondent aux changements faits sur les exigences en raison des deux semaines de retrait suite à la COVID-19.

Tableau 4.2 - Prototype v1.0			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Interface connexion	Client léger	2020-01-16	6h
Interface chat	Client léger	2020-02-18	6h
Partie réseau	Client léger	2020-02-01	15h
Interface connexion	Client lourd	2020-01-16	6h
Interface chat	Client lourd	2020-02-18	6h
Partie réseau	Client lourd	2020-02-01	15h
Implémentation socket et gestion connexions	Serveur	2020-01-19	4h
Gestion utilisateur et authentification	Serveur	2020-01-21	6h
Connexion base de données	Serveur	2020-01-19	3h
Envoi de message socket	Serveur	2020-01-24	12h

Tableau 4.3 - Prototype v1.1			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Tests & QA du prototype	Client léger	2020-02-05	12h
Tests & QA du prototype	Client lourd	2020-02-05	12h
Tests & QA du prototype	Serveur	2020-02-05	6h

Tableau 4.4 - Release v1.2			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Clavardage historique Call REST et BD messages	Serveur	2020-02-11	6h
Clavardage historique Implémentation REST	Client lourd	2020-02-12	3h
Clavardage historique Historique UI	Client lourd	2020-02-14	9h
Clavardage historique Implémentation REST	Client léger	2020-02-12	3h
Clavardage historique Historique UI	Client léger	2020-02-14	9h
Salle de clavardage BD Channel et REST	Serveur	2020-02-06	3h
Salle de clavardage Socket routage message & notification de subscribe/création	Serveur	2020-02-07	6h
Salle de clavardage Implémentation UI Rooms	Client lourd	2020-02-07	6h
Salle de clavardage Implémentation REST et socket chat	Client lourd	2020-02-09	6h
Salle de clavardage Implémentation UI Rooms	Client léger	2020-02-07	6h
Salle de clavardage Implémentation REST et socket chat	Client léger	2020-02-09	6h
Profil BD et REST	Serveur	2020-02-07	6h
Profil Interface profil utilisateur	Client lourd	2020-02-07	6h
Profil REST profil utilisateur	Client lourd	2020-02-08	3h
Profil Interface profil utilisateur	Client léger	2020-02-07	6h
Profil REST profil utilisateur	Client léger	2020-02-08	3h

Statistiques BD et REST	Serveur	2020-02-10	6h
Statistiques Interface statistiques/profil	Client lourd	2020-02-09	3h
Statistiques REST statistiques	Client lourd	2020-02-11	3h
Statistiques Interface statistiques/profil	Client léger	2020-02-09	3h
Statistiques REST statistiques	Client léger	2020-02-11	3h
Historique parties BD et REST	Serveur	2020-02-12	3h
Historique parties Hooks serveur	Serveur	2020-02-13	3h
Historique parties REST Historique	Client lourd	2020-02-13	3h
Historique parties Interface historique	Client lourd	2020-02-12	3h
Historique parties REST Historique	Client léger	2020-02-12	3h
Historique parties Interface historique	Client léger	2020-02-13	3h
Création de comptes BD et REST	Serveur	2020-02-09	3h
Création de comptes Interface & REST	Client lourd	2020-02-10	6h
Création de comptes Interface & REST	Client léger	2020-02-10	6h

