

JSTQBAへの道

結果にコミットするウエザップ勉強法



JASST実行委員 上田 和樹

このセッションで伝えたいこと

- ▶ JSTQB ALを題材に、資格試験の勉強法についての提案
 - ▶ 資格取得だけではなく、実務に活かすことも考える
 - ▶ 資格試験受験について周りを巻き込む方法
- ▶ うまくいったことや悩んだり回り道したりしたことを、そのまま伝える
 - ▶ 同じような悩みや取り組みをされている方もいると思いますので・・・

JSTQBとは？（公式WEBサイトより引用）

JSTQBとは、日本におけるソフトウェアテスト技術者資格認定の運営組織で、各国のテスト技術者認定組織が参加しているISTQB（International Software Testing Qualifications Board）の加盟組織として2005年4月に認定されています。

ISTQBの加盟組織の各国団体は資格および教育・訓練組織認証について相互認証を行っています。つまり、JSTQBが運営するソフトウェアテスト技術者資格は海外でも有効な資格となっています。

JSTQBについて
<http://jstqb.jp/committee.html>

Advanced Levelについて

- ▶ JSTQBは現在二つのレベルがある
 - ▶ Foundation Level（基礎技術者向け）
 - ▶ Advanced Level（高度技術者向け）
 - ▶ Expert Levelは、まだ試験未実施
- ▶ Advanced Levelは三種類
 - ▶ Advanced Level テストマネージャ
 - ▶ Advanced Level テクニカルテストアナリスト
 - ▶ Advanced Level テストアナリスト

過去の受験結果について

	受験者数	合格者数	合格率
ALトライアル試験 (2010/8/28)	189名	19名	10.05%
第1回AL試験 (2011/8/27)	278名	240名	83.6%
第2回AL試験 (2013/2/16)	277名	19名	6.86%
第3回AL試験 (2014/2/15)	91名	14名	15.38%
第4回AL試験 (2014/8/30)	357名	34名	9.52%
計	1192名	326名	-

異様に低い合格率

札幌の合格者0人
(実行委員も数名が
不合格・・・)



背景

- ▶ 私はとある企業のテストチームの管理職 
- ▶ JSTQB Foundation Levelは第一回試験にて合格
- ▶ 最近現場を離れて久しい
 - ▶ **アイツは本当にテストできるのかよ？の声もチラホラ**
- ▶ チームメンバが数名AdvancedLevelを受験するらしいことを耳にする
 - ▶ 当然、メンバの合格をサポートしないといけない
 - ▶ しかし負けたくはない！
 - ▶ 威厳を保たないと！！
 - ▶ まだまだ、若いモンには負けない！！！！

