

株式会社ネクストソート、新入社員向け研修の問題点

前提条件

- 本資料はネクストソート社員の太田が、2018年11月から2019年1月までの研修の経験と、2019年11月の面談待機の1ヶ月間で研修生の質疑対応した際に感じたことをまとめたものである。極力客観的な視点からの記述を心がけているが、一部主観的な記述もあるため注意が必要である。
- 文中に改善案のような一文が一部記述されているが、本資料は問題点を羅列したものであるため具体的な改善案は本資料を読んだ上で、研修生たちからヒアリングしたりメンターの所感などをまとめた後に改善案を議論すべきである。
- 具体的な改善案については、本資料の作成者である太田だけでなく、2020年1月現在メインのメンターを務める弊社社員の野口氏を筆頭に有識者を複数人採用し、改善案の草案を作成するべきであるとする。

1. 研修における問題点

■ 研修の全体的な話

- 問題文が短すぎるため、プログラミング未経験者からするとどんな正解を求められているのかわかりずらく、求められている正解がわからないために調べ方もわからない状態になっている可能性がある。
また、問題文中の用語や挙動の解説がないので、**どんな仕組みになっているかわからず、課題をクリアすることによって得られるスキルや知識が身につかない危険性がある**ので、現場にでてから困るはず（実際困ることがある）。
- 調べ方については、自然と身に着けていくものと捉えるとするならば、参考として紹介するリファレンス系のサイトは最小限の数で、かつ基本的な内容のものに留めるべきである。
- 研修詳細に記載されている、課題ごとの期間があまり意味をなしていない状態になっている。そのため、ダラダラと課題を進めてしまうと**3ヶ月目になり面談が始まるまでに必要なスキルや知識が身につかないまま現場にでてしまう**。
実際の現場では締め切りや工数管理の意識を持たなければならないため、研修の時から締め切りや時間に対する意識を持つように誘導する必要がある。
タスク管理ツールを導入している現場がほとんどのため、早いうちにタスク管理の癖をつけることは必要だと考える。
- どの程度 PC に対する知識があるのか、操作やショートカットがどれくらいできるかなどの研修生個々人の能力の差については、資料でなんとか埋めるのではなく通話や口頭ベースでカバーしていく方が効率がいい。
というのも、**ショートカットキーや PC の便利機能まで資料にしていると資料作成の手間がとんでもないことになる**し、弊社の研修の意図からは逸脱しているとも思う。なおかつ、個々人の能力差は資料でカバーしきれるものではなく、本人次第な部分が大きいからだ。
- 「いつでも声かけてください」と伝えても、延々と自分で調べて声をかけようとししないような気がするの、声かけやすい状況を作り出せていないのかそれともなんとか自分で解決しようとしているのか、メンターからも正直わからない。
自分で調べて解決しようとする姿勢は素晴らしいが、単純な記述ミスやエラーから原因を探るのは最初のうちはメンターと共に行った方が効率がよい、初學者の段階では積極的に質問していく方がスキル向上に繋がるはず。
とはいえ、なんでもかんでも質問するようでは技術は身につかないので「n分間調べてもわからなかったら、**実現したいこと、試してみたこと、試してみた結果**をメンターに伝える」といった具合で質問する時のルールを伝えることで、自分で調べる力と質問する力を両方身につけることができるのではないだろうか。
- メンターとともに問題解決する際に気をつけたいのは、メンターが答えを言うってしまうのではなく、本人がミスやエラーの原因に気がつけるように誘導するという点にある。研修生によるが答えを先に求めてしまうと、どういったことが原因でエラーが起こったのか、そもそもの記述の仕方にミスがあるのか、といった**エラーを潰すためのスキル**が身につかない。そのため、メンター自身も先に答えを言うってしまうのは避けるべき行為である。

- 3ヶ月目の研修内容についてだが、弊社社員によっては経験した課題内容が大きく異なり、要件定義からテスト、発表までをやった者もいれば、実践的な課題などはやらず、コーディングや学習に力を入れた内容をこなした者もいる。個人的な見解ではあるが、太田は同僚とチームで要件から作る実践的な研修内容を経験し、現場に出てからドキュメント関係には困らなかったという体験があるため、研修中にドキュメント作成やドキュメントから実装を行うスキルを身に付けておくことはとても重要であると考えている。

とはいえ、バージョン管理についてを研修でやっておらず、不安を抱えたまま出向した経験もあるため、課題としてチーム開発に重きを置くことがどこまで重要なかは判断が難しいところである。