Tableau 4.5 - Release v1.3			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Validation mot Trouver liste mots FR/EN	Serveur	2020-02-15	3h
Validation mot Blacklister les mots pour FR/EN	Serveur	2020-02-16	3h
Construction jeu manuel 1 REST Upload, indices nom BD	Serveur	2020-02-19	9h
Construction jeu manuel 1 Dessin traits socket	Serveur	2020-02-23	12h
Construction jeu manuel 1 Interface indices, dessin, mot	Client lourd	2020-02-21	6h
Construction jeu manuel 1 Sauvegarde des traits, ordre de dessin	Client lourd	2020-02-22	9h
Construction jeu manuel 1 Interface preview traits socket	Client lourd	2020-02-26	9h
Construction jeu manuel 2 Interface (dropdown)	Client lourd	2020-02-27	3h
Construction jeu manuel 2 Faire les traits	Serveur	2020-02-29	6h
Construction jeu assisté 1 REST interface	Serveur	2020-02-17	3h
Construction jeu assisté 1 Potrace + docker	Serveur	2020-02-19	9h
Construction jeu assisté 1 Divers modes de dessin	Serveur	2020-02-21	6h
Construction jeu assisté 1 Interface paramètres potrace	Client lourd	2020-02-18	6h
Construction jeu assisté 1 Implémentation REST	Client lourd	2020-02-19	6h
Tutoriel simple	Client lourd	2020-02-18	12h
Tutoriel simple	Client léger	2020-02-18	12h

Canevas dessin Base du canevas	Client léger	2020-02-16	6h
Canevas dessin Outils de dessins	Client léger	2020-02-19	9h
Canevas dessin Exportation SVG	Client léger	2020-02-20	3h
Canevas dessin Interface dessin traits automatiques	Client léger	2020-02-26	9h
Canevas dessin Socket réseau dessins traits	Client léger	2020-02-27	12h
Thèmes client léger	Client léger	2020-02-29	8h

Tableau 4.6 - Release v1.4			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Salle d'attente REST Création de partie et bridge socket	Serveur	2020-03-02	6h
Salle d'attente Game manager service, matching	Serveur	2020-03-02	6h
Salle d'attente Binding réseau	Client lourd	2020-03-04	3h
Salle d'attente Interface	Client lourd	2020-03-03	9h
Salle d'attente Binding réseau	Client léger	2020-03-04	3h
Salle d'attente Interface	Client léger	2020-03-03	9h
Mêlée générale Intégration et passage de la salle d'attente	Client lourd	2020-03-05	3h
Mêlée générale Intégration et passage de la salle d'attente	Client léger	2020-03-05	3h
Mêlée générale Game manager gestion de la partie	Serveur	2020-03-08	12h
Mêlée générale Gestion du temps et des événements partie	Serveur	2020-03-10	6h
Mêlée générale Interface de la partie	Client lourd	2020-03-10	15h
Mêlée générale Socket gestion événements partie	Client lourd	2020-03-14	12h
Mêlée générale Interface de la partie	Client léger	2020-03-10	15h
Mêlée générale Socket gestion événements partie	Client léger	2020-03-14	12h
Sprint solo Interface REST setup partie	Serveur	2020-03-15	3h
Sprint solo Socket événements adaptation	Serveur	2020-03-17	6h
Sprint solo Interface adaptation mode jeu	Client lourd	2020-03-16	6h

Sprint solo Socket adaptation	Client lourd	2020-03-18	6h
Sprint solo Interface adaptation mode jeu	Client léger	2020-03-16	6h
Sprint solo Socket adaptation	Client léger	2020-03-18	6h
Sprint coopératif Interface REST setup partie	Serveur	2020-03-19	3h
Sprint coopératif Socket événements adaptation	Serveur	2020-03-21	6h
Sprint coopératif Interface adaptation mode jeu	Client lourd	2020-03-20	6h
Sprint coopératif Socket adaptation	Client lourd	2020-03-22	6h
Sprint coopératif Interface adaptation mode jeu	Client léger	2020-03-20	6h
Sprint coopératif Socket adaptation	Client léger	2020-03-22	6h
Joueurs virtuels Game handle	Serveur	2020-03-03	9h
Joueurs virtuels Chat et messages	Serveur	2020-03-04	3h
Joueurs virtuels Profil et comportement	Serveur	2020-03-04	3h
Effets visuels, sonores Son mot deviné	Client lourd	2020-03-11	1h
Effets visuels, sonores Son mot deviné	Client léger	2020-03-11	1h
Effets visuels, sonores Animation mot deviné	Client lourd	2020-03-12	6h
Effets visuels, sonores Animation mot deviné	Client léger	2020-03-12	6h
Effets visuels, sonores Son réception message	Client lourd	2020-03-01	1h
Effets visuels, sonores Son réception message	Client léger	2020-03-01	1h
Effets visuels, sonores Transitions entre les mots	Client léger	2020-03-11	1h

