4.2 相手を知る＝相手の人物モデルを自分の中に持つ

コミュニケーションを効果的に行うためには、相手に合わせることが重要です。 相手に合わせるためには、相手を知らなければいけません。 では、相手を知るということはどういうことでしょうか？

【ヒント】 相手を知るとは、相手の「人物モデル」を自分の中に持つこと。

それは相手の人物モデルを自分の中に持つことだと思います。 人物モデルは、以下のような情報で構成されるでしょう。

- 状況
- 価値観
- 思考法
- 期待
- その他いろいろ(趣味、志向、特技、人柄、家庭、・・・)

相手の名前も知らないという状況でのコミュニケーションが、どれだけ一方的なのかを理解できるでしょうか。 アジャイル開発を上手く運営するノウハウの一つとして、最初に飲み会やランチ会を実施するというものがあります。 これはチームの各メンバの人物モデルを相互に作り合うことに非常に役立ちます。 相手の趣味や家庭を知ることも人物モデルの質を上げることに役立つことがあります。

■ コミュニケーションが苦手という担当者もいます。

日本では文系・理系という区別があり、その決定(自己判断理由)の一つにコミュニケーションが苦手であることを理由にしてきた場合があります。 また、プログラマという職業を選択した際にもコミュニケーション(が苦手)という理由があったかもしれません。メンバにはそういう気持ちの人も含まれているかもしれないし、あなた自身がそうかもしれません。

けれども、アジャイル開発のメンバとして活躍するにはコミュニケーション・スキルが重要です。 ・・・と言っても、コミュニケーションとして単に口数が多いこと、難しい日本語表現を使いこなすことを、を求めている訳ではありません。 まずは、相手に興味を持って人物モデルを持ち、相手を理解していくというコミュニケーション**姿勢**を持つようにしましょう。 コミュニケーションは徐々に成長するものです。 少しずつ相手の人物モデルへの対応について成功体験を積み上げていきましょう。

■ あなたの情報・判断理由を伝えて、あなたの人物モデルを正しく持ってもらいましょう。

各判断に対して、あなたの判断の理由(状況、価値観、思考法、期待)などを含めて伝えていきましょう。 皆が正しい判断を出せるだけの情報を適切に与えて、できるだけ自立性を高めてもらうように、あせらず徐々に導きましょう。 小さい成功のループを何度もまわすことが近道です。

■ あなたの思考が、論理的・合理的であり、一貫性を持っていると、相手は早く理解してくれます。

思考の論理性・合理性は、コミュニケーションにも表れます。 論理性があれば、前提や結論などをシンプルに語ることができます。 コミュニケーションが苦手という原因の中に、論理的な思考が苦手な場合も含まれますが、その場合には、会話以前に論理的な思考について習熟していきましょう。

■ 話をする際は、前提が合っていることをまず確認しましょう。

相手が今、どんな文脈にいるかを考えてみましょう。 結論を先に述べるのは正しいスタイルですが、そもそも話の前提は合っているでしょうか。 こそあど言葉(あれ、これ、それ、どれ)は、受け取る人の状況で誤解されるかもしれません。 議論の最初はできるだけ具体的な言葉・用語を使って、前提を合わせることを意識してコミュニケーションを進めましょう。

4.3 重要なことを先に。段階的に詳細化し相互理解する

■ 重要なことを最初に短く伝えて、その後に詳細説明・質問・回答等を組み合わせて理解を深めていきましょう。

■ それは普通のビジネス・シーンのコミュニケーション・スタイルと一緒です。

「重要なことを最初に短く伝えること」、「前提を伝え、報告なのか／相談なのか、を伝え、 まず結論を言うこと」、その後、「相手の反応を見て、詳細説明／質問／回答等を組み合わせて、相互に理解を深めること」、これらは普通にビジネス・シーンで求められるコミュニケーション・スタイルです。 その際には相手の人物モデルを使いましょう。 TPO は大切です。 たとえば、相手が５分後に会議があると表明しているのに、回りくどく詳細から話すのは良くありませ。 ネガティブな対応をされてしまうでしょう。 一方で、そんな状況であっても商用サービスの停止などの緊急情報であれば、相手の価値観は状況の連報を受けることを、会議より優先するでしょう。