AL受験を決意！

ウエザップ勉強法



読む

JSTQB
シラバス

疑問点、気
づき、理解
点など

リアルタイムにPCで
書く

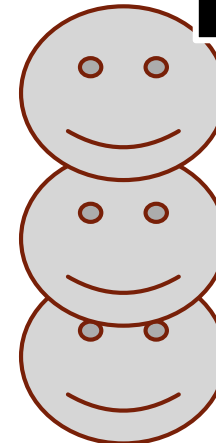


投稿

社内SNS

閲覧・書込

メンバ



みんなで合格！

結果にコミット

我慢

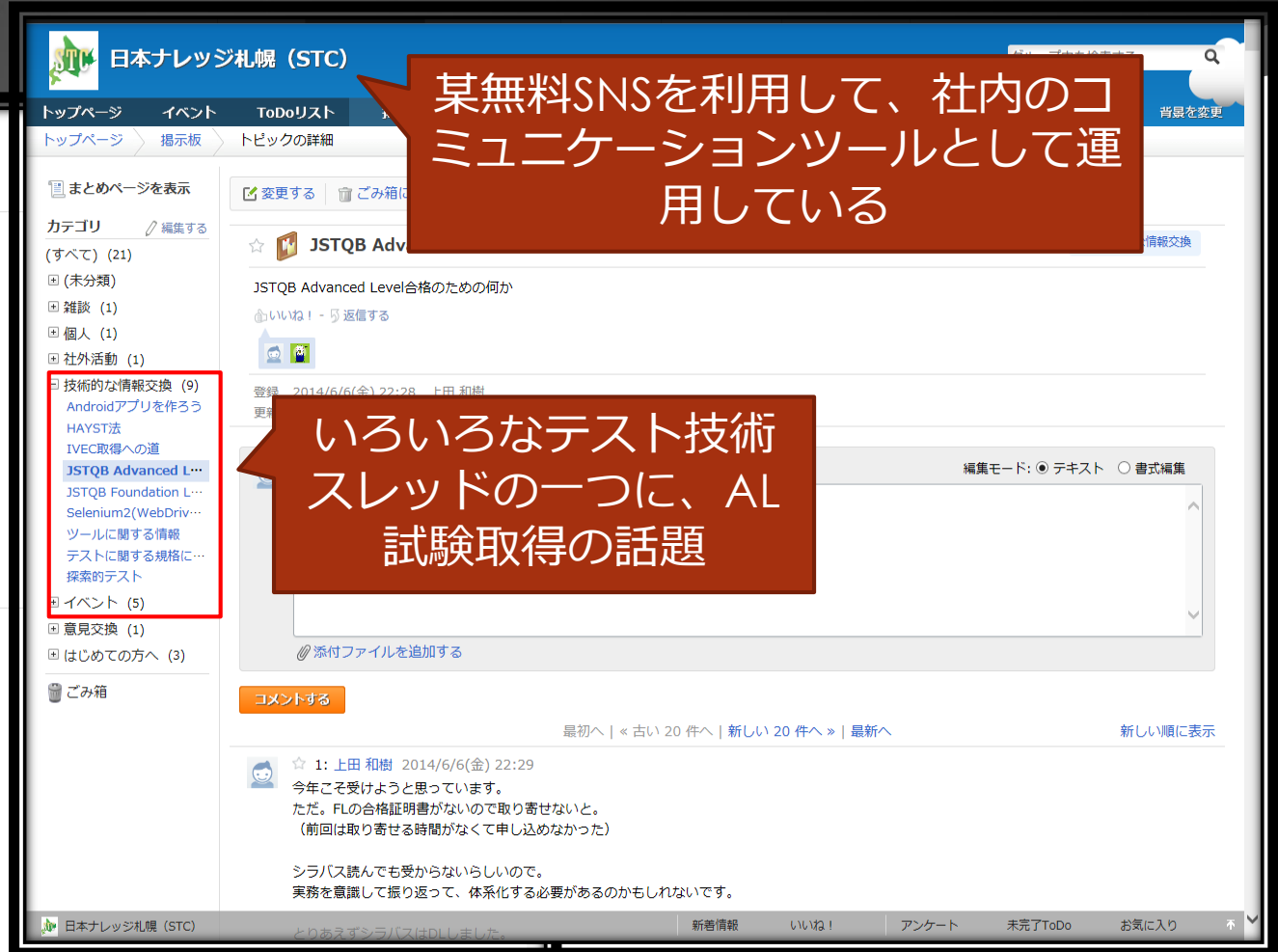


SNSにて展開した勉強記録

掲示板形式で、誰でも投稿可能。返信や「いいね」機能アリ。

某無料SNSを利用して、社内のコミュニケーションツールとして運用している

いろいろなテスト技術スレッドの一つに、AL試験取得の話題



シラバスの全体像

- ▶ 1.テストプロセス (8P)
- ▶ 2.テストマネジメント (29P)
- ▶ 3.レビュー (6P)
- ▶ 4.欠陥マネジメント (5P)
- ▶ 5.テストプロセスの改善 (5P)
- ▶ 6.テストツールおよび自動化 (6P)
- ▶ 7.スタッフのスキル – チーム構成 (7P)

2014/6/6 ～受験の決意～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと85日

試験日

まずは試験を受けることを決意

試験約 3 か月前

1: 上田 和樹 2014/6/6(金) 22:29



今年こそ受けようと思っています。

ただ、FLの合格証明書がないので取り寄せないと。
(前は取り寄せる時間がなくて申し込めなかった)

シラバス読んでも受からないらしいので。
実務を意識して振り返って、体系化する必要があるのかもしれないです。

とりあえずシラバスはDLしました。
まだ5ページくらいしか読んでないけど。

社内SNSで試験を受けることを宣言。ただ、特に今後の勉強法はノープラン。

札幌での第一回試験は、合格者が0・・・との噂は聞いていた。

FLの合格証がないと受けられない

- ▶ FLの合格証は、とっくの昔に紛失
- ▶ JSTQB事務局に問い合わせれば再発行は可能
- ▶ 前回試験は、合格証の取り寄せが間に合わなく、受験は断念
- ▶ 今回は再発行が間に合った
- ▶ ただ、日数がかかり経ってしまった・・・

