

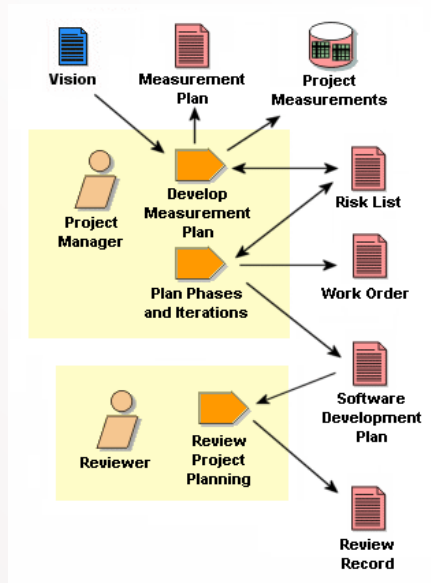


Conduite de Projets

Ingénieurs 3^o année

III c – Plan Projet (SDP)

1 – Introduction



Rôle et nécessité du SDP

Rôle du Plan Projet

❑ Le Plan Projet

- rassemble **toutes les informations** requises pour le **management du projet**
- permet de prendre en temps voulu les décisions nécessaires.

❑ Il est souvent nommé Software Development Plan (SDP – terme UP)

- Synonymes:
 - ✓ Plan de Management
 - ✓ Plan de développement
 - ✓ Plan méthodologique

❑ Il définit précisément :

- Les principes clés de la méthode de conduite du projet
- L'organisation générale du projet : rôle et responsabilités de chaque instance (comité de pilotage, groupes utilisateurs, direction de projet, MOA, MOE, ...)
- La listes des livrables et le mode de gestion de la documentation
- Les aspects budgétaires et le suivi des dépenses
- Les contrôles de l'état du projet : revues et audits
- Les ressources nécessaires
- Les guidelines destinés aux membres de l'équipe
- ...

❑ Il permet ainsi la maîtrise des risques, des performances, des coûts, des délais et donne l'assurance de la qualité.

Nécessité du Plan Projet ?

- ❑ La plupart des méthodes imposent sa rédaction
 - Les méthodes traditionnelles
 - Les méthodes orientées « Grand projet »
- ❑ Les méthodes agiles ne le préconisent pas (formalisme trop lourd/activité improductive)
- ❑ Un plan projet léger peut cependant s'avérer utile pour clarifier les choix méthodologiques de l'équipe
 - Notamment pour les équipes novices en matière d'agilité

2 – Rubriques du SDP

70-20-10 Development Plan		
Competency	Organization Politics	
Target Skill	Influence without authority	
Target Skill Level 3	<ul style="list-style-type: none">• Works well across teams to facilitate identifying common concerns	
3 on-the-job activities	2 people to ask for help	1 formal learning activity
<ul style="list-style-type: none">• Complete a stakeholder profile (influence x interest)	<ul style="list-style-type: none">• Seek out a mentor who is exceptional at influence	<ul style="list-style-type: none">• Read a good book on network-building
<ul style="list-style-type: none">• Review charters for projects other than your own for common goals	<ul style="list-style-type: none">• Brainstorm with a manager or project manager about possible common interests among concurrent project.	
<ul style="list-style-type: none">• Starting from the network of own stakeholders,		

Principes méthodologiques

- ☐ Présenter les principes clés de la méthode de conduite de projet employée
- ☐ Justifier le choix de cette méthode
- ☐ Expliquer la façon dont elle sera mise en œuvre

- ❑ Pour rappel les méthodes agiles ne sont qu'un cadre général
 - Des choix précis sont à instancier et justifier
 - Ne jamais appliquer un principe méthodologique aveuglément
 - Chaque pratique ne doit être retenue que si elle apporte quelque chose au projet

Toujours adapter la méthode au contexte

➔ *Relire son SDP et se demander objectivement si ce qui est écrit est « passe-partout » ou spécifique à ce projet...*

Exemples de pratiques à instancier

❑ Utilisation du PO

- Qui est le PO ?
- Quel rôle lui faire jouer? Parmi les tâches classiques d'un PO
- Quelle est sa charge de travail ?
- Est-il nécessaire/possible de répartir ses tâches sur plusieurs PO ?
- Quels créneaux pour les réunions avec lui ?

➔ Tout ceci est à établir avec grande précision

❑ Validation des US/Recette

- Qui en est responsable ? TM, PO, Key user, ...
- A quelle fréquence les séances de validation ont-elles lieu ? Question cruciale si le PO n'est pas dispo à 100%
 - ✓ A chaque completion d'US ?
 - ✓ Une fois par semaine ?
 - ✓ Lors de la Iteration Review ?
 - ✓ ...
- Dans quels créneaux ?
- Quelle charge de travail ce travail représente-t-il ?

