

システム開発演習ガイド

Ver 1.05

1 プロジェクト

1.1 プロジェクト内のチームワーク

プロジェクトに対応するために作られる組織が、プロジェクトチームです。プロジェクトチームは複数の人で構成されます。

大きなプロジェクトになると、数千もの人が1つのプロジェクトに携わることもあります。したがって、プロジェクトが大きくなればなるほど、プロジェクトに参加している全員が共通の目標意識をもち、目標を達成するために、お互いに協力する姿勢が必要となります。つまり、チームワークが重要になってきます。

情報システム開発のプロジェクトにおいても、プロジェクトに参加するメンバー全員が同じ意識を持ち活動することが重要です。

- 高い品質の物を作り上げる
- コスト意識を持つ（無駄な費用を発生させない）
- 納期を厳守する（決められた期日に納品する）

1.2 プロジェクト管理

プロジェクトの目標を所定の期間と予算で達成するためには、各メンバーの成果物が目標を達成するのに十分な内容か、各メンバーが自分の作業を予定どおりに行っているか、さらに今までの作業に掛かったコストが予算内に収まっているかを、絶えず正確に把握しながらプロジェクトを推進する必要があります。このような活動をプロジェクト管理(プロジェクトマネジメント)と呼んでいます。プロジェクト管理は、プロジェクトのライフサイクル全体を管理します。

プロジェクト管理には、成果物の品質管理、作業の納期管理、作業に掛かったコスト管理などの要素が含まれます。

「管理」というと、リーダーの役目のような気がしますが、決してそうではありません。プロジェクトチームの全員がプロジェクト管理に参加しなければ、プロジェクトの目標を所定の期間と予算で達成することはできません。

さらに、最近のシステム開発は非常に大規模になり、成果物の品質や作業の納期、作業に掛かったコストをリーダーが一人で正確に把握することが難しくなっています。そのため、プロジェクトチームのメンバー一人一人が、プロジェクト管理に参加する必要があります。

- 自分の作業の品質、納期、コストを正確に把握する
- 現在の状況をリーダーや他のメンバーに連絡する
- 問題が発生する恐れがある場合には、すばやく報告や相談をする

1.3 作業の記録・共有

プロジェクトを推進する上で注意することがあります。それは、プロジェクトチーム内でのコミュニケーション、つまり情報伝達の問題です。

プロジェクトが失敗する要因の1つに、プロジェクトチーム内でのコミュニケーション不足が挙げられます。コミュニケーションが不十分なために起きたトラブルを挙げたらきりがありません。たとえば以下のような事例があります。

- メンバー全員にプロジェクトの目的が共有されず、目標意識が低い
- チーム内で共通認識がなく、間違った判断や誤解のもとにシステムを設計する
- 一部のメンバーにしか情報が共有されず、システム全体で不整合が生じる

こんな失敗をしないために、以下のことをプロジェクトチーム全員で心がけましょう。

情報をドキュメント(書類)として残し、プロジェクトメンバー全員が共有する。作業が忙しく少しでも時間が惜しくなってくると、簡易的かつスピーディーな口頭での情報伝達をしまいがちになります。しかし、口頭での情報伝達は、人から人へ伝達されていく過程で、だんだんと内容が変化したり、時間の経過とともに情報そのものが欠落したり、忘れられたりする危険性があります。プロジェクトチーム全員に情報伝達するためには、速く(迅速性)正しく(信頼性・正確性)伝えるだけでなく、残す(保存性)ことも必要です。

また、ドキュメントはプロジェクトチームの誰もがいつでも参照できるようにします。常に最新の情報が全員に伝わる事を意識しましょう。

2 システム開発演習の進め方

2.1 全体の流れ

皆さんは、3～5名で1つのプロジェクトチームを構成します。それぞれのプロジェクトチームは、独立した情報サービス会社に属します。

各会社は、顧客から受注したシステム（以下、「課題システム」という）を開発します。課題システムの詳細については、4章の「課題システム」を参照してください。

演習において学習者は、プロジェクトチーム単位で以下の作業を行います。

プロジェクト計画

課題システムの開発を行うために、プロジェクトチームを作ります。プロジェクトチームでは、与えられた要件を基に具体的なシステム開発の計画を立案します。

システム開発、プロジェクト管理

システム開発作業を行います。実施するシステム開発工程は、要件定義、設計、実装（プログラミング）、テストです。プロジェクト管理をしっかり行い、プロジェクトの目標を達成してください。