Effets visuels, sonores Transitions entre les mots	Client lourd	2020-03-11	1h
--	--------------	------------	----

Tableau 4.7 - Release v1.5			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Vue administrateur, gestion comptes REST api comptes, delete, update	Serveur	2020-03-23	3h
Vue administrateur, gestion comptes REST ajout permission handle	Serveur	2020-03-23	1h
Vue administrateur, gestion comptes REST ajout permission handle	Serveur	2020-03-23	3h
Vue administrateur, gestion comptes Interface, liste et menu édition	Client lourd	2020-03-24	6h
Vue administrateur, gestion comptes Implémentation REST	Client lourd	2020-03-24	3h
Vue administrateur, gestion des jeux REST api jeux, delete, update	Serveur	2020-03-25	3h
Vue administrateur, gestion des jeux REST api jeux, delete, update	Client lourd	2020-03-26	6h
Vue administrateur, gestion des jeux Implémentation REST	Client lourd	2020-03-26	3h
Vue administrateur, statistiques REST api statistiques read	Serveur	2020-03-27	3h
Vue administrateur, statistiques Data collection	Serveur	2020-03-28	6h
Vue administrateur, statistiques interface	Client lourd	2020-03-28	6h
Vue administrateur, statistiques Implémentation REST	Client lourd	2020-03-28	3h
Traduction jeu Traduire les strings	Serveur	2020-03-23	3h
Traduction jeu Traduire les joueurs virtuels	Serveur	2020-03-24	6h
Traduction jeu Interface ajout menu langue	Client lourd	2020-03-23	3h
Traduction jeu Traduction des strings	Client lourd	2020-03-24	6h

Traduction jeu Interface ajout menu langue	Client léger	2020-03-23	3h
Traduction jeu Traduire les strings	Client léger	2020-03-24	6h
Trophées (Achievements) Implémentation achievement service	Serveur	2020-03-24	6h
Trophées (Achievements) Update REST profile pour inclure les achievements	Serveur	2020-03-24	3h
Trophées (Achievements) Interface profile achievements & REST	Client lourd	2020-03-25	3h
Trophées (Achievements) In game achievements popup	Client lourd	2020-03-25	3h
Trophées (Achievements) Interface profile achievements & REST	Client léger	2020-03-25	3h
Trophées (Achievements) In game achievements popup	Client léger	2020-03-25	3h

Tableau 4.8 - Bug fix v1.6			
Nom de la tâche	Plateforme	Date limite	Charge de travail
Assurance qualité	Serveur	2020-04-02	15h
Assurance qualité	Client lourd	2020-04-02	15h
Assurance qualité	Client léger	2020-04-02	15h
Mise à jour de la documentation	Documents	2020-04-05	20h

5. Équipe de développement

L'équipe est composée de six étudiants en génie logiciel à Polytechnique Montréal. Leurs responsabilités et leurs expertises sont indiquées dans le tableau ci-dessous. L'équipe est hautement qualifiée pour le développement du logiciel *Fais-moi un dessin* puisqu'elle est composée de membres ayant une expérience très variée et ayant des compétences avec chaque technologie utilisée pour le logiciel. De plus, l'équipe a une haute cohésion et est déterminée à produire un logiciel de haute qualité.