❑ Organisation des réunions en interne

- Qui en prend l'initiative ? SM, TM ?
- Existence d'un Ordre du jour publié ?
- Time keeping durant la réunion ?
- CR disponible à l'issue de la réunion ?

❑ Cas particuliers

- Les réunions de synchronisation entre TM (US connexes, composants partagés, ...): quand ont-elles lieu ? Quelle durée ?
- Les sessions de planification (release et itération) et clôture: idem

❑ Couverture de tests

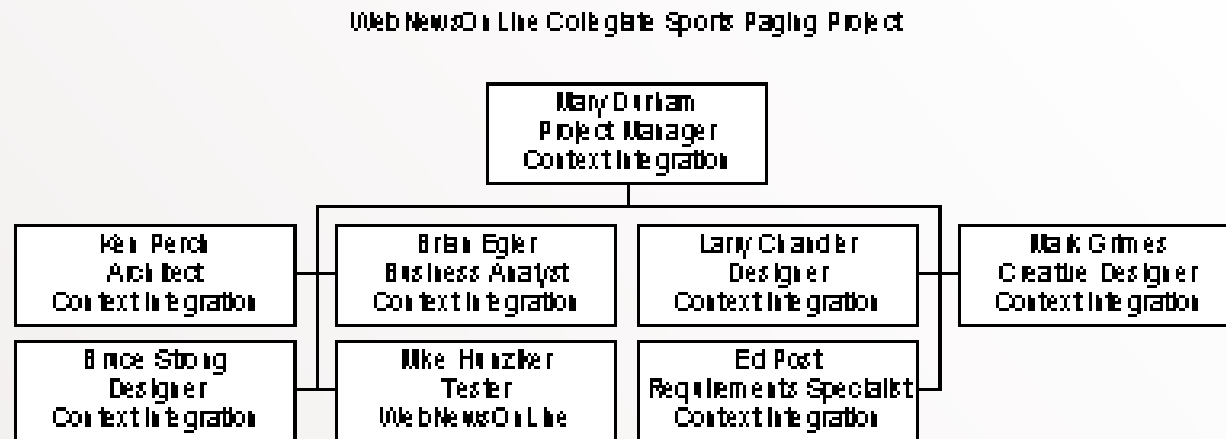
- Combien de TU/TA par US ?
- Existence de TI ?
- Automatisation ?

❑ Fréquence des builds

- Automatiques ?
- Builds manuels additionnels ?

Rôles et responsabilités dans l'équipe

- ❑ Organigramme de l'équipe projet



✓ NB: à priori inutile en agile

❑ Rôles et responsabilités

Role	Responsibility
Project Manager	The Project Manager allocates resources, shapes priorities, coordinates
Architect	The Architect leads and coordinates technical activities and
Business Analyst	The business analyst leads and coordinates business use-case modeling, by outlining and delimiting the organization being modeled. For example, establishing what business actors and business use cases exist and how they interact.

❑ En agile

- Les TM ont-ils un rôle spécifique ? Eg: référent sur telle technologie ou activité
- Les TM sont-ils totalement polyvalents ?
- Le rôle de SM est-il tenu par les TM en rotation ?
- Pair programming, travaux en binômes ?

Livrables

- ❑ Lister les livrables produits durant le projet

→ Cf document « **Livrables du Sof. Dev. Plan.xls** »

- Cette liste est évidemment variable d'un projet à l'autre et doit être strictement adaptée aux besoins du projet
- En agile la liste est beaucoup plus légère mais existe nécessairement

Budget et Dépenses

Budget

C-Registration Project - Budget for Releases 1 & 2		
Labor		
Activities	Effort (PDs)	Cost
Business Modelling	45	\$31,500
Requirements	90	\$63,000
Analysis & Design	130	\$91,000
Implementation	206	\$144,200
Test	140	\$98,000
Management	80	\$56,000
Deployment	50	\$35,000
Environment	90	\$63,000
<i>Total Effort:</i>	831	
	TOTAL LABOR:	\$581,700
NonLabor (attach supporting details)		
Travel and Accommodation:		0
Freight & Duty:		0
Services:		6000
Materials:		32000
Other Direct Charges:		7500
	TOTAL NONLABOR:	\$45,500
	TOTAL BUDGET:	\$627,200

✓ Peut être adapté en agile avec l'usage des SP

Ressources humaines nécessaires

- ❑ Identifier les ressources internes allouées au projet (référence aux organigrammes) et leur compétences
- ❑ Lacunes : lister les manques en ressources internes (ex : manque de développeurs et concepteurs pour atteindre les objectifs du projet)
- ❑ Acquisition de ressources prévues (ex : recrutement d'un développeur sénior ayant plusieurs années d'expérience en C#, 2 développeurs juniors,)
- ❑ Plan de formation prévu (ex : avant le démarrage du projet, les équipes suivront un cours de Méthodologie et Conduite de Projets, un cours d'Analyse et Conception Orientée Objets,)