4.4 モチベーションを上げるために（PO 視点）

コミュニケーションの狙いはいくつかありますが、モチベーションを維持し、上げることにつながることが重要です。

■ モチベーションが高いと、積極性の向上、主体性の発揮につながります。

モチベーションを上げると、開発に積極的に協力してもらえるようになります。 モチベーションが高く、技術スキルの高いメンバによる開発は、非常に前向きで強力なものになります。

アジャイル開発の弱点の一つとして全ての判断が PO に集中し、PO が忙殺されてしまうことがあります。 メンバの主体性が高まると、その状況を緩和することが可能になります。

■ 褒めることが出来ていますか？ 感謝することが出来ていますか？

褒めることは、モチベーションを上げます。 その判断が良かった・正しかったことをフィードバックすることで、主体性を高めることにつながります。 感謝することもそれと同様です。 コミュニケーションは、人によっては面倒なことです。 ドキュメントを書いてフォルダに置くだけにしたいのかもしれません。 そんなや、主体性を発揮して相談してくれたことに対して、ぜひ、日常的に褒めて感謝する関係を作り育てていきましょう。

■ モチベーションを下げることは一瞬で起こります。

モチベーションを上げるのが徐々にしかできないのに対し、下げることは一瞬でできてしまいます。 以下のような状態が発生すると、PO への信頼感が低下し、モチベーションが下がります。 コミュニケーションを丁寧に実施して回避していきましょう。

- Win-Win の意識がなく、常に上から目線で指示をする。
- 自らを省みず、一方的にチームメンバの責任を問う。
- 予実管理(予定と実績の乖離分析)が大好きで、乖離を責める。
- 急がせておきながら、話が長くミーティングが大好き。
- PO 自身に主体性がなく、いつも他人やステークホルダの言いなり。
- チームの現状を知っていながら、ステークホルダと勝手な約束をしてくる。
- 現場が納得できない計画で進めようとする。
- 優先順位の決定理由や変更理由が伝えられず、これまでの作業を無駄にする理由が理解できない。
- 皆が何のために作業しているのか(ゴールが)わからなくなっている。
- 大きな声で怒鳴り散らすだけで解決に一切役立たない。

■ 方針変更や仕様変更の際には特に丁寧に理由を説明しましょう。

アジャイル開発では仕様の変更や優先度変更が発生します。 場合によっては朝礼暮改のように見えることもあるでしょう。 PO の信頼が揺らぐのは防がなければいけません。 方針変更や仕様変更の際はその理由や背景の説明を積極的に行い、その必然性や重要性を理解してもらえるように尽くしましょう。

4.5 チームの質も上げよう（PO 視点）

チーム内でのコミュニケーションが上手く成立していると、チーム力が高まります。

■ チームとしての技術的能力を上げるコミュニケーションを促しましょう。

コミュニケーションは PO と開発メンバとが行うだけではありません。 開発メンバの間でも積極的な技術相談等が成立するように誘導し、活性化を見守らなければいけません。 人毎に偏る専門性が上手くチーム全体で生かせればチームとしての能力が一層向上します。

■ チームとしての可用性を上げるコミュニケーションを促しましょう。

ある開発メンバの作業が遅れた時や突然風邪で休んだ時、残りの開発メンバが主体的にそれをフォローできる体制になることが理想です。 そのためにも、自分の状況は PO だけに伝えるのではなく、チーム内に広く伝えるように促しましょう。

4.6 悪い情報を素早く収集する仕組みを作って、大失敗を回避する（PO 視点）

アジャイル開発において PO が判断をする際に、正しい開発現場の情報を知っていることが重要です。 特に、悪い情報には早く対応しなければいけません。 PO の責任ある良い優先度判断は、それらの情報の上に成立します。

■ 悪い情報・問題は、早めに気づいて対処した方が楽です。

悪い情報・発生した問題についても解決の最終責任は PO です。 そうであれば、問題は、早めに手を打った方が選択肢が多く、楽に解決できる可能性が高まります。

■ 悪い情報ほど、PO 自らが早く気づくように努力しましょう。

悪い情報ほど、早く報告して欲しいとお願いします。 しかし、悪い情報を報告することに慣れていない担当者や、以前悪い情報を報告した際に感情的な叱責を受けた担当者等がいれば、その願いはなかなか理解されないかもしれません。 そもそも心理的にも抵抗感があるはずです。