Tableau 5.1 - Équipe de développement		
Membre	Expertise	Responsabilités
Martin Pouliot	<ul style="list-style-type: none"> • Développeur web • Expérience serveur en C# • Expérience en DevOps • Gestion de serveur et infrastructure 	<ul style="list-style-type: none"> • Responsable du serveur • Responsable du déploiement sur une machine virtuelle dans Azure
Samuel Saito-Gagné	<ul style="list-style-type: none"> • Expérience avec Kotlin. • Expérience en conception d'interface utilisateur. • Connaissance de base de développement mobile. 	<ul style="list-style-type: none"> • Responsable en partie du mobile.
Allan Beddouk	<ul style="list-style-type: none"> • Expérience en conception d'interface utilisateur. • Connaissance de base de développement Android. • Développement avec NodeJs et Flask. 	<ul style="list-style-type: none"> • Responsable en partie du serveur
Philippe Côté-Morneault	<ul style="list-style-type: none"> • Expérience de développement d'application Android. • Expérience de développement de serveur avec Node.JS et Flask. 	<ul style="list-style-type: none"> • Responsable en partie de l'application mobile.
Pascal-Alexandre Morel	<ul style="list-style-type: none"> • Expérience avec les interfaces Angular et l'outil Angular Material • Expérience avec les serveurs Python (Django) et Node • Stage en développement Web 	<ul style="list-style-type: none"> • Responsable en partie de l'application .NET.
Cédric Tessier	<ul style="list-style-type: none"> • Expérience en WPF et Winforms • Expérience en C# • Expérience avec Visual Studio 	<ul style="list-style-type: none"> • Responsable en partie de .NET.

6. Entente contractuelle proposée

6.1 Définitions

Le client:

Le client est Polytechnique Montréal, représenté par M. Olivier Gendreau et M. Nikolay Radoev.

Le soumissionnaire :

Le soumissionnaire est l'équipe 203, une entreprise de génie logiciel dûment incorporée dans la région de Montréal.

6.2 Objet de l'entente

La présente entente a pour but de définir les conditions d'exécution du logiciel *Fais-moi un dessin*.

6.3 Durée de l'entente

La présente entente prend échéance lors de la remise de la totalité du projet, le 13 avril 2020. De plus, le client doit accepter le produit final selon les exigences et spécifications convenues dans la présente entente pour que l'entente arrive à échéance.

6.4 Rémunération

Le type de contrat visé par la présente entente est à prix ferme. Le montant total de la présente entente est décrit ci-dessous. Les modalités de paiement et du contrat sont définies dans les sections suivantes.

Le montant total réclamé par le soumissionnaire est basé sur des taux horaires pour les divers employés du soumissionnaire ainsi qu'une estimation des heures nécessaires pour la complétion du projet.

Les taux horaires des employés du soumissionnaire sont comme suit:

- Développeur: 100\$/h
- Gestionnaire de projet: 125\$/h

Considérant l'expertise de plusieurs employés du soumissionnaire et la date de livraison du projet, certains développeurs agiront en tant que gestionnaires de projet pour certains lots de travail. Ainsi, le taux horaire des employés dépend du type de tâche qu'ils exécutent.

Selon l'échéancier établi par le soumissionnaire, le temps de travail est estimé à 1080h. On estime 3h par semaine pour la gestion. Ces heures sont décomposées selon le type d'employé les exécutant comme suit:

- 1040h de développeurs
- 40h de gestionnaires de projet

Le montant total réclamé pour l'exécution du projet visé par la présente entente est 109 000\$. Ce montant est divisé comme suit:

- Salaires des développeurs: 104 000\$
- Salaires des gestionnaires de projet: 5000\$

6.5 Modalités de paiement

Le soumissionnaire exige un dépôt de 50% de la somme totale lors de la signature de la présente entente et le dernier 50% lors de la livraison du produit final et l'acceptation du client. La maintenance du présent logiciel n'est pas incluse dans ce contrat. Le client assume toute responsabilité de la maintenance après la date de livraison.