演習においての講師は、プロジェクトチームの上司の役割を果たします。それを前提として接するようにしましょう。

2.2 上司からの要件

今回のシステム開発にあたり、上司から皆さんに提示された要件を示します。

開発期間

新宿図書館から「納期の厳守を最優先としたい」との申し出があるとおり、当社の納品が遅れることは、お客様に多大なる迷惑をかけることになる。ABC システムの信用にかけても、納期は絶対に死守してほしい。

報告について

毎朝のチームミーティング終了後に、プロジェクトリーダーは、進捗状況と問題点、対策について必ず報告してほしい。

また、3時間に一度はレビューの時間を取り、上司にコードの確認や疑問点の解消に来てほしい。（現場では1日に1度程度のことも多くなりますが、演習のため頻度を多くします）

その他のタイミングでも、必要に応じて報告や相談をしてほしい。とにかく、早め早めの対応が、問題を極力小さく抑え、問題の解決を容易にする。

プロジェクトの進め方

プロジェクト成功の鍵の1つに、プロジェクトメンバー間の協力がある。プロジェクトメンバー相互で十分にコミュニケーションを取り、協力し合ってこそプロジェクト成功の基盤が整う。この基盤がなければ、いくら一人一人のメンバーの力が優れていてもその力を集中できず、決してよい成果は得られない。プロジェクトメンバーは、お互いに協力し合い、定められた期間内で目標を達成するために、全力を傾けてほしい。

レビューについて

以下のタイミングで開発者と上司の間でしっかりとしたレビューを行いたい。（上で記した3時間に一度の報告をこれに代えても構わない）

なお、レビューの実施において、自身が担当した箇所は、その本人がレビューの主催を行うものとする。

- ① プロジェクト計画（開発スケジュールぎめ）終了時
- ② 要件定義終了時
- ③ 外部設計終了時（完璧ではなくとも8割程度の完成で構わない）
- ③ 内部設計終了時（完璧ではなくとも8割程度の完成で構わない）
- ④ 実装終了時
- ⑤ テスト終了時

役割の分担について

プロジェクトをすすめるだけでなく、全員のスキルアップも同時に行うことが目的になっている。そのため、各人が1機能は必ず作成することを義務付ける。MVC の View だけを実施する人などが出ないようにタスクを分担すること。

また、チームで一番スキルがあるメンバーは自身がすべて作るという意気込みではなく、他のメンバーができるようにサポートし、自身も深く内容を理解していくというスタンスで望むこと。

2.3 プロジェクト発足

ここでは、プロジェクトの発足から始まり、どのようにプロジェクトを管理していくのかなどの、プロジェクトの進め方について説明をします。

プロジェクトの名前・体制・目標の決定

プロジェクトチーム内で討議し、プロジェクトの名前・体制・目標を決めます。目標はプロジェクトチーム全員で決めてください。全員の合意で決めることが、全員で守ることにつながります。

プロジェクト体制としては、以下に挙げる5つの役割をメンバーに割り振ってください(人数によって一人二役になる場合もあります)。各リーダーは、その役割に対する最終意思決定者です(決してリーダーがひとりでその役割を実施するわけではありません)。

・プロジェクトリーダー

定められた期間内に特定のゴールに到達するためのプロジェクト管理を行います。メンバーの進捗管理やチームミーティングの開催・進行、上司への報告などを行います。重要な課題について優先度の判断や担当者の割り振りも行います。

・テクニカルリーダー

チーム内の技術的な要望や課題について、解決のために検討や調査を行います。

・データベースリーダー

データベースの設計・管理、テストデータ等の作成、バックアップを行います。

・ドキュメントリーダー

プロジェクトに必要な情報・ドキュメントを管理します。ドキュメントの管理方法を決定し、チームメンバーに周知徹底します。

・テストリーダー

成果物の品質を保証、維持できるテストを実行するため、テストケース基準の作成や作成されたテストケースの妥当性判断を行います。

以上について決まったら、プロジェクト管理ドキュメントを作成しましょう。

3 納品物

下記が本研修での納品物の一覧になります。

プロジェクト管理用ドキュメント

- プロジェクト体制と目標をまとめたドキュメント
- スケジュールを管理するガントチャート

要件定義

- ユースケース図
- ユースケース記述
- 用語辞書

設計

- 画面遷移図
- 画面レイアウト設計書
- テーブル設計書
- ER 図
- クラス設計書
- (クラス図: 時間次第の任意作成)

実装

- ソースコード

テスト

- 結合テスト仕様書兼成績書

【重要】

はじめての開発時、システムの全体像がイメージしにくいことから、ドキュメントは大幅に修正になることが多くあります。外部設計と内部設計は8割程度の完成度合いで開発を進めて構いません。その後、修正を加え完成させてください。