チーム開発の経験は非常に重要だが、その中でバージョン管理の経験までできれば一石二鳥であるとはいえ同じ3ヶ月目の研修生がいるとは限らず、必ずしもチームでの研修が可能ではない。

チーム開発を課題にしなくとも、ドキュメント作成やテストなどの現場に出てから使うスキルを少なからず触れておくことで出向する前の不安を少しでも払拭できるはず。

■ 研修課題資料について

- 研修課題資料については、現在弊社社員の野口氏が作成しているように Markdown形式での作成が最適であるとする。メンターを請け負って資料作成をする場合、メンターはプログラマーであるため Markdown記法は学習コストが低く導入しやすいためだ。また、VSCodeの拡張機能を使えばPDFの書き出しにも対応できるため、Adobe Illustratorなどの高価で学習コストの高いソフトウェアを導入する必要もない。

- 現在使用している課題資料は、**問題を並べているだけでクリアすることで得られるスキルや習得してほしい技術などの記述がない**ため、目的意識を持つのが厳しい。各セクションごとに課題の目標を明記し、それを意識しながら取り組んでもらうことである程度深い知識と技術の習得を可能にするのではないだろうか。

ただし、HTMLとは何か、PHPとは何か、といった部分の解説はリファレンス系のサイトを参照してもらう形で構わないと思う。

- 一部の課題では、**ブラウザでの挙動をイメージしづらいものがあり、画像だけでは課題で求められている解答を見つけ出すのが難しい場合がある**ので、複数の静止画像ないし GIF 画像でのサンプルが望ましい。ただし、サンプルコードは解答となってしまうため、ブラウザ上での挙動を見せるだけに留めるべきである。

MVC 関係の課題についてはブラウザ上の挙動では理解し切れない部分も多いため、こういった伝え方をすればいいのかはじっくり考える必要があると考える。（MVC 関係の理解度については PHP の理解度によって大きく左右されるため、いかに PHP 課題での習熟度を上げられるかがメンター側の課題でもある）

- 『いきなりはじめる PHP』、『気づけばプロ並み PHP』、こちらの2冊については、PHP はこういう使い方をするよ程度の理解度で問題ないと思うが、基本的には写経なのでエラーが出て原因は大体が誤字脱字だったりする。そのため、PHP を書けるようになるかといった部分では、研修の資料としては不足している気がする。とはいえ、その後の PHP 課題でしっかりカバーできれば、上述の2冊は逆引き辞典としての使い方に切り替えていくのは大いに賛成である。ただし、こればかりに時間かけていても次には進めないで研修生個々人の差はあれど、**この2冊を4～6営業日で終了させるのが理想と考える。**

2.各課題についての問題点

■ HTML 課題についての問題点（野口氏作成の課題資料）

- 課題をクリアして習得できるスキルや目標の記述がない

■ CSS 課題についての問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 実行結果のサンプルに余計な情報が多い（ブラウザのUIなど）
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない

■ JavaScript 課題の問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 正解には動的な動きがある問題に対して、実行結果のサンプルが静止画像一枚だけなので挙動のイメージがしづらい（大体 1 - 6 を聞かれる）
- 関数という言葉が突然でてくる
- [JavaScript機能説明.pdf](#)に一通りJavaScriptについて説明がされているが、コードが見つらい

■ jQuery 課題の問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 実行結果のサンプルがない
- 問題が 1 問しかない

■ Bootstrap 課題についての問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- [Bootstrapナビゲーションテキスト.pdf](#)にも[Bootstrap課題.pdf](#)にもBootstrapについての説明がされてない
- 読解力がなければ読めないわけではないが、問題文が簡潔すぎる（気がする）
- いままでの資料では登場しなかった「レスポンシブ」という単語が突然でてきて説明がない

■ PHP基礎研修基本課題の問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 実行結果のサンプルがない
- リファレンスがないために、初めて見る単語が多いはず
- 問題文の日本語が怪しいせいで、読解力があっても読みづらい
- 制御文課題で、パッと見ではif文、switch文、while文の区切りがわからない
- 教科書の理解度を無視するのであれば、全体的に問題文が簡潔すぎてわかりづらい
- 全体的に解説もなく突然いろんな単語がでてくる

■ PHP基礎研修クラス課題の問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 実行結果のサンプルがない
- クッキーとセッションの課題とファイル関連の課題が混ざっている

■ PHP基礎研修DB課題内容、PHP基礎研修DB連携課題内容の問題点

- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 実行結果のサンプルがない
- 問題文の日本語が怪しいせいで、読解力があっても読みづらい