6.6 Modification des exigences

Toute demande de modification par le client est sujette à négociation. Une modification des exigences pourra entraîner une modification de la date de livraison des livrables et un coût supplémentaire. La modification des exigences doit se faire selon la section 3.1 du présent document «Gestion des exigences». Cette procédure a pour but d'informer

toutes les parties dans le cas d'un changement et de venir à un commun accord.

6.7 Propriété intellectuelle

Tout artéfact, code source et documentation produite par le soumissionnaire appartient à celui-ci. Le soumissionnaire autorise le client à utiliser les résultats du projet pour des fins d'enseignements et de formation à l'École Polytechnique de Montréal.

6.8 Résiliation du contrat

Advenant faute de paiement de la part du client, le soumissionnaire se réserve le droit d'arrêter les travaux, et ce, pour une durée indéterminée. De plus, le soumissionnaire se réserve le droit de résilier le contrat advenant faute de paiement de la part du client. Le soumissionnaire autorise le client à résilier le contrat advenant faute de livraison du logiciel complet à la date convenue au point 3 de la présente entente.

6.9 Responsabilité post-livraison

Le soumissionnaire n'a aucune responsabilité envers le client suite à la livraison du logiciel final. Le soumissionnaire ne fournit pas l'hébergement du serveur suite à la livraison du logiciel final.

6.10 Signatures

Martin Pouliot
Martin Pouliot
Développeur

Le vendredi 7 février 2020
Date

Samuel Saito-Gagné
Samuel Saito-Gagné
Développeur

Le vendredi 7 février 2020
Date

Philippe Côté-Morneault
Philippe Côté-Morneault
Développeur

Le vendredi 7 février 2020
Date

Allan Beddouk
Allan Beddouk
Développeur

Le vendredi 7 février 2020
Date

Pascal-Alexandre Morel
Pascal-Alexandre Morel
Développeur

Le vendredi 7 février 2020
Date

Cédric Tessier
Cédric Tessier
Développeur

Le vendredi 7 février 2020
Date

7. Annexes

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors	January					2020 February					2020 March					2020 April								
							07	12	17	22	27	01	06	11	16	21	26	02	07	12	17	22	27	01	06	11	16			
1	🚀	Documentation	27 days?	Wed 20-01-15	Mon 20-02-10																									
2	🚀	Plan de projet	5 days?	Wed 20-01-15	Sun 20-01-19																									
3	🚀	Gestion et suivi de l'avancement	4 days	Wed 20-01-15	Sat 20-01-18																									
4	🚀	Section 5,6	4 days	Wed 20-01-15	Sat 20-01-18																									
5	🚀	Introduction/ Énoncé des travaux	4 days	Wed 20-01-15	Sat 20-01-18																									
6	🚀	SRS	11 days	Thu 20-01-16	Sun 20-01-26																									
7	🚀	Introduction	4 days	Thu 20-01-16	Sun 20-01-19																									
8	🚀	Description	2 days	Fri 20-01-17	Sat 20-01-18																									
9	🚀	Exigences fonctionnelles	5 days	Mon 20-01-20	Fri 20-01-24																									
10	🚀	Exigences non fonctionnelles	5 days	Mon 20-01-20	Fri 20-01-24																									
11	🚀	QA	2 days	Sat 20-01-25	Sun 20-01-26	7,8,9,10																								
12	🚀	Plan d'architecture	6 days	Wed 20-01-29	Mon 20-02-03																									
13	🚀	Vues	4 days	Wed 20-01-29	Sat 20-02-01																									
14	🚀	Introduction / Objectifs et contraintes architecturaux / Taille et performance	3 days	Wed 20-01-29	Fri 20-01-31																									
15	🚀	Protocole de communication	16 days	Wed 20-01-15	Thu 20-01-30																									
16	🚀	Introduction	2 days	Sun 20-01-26	Mon 20-01-27																									
17	🚀	Partie pour prototype	4 days	Wed 20-01-15	Sat 20-01-18																									
18	🚀	Stateless	4 days	Mon 20-01-27	Thu 20-01-30	17																								
19	🚀	Temps réel	4 days	Mon 20-01-27	Thu 20-01-30	17																								
20	🚀	Plan de tests	16 days	Sun 20-01-26	Mon 20-02-10																									
21	🚀	Introduction/Exigences	6 days	Sun 20-01-26	Fri 20-01-31																									
22	🚀	3.1.1-3.1.5	5 days	Sat 20-02-01	Wed 20-02-05																									
23	🚀	3.1.6 - 3.1.11	5 days	Sat 20-02-01	Wed 20-02-05																									
24	🚀	Ressources / jalons (Fin document)	2 days	Fri 20-02-07	Sat 20-02-08																									
25	🚀	Développement	89 days?	Wed 20-01-15	Sun 20-04-12																									
26	👉	Prototype v1.0	18 days?	Wed 20-01-15	Sat 20-02-01																									