しかし、「現場の悪い情報がなかなか上がってこない」と、文句を言っていはいけません。 悪い情報がないのは PO は現場に探しに行かなければいけません。 積極的に検出する質問をしなければいけません。 これを「現場に Ping を打つ」と言うこともあります。 待ちの PO ではなく、自らが現場の担当者に対して、「心配事ない？」、「課題は無い？」、「待ちになっている作業はある？」、「無理にでも最も悪い情報を言うとすると、どんなこと？」と、できれば日々確認していきましょう。

■ 悪い情報の連報には「感謝」を表明しましょう。そして一緒に解決しましょう。

悪い情報が早く上がる仕組みを実現するためには何をすれば良いでしょうか？ 悪い情報が上がっても冷静に受け止めて、素々と紳士的に対処されることを示す必要があります。 悪い情報を責任者に伝えることは、慣れていない人にとっては苦痛なはずです。

悪い情報を、素早く共有してくれたことについては、開発メンバ全員の前で「感謝」を示しましょう。 悪い情報は、情報が確定しないうちにでも連報でもらえるようにしましょう。 情報が不正確であっても叱責してはいけません。 どこまでが事実で、どこからが推測なのか？それとも不明なのか？情報源はどこか？等を確認し、不足する情報について更に継続収集と精度の向上を依頼するか？あなた自身が直接事実確認に動くか？などの判断を行っていきましょう。

■ 次回以降も、情報が素早く入ってくることを奨励する態度が大切です。

絶対に、不正確な情報を報告したことを叱責してはいけません。 まずは「感謝」です。 さらに、PO が一緒に解決してくれることを示す必要があります。 それらを、続けることによって、各担当者は**安心して悪い情報を報告できる**ようになります。

世の中には、悪い報告に対して、紳士的ではなく激情的に反応するリーダもいたと思います。 そういう人が PO の場合、プロジェクトが大失敗する前に、アジャイル開発の採用はやめた方が良いと思います。

■ 報告が遅れた場合には、その理由を聞き、次回から早くなるように改善しましょう。

期待より報告が遅い場合には、その理由を確認し、次回からもっと早く報告できるような判断基準の変更を合意しましょう。 そもそも慣れていないのであれば、徐々に改善するように誘導しましょう。

■ 正直なチームの文化を築いて、信頼されるリーダになりましょう。

PO も神ではありません。 予測を含んだ判断では間違えることもあるでしょう。 そのときは隠蔽化さず素直に反省する態度を見せて、自らが正直なチーム文化を築いていく姿勢を貫きましょう。 他人に厳しく自分に甘いリーダでは、決して信頼されることはありません。

■ チームに対する自分の判断ミスは、公表し、模範となる反省の態度を見せましょう。

同様に、あなた自身が、悪い情報をできるだけ早めにステークホルダに共有する姿勢を持つことも、ステークホルダとの信頼関係を強化することに役立つでしょう。
正直が開発チームの文化であることを示しましょう。

4.7 ウォーターフォール型コミュニケーションの悪癖

■ 発注側は受注側のメンバと対等なチーム・コミュニケーションを実現しましょう。

ウォーターフォール開発で育った人材で、発注側が上、受注側が下という意識が強く残っている場合、アジャイル開発ではコミュニケーションの質を下げてしまうことがあります。 相手のモチベーションを常に意識し、理解度(言った、書いた、ではなく、伝わったか)も確認して進めなければいけません。

■ 「まずは、報告書としてまとめさせてください」と言って、時間をかけることは望ましくありません。

報告書としてまとめる前に、するべき会話がさっとあります。 スピードアップのために、合意してから記録するスタイルを目指しましょう。

■ 受注側は、端的に素直なミュニケーションを実施しましょう。悪い情報も素早く共有しましょう。

お客様を刺激しない曖昧なコミュニケーションは駄目です。 端的に伝える努力をしましょう。 違憲して丁寧語を使いすぎて、結局、何を言いたいのかわからない会話は駄目です。 「はい」は、「Yes」、「正しい」、「同意」に使うべきです。 「やるのか」、「やらないのか」、「情報不足で判断できないのか」も曖昧では困ります。 「できるのか」、「できると思っているのか」、「可能性をチャレンジしてみるのか」、「できない