■ Laravel資料の問題点

- 資料が古いため、バージョンの違いで挙動に差が出る（最新バージョンではBootstrapが含まれていないので自分でインストールする必要があるなど）
- 課題をクリアして習得できるスキルや目標の記述がない
- 前提条件がない
- 参考にしてほしいリファレンス系のリンクがない
- 問題文の日本語が怪しいせいで、読解力があっても読みづらい
- そもそもフレームワークとはなんなのかななどの解説がない
- 文体の敬体と常体が混在していて読みづらい
- コードが読みづらい
- コードとコマンドの表記方法が分かれてないため、どちらなのかわかりづらい
- 突然出てきたORMについての解説がない（かつては、早河さんによる口頭での解説があった）

2. 追加したい課題

■ バージョン管理関連

- 現場では、使っていないところはまずないと言っていいバージョン管理。野口氏による研修では3ヶ月目にGitの研修をやっているが、口頭だけでなく課題資料を作成すべきだ考える。
- 1、2ヶ月目でやるには少しハードルが高い気がするので、3ヶ月目でやるのがベスト？
- 最終的に現場ではGUIツールを使うはずなのだが、最初はコマンドでの作業をさせることでCUIへの精神的ハードルを少しずつ排除することも可能になる。

■ タスク管理関連

- Redmine、Backlog、InnoPM…タスク管理ツールは数多くあり、全てを網羅することは不可能であると同時に、現場によって使用しているツールが違うため一概にこれ！というのは決められないが、なにかしら触れていると現場でどんなツールを使っているかにも苦にはならないはず。
- タスク管理については、1ヶ月目からでも導入可能であると考え。早い段階で自分でタスク管理する意識とスキルを身に着けることで、だらだら研修をやることを抑制する効果があるのではないだろうか。
- 問題としては、研修の3ヶ月だけのためにアカウントを作る必要があること。研修生の中には学生もあり、平日すべて出勤することができない場合もあること（学生の場合は正式に社員として入社した際にアカウントを作成すれば大丈夫？）。
- ツールによってはドキュメントを置いておけるものもあるはずなので、課題資料のPDFをわざわざZIPファイルにして配布する必要もなくなる。
- プログラミングの課題というわけではないので、研修開始の最初の数日に口頭で使い方を伝える程度に留めて、課題の進捗を本人、メンターとともに確認するようにしていく。

■ ドキュメント関連

- 開発において、ドキュメントを参照しながらの作業があるのは当たり前ではある（ドキュメントがない会社中にもある）。そこで、研修中にドキュメントに慣れておくことは、出向後の仕事への不安をある程度払拭することができる。
現場ごとにドキュメントのテンプレートがあるので、実際にはそのテンプレートに従って作業するのが望ましいが、ドキュメントとは何かを知ってるだけでも、ドキュメント関連の作業をする際に学んだことは大いに役に立つ。
- できることなら、要件定義からドキュメント作成の経験ができるのが望ましい。

■ テスト関連

- 開発が完了したら必ず動作確認のテストを行う。これに関しても現場ごとにテスト仕様書があったりテストの粒度が違うので、研修でやるとするのであればテストにおいて重要な観点や着眼点に重きを置いて、実際にテストを実施して経験を積むことができる。
- 問題としては、なにをテストするか、である。研修で作ったものだとアプリケーションとして規模が小さすぎる。かといって、なにか他のアプリケーションがあるのかと言うとそれも微妙だったりする（太田が研修で作ったランチ検索アプリならDockerで実行可能だし、テスト仕様書などもあるから資料としては提出できる）。

3.まとめ

■ 今後について

- 上述した問題点については、太田の視点によるものであるため研修生が実際に感じているものなどとは乖離したものであるため、改善案については、最近出向した人や現在研修している人に問題点などをヒアリングをして現場の感想を集める必要がある。
- また、現場からの情報だけでなく出向している社員からも、現場に出てから困ったことなど出向前に必要な情報や技術、知識を集めておくことで、改善案の精度を高めることができる。
- 今後の予定としては、集めた情報を元に社内の有識者を集め改善案の作成を行う。草案としておおまかな研修予定を作成、社長や有識者のレビュー後、修正。修正後に各セクションの詳細な研修課題作成を行う。

■ 資料作成について

- 課題の資料作成については、Markdown形式に行うのが吉。
- 作成の指針については、改善案がまとまった段階で指針に関する資料を作成し、今後メンターとして本社にいる人間が課題資料を作成すること。
- 課題作成のテンプレートに関しては、現在野口氏が作成したHTML課題のMarkdownファイルを参照すること。

以上。