Project: Echeancier-Principal-v3
Date: Fri 20-01-31

Task

Split

Milestone

Summary

.....

◆

◆

Project Summary

Inactive Task

Inactive Milestone

Inactive Summary

Manual Task

Duration-only

Manual Summary Rollup

Manual Summary

Start-only

Finish-only

External Tasks

External Milestone

Deadline

Progress

Manual Progress

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors																												
27		Android	18 days	Wed 20-01-15	Sat 20-02-01																													
28		Interface connexion	2 days	Wed 20-01-15	Thu 20-01-16																													
29		Interface chat	2 days	Fri 20-01-17	Sat 20-01-18	28																												
30		Partie réseau	13 days	Mon 20-01-20	Sat 20-02-01	32																												
31		Server	11 days?	Sat 20-01-18	Tue 20-01-28																													
32		Temps réel	2 days	Sat 20-01-18	Sun 20-01-19																													
33		Gestion utilisateur	2 days	Mon 20-01-20	Tue 20-01-21	32																												
34		Envoi message	7 days	Wed 20-01-22	Tue 20-01-28	33																												
35		Connexion BD	1 day?	Sun 20-01-19	Sun 20-01-19																													
36		Gestion compte REST	2 days	Mon 20-01-20	Tue 20-01-21	35																												
37		Desktop	18 days	Wed 20-01-15	Sat 20-02-01																													
38		Interface connexion	2 days	Wed 20-01-15	Thu 20-01-16																													
39		Interface chat	2 days	Fri 20-01-17	Sat 20-01-18	38																												
40		Partie réseau	13 days	Mon 20-01-20	Sat 20-02-01	32																												
41		Prototype v1.1	4 days	Sun 20-02-02	Wed 20-02-05	26																												
42		Android tests	4 days	Sun 20-02-02	Wed 20-02-05	27																												
43		Desktop tests	4 days	Sun 20-02-02	Wed 20-02-05	37																												
44		Server tests	2 days	Sun 20-02-02	Mon 20-02-03	31																												
45		Release v1.2	9 days?	Thu 20-02-06	Fri 20-02-14	41																												
46		Clavardage historique	5 days?	Mon 20-02-10	Fri 20-02-14	52																												
47		Serveur Call REST et BD messages	2 days	Mon 20-02-10	Tue 20-02-11																													
48		Desktop implémentation REST	1 day?	Wed 20-02-12	Wed 20-02-12	47																												
49		Desktop Historique UI	3 days	Wed 20-02-12	Fri 20-02-14	48SS																												
50		Android Implémentation REST	1 day?	Wed 20-02-12	Wed 20-02-12	47																												
51		Android Historique UI	3 days	Wed 20-02-12	Fri 20-02-14	50SS																												
52		Clavardage rooms	4 days?	Thu 20-02-06	Sun 20-02-09																													
53		Server BD Channel et REST	1 day?	Thu 20-02-06	Thu 20-02-06																													
Project: Echeancier-Principal-v3 Date: Fri 20-01-31		Task		Project Summary		Manual Task		Start-only		Deadline																								
		Split		Inactive Task		Duration-only		Finish-only		Progress																								
		Milestone		Inactive Milestone		Manual Summary Rollup		External Tasks		Manual Progress																								
		Summary		Inactive Summary		Manual Summary		External Milestone																										
Page 2																																		