その他ドキュメント

- 成果プレゼン資料。最終日に発表を行う。そのため、事前にプレゼンテーションの資料を作ってもらい。内容は下記を含ませる。
 1. チームとしての目標
 2. 各個人の目標
 3. 作成物概要(ユースケース図や画面遷移図などを見せながら解説)
 4. 工夫した点
 5. デモ
 6. 苦労した点
 7. 今後の課題
 8. 学んだ点、よくできた、成長した点。配属に向けての意気込み

4 課題システム

システム開発演習では、3つの課題システムから一つを選択します。各課題システムを開発するために必要なドキュメントと成果物のテンプレートをまとめたものを以下の URL からダウンロードしてください。

4.1 図書管理システム

「図書管理システム」を実施するチームは、以下の URL からダウンロードしてください。

http://s.linuxacademy.ne.jp/java_practice_lib.zip

4.2 宿予約システム

「宿予約システム」を実施するチームは、以下の URL からダウンロードしてください。

http://s.linuxacademy.ne.jp/java_practice_inn.zip

4.3 教科書売買システム

「教科書売買システム」を実施するチームは、以下の URL からダウンロードしてください。

http://s.linuxacademy.ne.jp/java_practice_text.zip

5 参考

授業内では解説をしますが、参考にしてください。

5.1 システム開発工程とレビュー

レビューとは、システム開発の各工程で作成された成果物が、あらかじめ決められた基準や要件を満たした状態であるかを確認することです。レビューは、テストと並んでシステムの品質を確保するための代表的な作業です。

レビューは実施形態によって、会議レビュー、対面レビュー、自己レビュー、回覧レビューなどの種類に分けられます。また、レビューはさまざまな役割を持っており、プロジェクトにとって多くの利点があります。

レビューの形態

会議レビュー	作業に関連する担当者が集まって、会議形式でレビューを行う。開発工程作業の凍結、作業方針の決定などが行われる
対面レビュー	レビューを受ける人とレビューする人の2人でレビューを行う
自己レビュー	成果物の作者自身が見直す
回覧レビュー	レビューの対象物を作業に関連する担当者に回覧し、指摘事項を記入してもらう

その一方でレビューを怠ると、プロジェクトにさまざまな弊害を及ぼします。たとえば、設計工程で作成する設計書に大きな間違いがあり、それに気付かないまま実装工程に入った場合、大きなバグを含むプログラムになってしまいます。そのため再設計、再実装が必要になり、新たに作業を発生させ、作業進捗を遅らせてしまいます。

このようなことが起きないように、各工程で作成したドキュメントの正当性を確認するための作業としてレビューが必要です。レビューによって、技術的な間違いやドキュメントの不備、記入漏れなどを排除できます。

レビューの役割

確認	成果物の正当性を確認し、間違いがあれば修正する
改良	機能、性能、品質、効率などのシステム開発の目標について検討し、洗練する
伝達	成果物を作成者以外のプロジェクト要員に理解させる
教育	プロジェクト要員の技術を向上させる
保証	あるプロジェクト要員が作成した成果物をプロジェクトとしての成果物であると認める
進捗管理	作業の進み具合を確認し、次の作業を指示する

5.2 システム開発工程と進捗管理

システム開発スケジュールは、システム開発の各工程を1つの単位として計画します。システム開発工程ごとに開始日と終了日を決定し、この日程に従って作業を進めます。終了日までにはすべての成果物が完成しなければ、他の開発要員に多大な迷惑をかけ、作業スケジュールを遅らせてしまいます。

スケジュールの遅延を防ぐには、各作業の作業量を的確に見積もり、現在の作業が全体の何%終了していて、残りを完成させるのにあとどのくらいの時間が掛かるかを把握しておくこと、つまり進捗を管理することが重要です。

作業量を見積もる項目には、ユースケース数、クラス数、画面数、テスト項目数などが挙げられます。

工程	レベル別作業名	作業量を見積もる項目
要件定義	ユースケース設計	ユースケース数
	画面設計	画面数
設計	オブジェクト設計	クラス数
	相互作用分析	シナリオ数
	コーディング	クラス数
実装	実装	クラス数
テスト	単体テスト	テスト項目数
	結合テスト	テスト項目数またはシナリオ数

作業の進捗度は、以下の計算式で求められます。

$$\text{作業進捗度} = \text{作業完了した数量} / \text{作業全体の数量} \times 100(\%)$$

たとえば、全 20 クラスが存在し、そのうち 13 クラスのコーディングが終了していれば、コーディング作業は、 $13 / 20 \times 100 = 65\%$ 完了していることになります。

また、全 25 シナリオが存在し、そのうち 10 シナリオの結合テストが完了していれば、結合テスト作業は $10 / 25 \times 100 = 40\%$ 完了していることになります。