と思うけど仕方なくやるのか」、「やることに反対なのか」、「理解できないのか」についても端的に伝えるべきです。

お客様にとって耳障りな情報であっても開発プロジェクトにとって悪い情報は速やかに伝えなければいけません。 お客様のゴールや価値観は意識しつつ、チームメンバとして対等な関係で議論・提案する姿勢が必要です。

5. 現場でのアジャイル開発スタイル（PO 実践編）

発注者は開発者から信頼された仲間になるべし。
ゴールを示し、皆に理解してもらってチームを作り、
一人ひとりを知って適材適所に配し、
状況・問題点・今後の方針を積極的に聞き、対外調整を実施し、
丁寧な対話とバランスの良い優先度判断を常に心がけ、
仲間であり人格者のリーダとして信頼されるようになり、
仲間とともに持続的な Win-Win と成長を目指すべし。

5.1 技術者を集める

良いソフトウェアを作るために、１章で示した【優良ソフトウェア開発原理】が重要です。 優秀な技術者を集めて/育てて、チームを作り、高いモチベーションを実現し、持続させていかなければいけません。

■ 良い技術者を集めることが重要です。

チームメンバに期待する能力／意欲／姿勢、条件にはどんなものがあるでしょうか？ 以下に例を示します。

- 優秀なソフトウェア技術力(設計～コーディング・試験まで全て)
- 相手に理解してもらえるコミュニケーションの基礎的な能力
- 開発システムへの興味とモチベーション(意欲)
- PO との協働姿勢、素直で正直
- 積極性・提案する姿勢
- 専念できる(他と兼務しない)

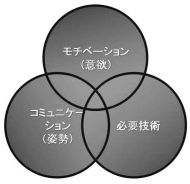

図 5-1 3つの大きな期待

■ 良い技術者はチームとして育てていきましょう。

少なくとも１名、可能なら２~３ 名は、設計からコーディング・試験まで、全ての工程に精通した技術リーダがアジャイル開発では必要でしょう。

しかし、全メンバが全ての条件を満足することは一般には困難です。 特に、短期間で体制を作る場合には条件を緩める必要があるでしょう。 不足する部分については、チーム内で成長を促し育てることが大事です。 将来必要になるスキルを早めに伝え、必要な調査時間・学習時間は業務に組み込むことが必要になるでしょう。 チームのメンバは相互に成長しあえるように、ノウハウを共有するように誘導しましょう。

■ それでもチームをまとめるために交代してもらうケースはあります。

以下にようにコミュニケーションの基礎からの育成が必要になる場合や、アジャイル開発の考え方に適合できない場合には、スキル不一致で交代することもやむを得ません。

- 真実を言わない。 事実と予想が区別できない。
- 声が小さすぎて聞き取れない。
- 指示がないと動かない。 黙って待っている。
- 論理的な思考・会話が困難である。
- 一度決めたことを変える事を拒絶する。

また、技術力は抜群であっても、それ以外の面でチームに悪影響を与えるメンバであることもあります。 相互不信に陥る前に交代を求めていきましょう。

- 仲間を攻撃する。 共に成長するという視点に欠ける。
- 反抗的な姿勢をとる。 ゴールに反対し続ける。

■ プロジェクトが長期にわたる場合、要員の交代にも備えましょう。

長期のプロジェクトでは技術的キーマンの入れ替えは当然発生します。 突然、キーマンを抜かれないための注意も必要です。 準委任契約は、人を個別に指定して契約することはできません。 受注会社でも人材育成とキーマンを次のポジションに移すことは避けられません。 受注会社としての人材育成に伴うキーマン交代計画は、早めに共有してもらって備えるようにしましょう。 他にも、不慮の事故、親や家族の介護、体調不良や病気など、交代が必要になるシーンはプロジェクトが長期化すれば、避けることは困難です。 備えることを考えなければいけません。

■ 属人化はアジャイル開発の弱点の一つです。

■ 自動テストの整備をして属人化リスクの低減を図りましょう。

アジャイル開発では、特定の技術的キーマンが発生することが普通です。 開発現場側のキーマンが一人に集中しないように PO も配慮しなければいけません。 重要な情報はキーマンの外に見える状態で残されていなければいけませんが、自動リグレッション・テストを整備することが属人化を避ける最大の対策となります。 維持管理や機能拡張を行っていく際にも大切です。 リグレッション・テストの自動化ができると開発プロジェクトは一旦安定状態に入ったと考えることができます。