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors	2020 January					2020 February					2020 March					2020 April					
							07	12	17	22	27	01	06	11	16	21	26	02	07	12	17	22	27	01	06	11	16
54		Server Socket routage message & notification de subscribe/création	2 days	Thu 20-02-06	Fri 20-02-07	53SS																					
55		Desktop Implémentation UI Rooms	2 days	Thu 20-02-06	Fri 20-02-07																						
56		Desktop Implémentation REST et socket chat	2 days	Sat 20-02-08	Sun 20-02-09	55SS,54																					
57		Android Implémentation UI Rooms	2 days	Thu 20-02-06	Fri 20-02-07																						
58		Android Implémentation Rest et Socket chat	2 days	Sat 20-02-08	Sun 20-02-09	57SS,54																					
59		Profil	3 days?	Thu 20-02-06	Sat 20-02-08																						
60		Server BD et REST	2 days	Thu 20-02-06	Fri 20-02-07																						
61		Desktop Interface profil	2 days	Thu 20-02-06	Fri 20-02-07																						
62		Desktop REST profile	1 day?	Sat 20-02-08	Sat 20-02-08	61SS,60																					
63		Android Interface profile	2 days	Thu 20-02-06	Fri 20-02-07																						
64		Android REST profil	1 day?	Sat 20-02-08	Sat 20-02-08	63SS,60																					
65		Statistiques	3 days?	Sun 20-02-09	Tue 20-02-11	59																					
66		Server BD et REST	2 days	Sun 20-02-09	Mon 20-02-10																						
67		Desktop Interface statistiques/profil	1 day?	Sun 20-02-09	Sun 20-02-09																						
68		Desktop REST statistiques	1 day?	Tue 20-02-11	Tue 20-02-11	66,67SS																					
69		Android Interface statistiques/profil	1 day?	Sun 20-02-09	Sun 20-02-09																						
70		Android REST statistiques	1 day?	Tue 20-02-11	Tue 20-02-11	66,69SS																					
71		Historique parties	2 days?	Wed 20-02-12	Thu 20-02-13	65																					
72		Server BD et REST historique	1 day?	Wed 20-02-12	Wed 20-02-12																						
73		Server Historique Hooks	1 day?	Thu 20-02-13	Thu 20-02-13	72																					
74		Desktop REST historique	1 day?	Thu 20-02-13	Thu 20-02-13	75SS,72																					
75		Desktop Interface historique	1 day?	Wed 20-02-12	Wed 20-02-12																						
76		Client REST historique	1 day?	Wed 20-02-12	Wed 20-02-12																						
77		Client Interface historique	1 day?	Thu 20-02-13	Thu 20-02-13	76SS,72																					
78		Création de compte	2 days?	Sun 20-02-09	Mon 20-02-10	59																					
79		Server BD et REST	1 day?	Sun 20-02-09	Sun 20-02-09																						
80		Desktop Interface register & REST	2 days	Sun 20-02-09	Mon 20-02-10	79SS																					
81		Android Interface register & REST	2 days	Sun 20-02-09	Mon 20-02-10	79SS																					

Project: Echeancier-Principal-v3

Date: Fri 20-01-31

Task

Split

Milestone

Summary

Project Summary

Inactive Task

Inactive Milestone

Inactive Summary

Manual Task

Duration-only

Manual Summary Rollup

Manual Summary

Start-only

Finish-only

External Tasks

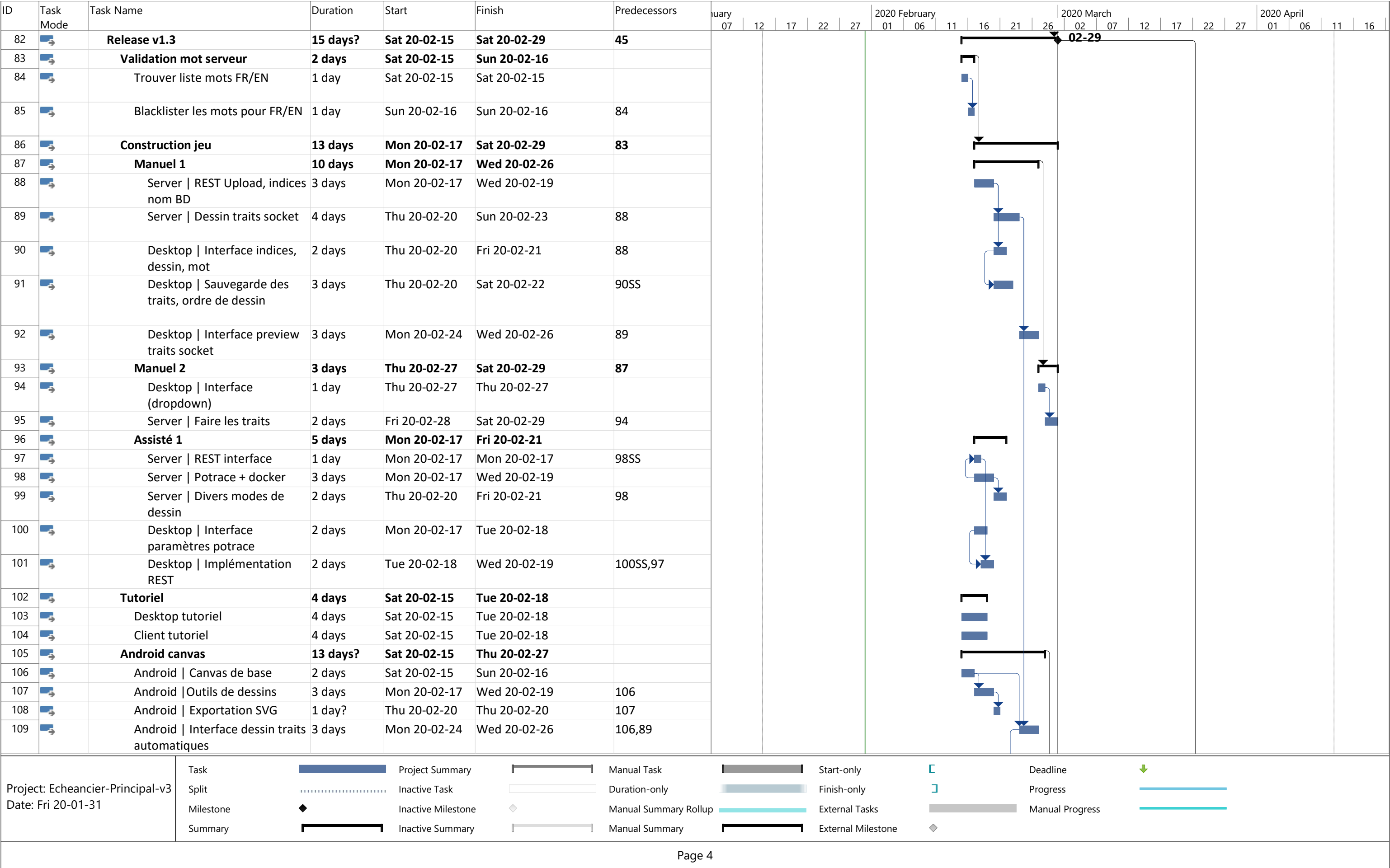
External Milestone

Deadline

Progress

Manual Progress

Page 3



Deadline

Progress

Manual Progress

Page 4