2014/6/24 ～勉強法に悩む～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと67日

試験日

さて、どうしたものか

2: 上田 和樹 2014/6/24(火) 18:21



申し込み完了しました。さて、どうしたものか。。

受けるからには受かりたい。。。

噂では、シラバス読んで覚えるだけじゃ合格しないとのことですね。

「実務」と「シラバス」を照らし合わせながら読んでいかないといけないのかなと思いました。

最終的には（身もふたもないけど）実務経験かなあ。

過去の実務経験を整理する必要があるそう。

んで、当然FLの知識があってこそそのALなのでFLの再確認も。。。

良い勉強法や、対策があるかたは是非意見交流しましょう！

思いついた日から数週間経過。なかなか腰が重い。

過去の実務経験から、実務に結び付けた勉強方法にチャレンジ。

社内SNSでスレを立てて、仲間を巻き込もうとチャレンジ

読みながらSNSに書き綴ってみることにした

2: 上田 和樹 2014/6/24(火) 18:21



とりあえず、ちょっとシラバス読んでみました。

気が付いたことを読みながらSNSへ記載してみることで、

自分の理解度を深めるとともに、

他の受験者への気づきになれば幸いぐらいのノリで書いてみます。

SNSに書きながら勉強することを思いつく。

- 自分の理解を深める
- あとで自分の思考プロセスが見返せる
- 他の人の気づきになるかな？

こんな感じで、書きながら考える

2: 上田 和樹 2014/6/24(火) 18:21



1.1 イントロダクション

テストプロセスは、重複して行うケースもあるんですね。

どういう時だろうか？分析と設計とかを一緒にやるケースとかかな？

テストプロセスがFLより詳細化してますね。

ところで、「テストプロセス」「テストレベル」を混同している方は一回整理したほうがよいかもしれません。

1.2.1 テスト計画

メトリクスは、計画段階で目的と種類と基準を設定する必要がありますね。

基準と目的なきメトリクス収集は無意味です。

<以下略>

JSTQBシラバスVersion2012.J3より原文引用（以降、同様）：「プロセスの活動は理論的には順番に行われたり、重複したり同時に行われたりする場合もある」

・ 疑問点を出しながら読む

・ 考えながら読む

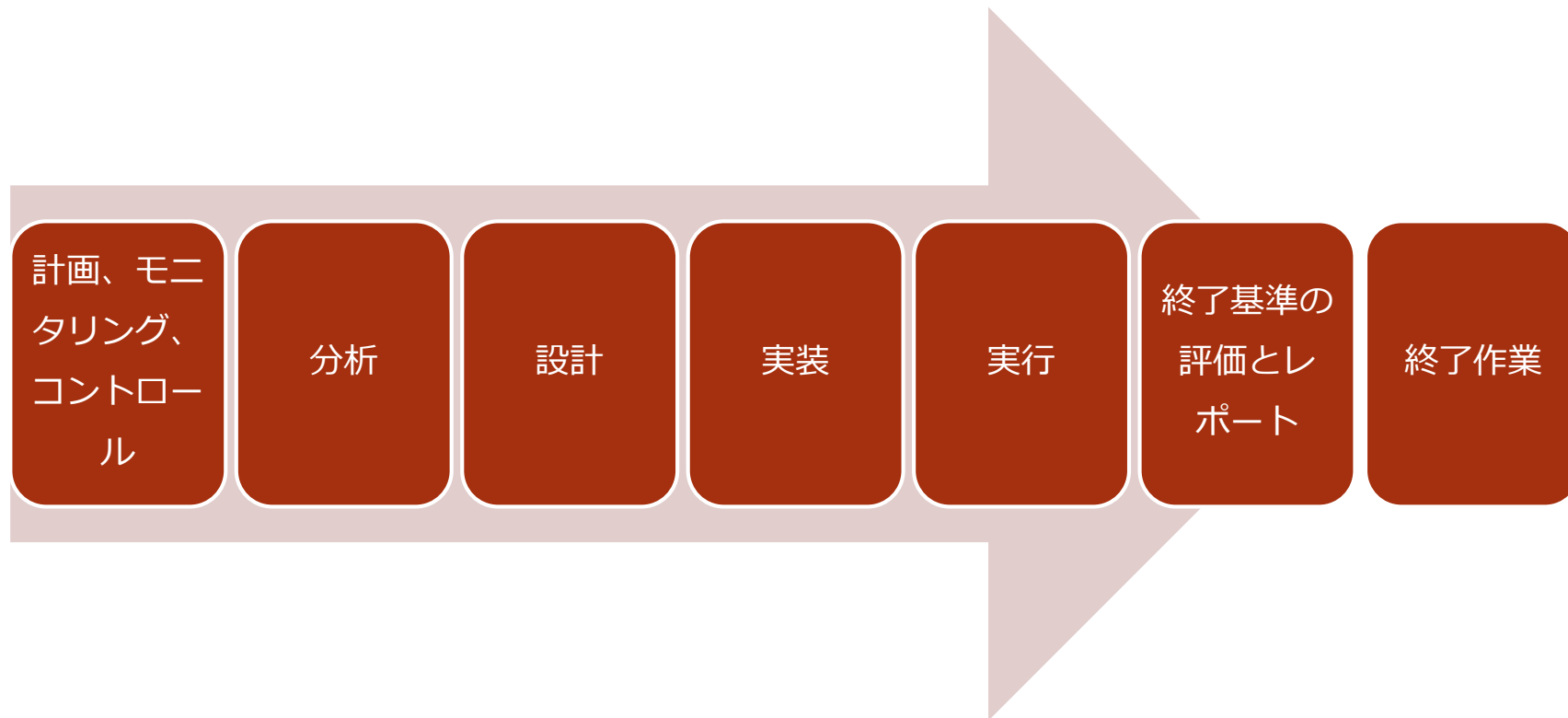
・ 実務経験を結び付ける

テストプロセス：テスト計画／テスト分析・・・など
テストレベル：コンポーネントテスト／結合テストなど
★レベルとプロセスは意味合いが違う

JSTQB Foundationのテストプロセス



JSTQB Advancedのテストプロセス



テストレベル

JSTQB用語集より引用

テストレベル：系統的にまとめ、管理していくテストの活動のグループ

コンポーネントテスト

統合テスト

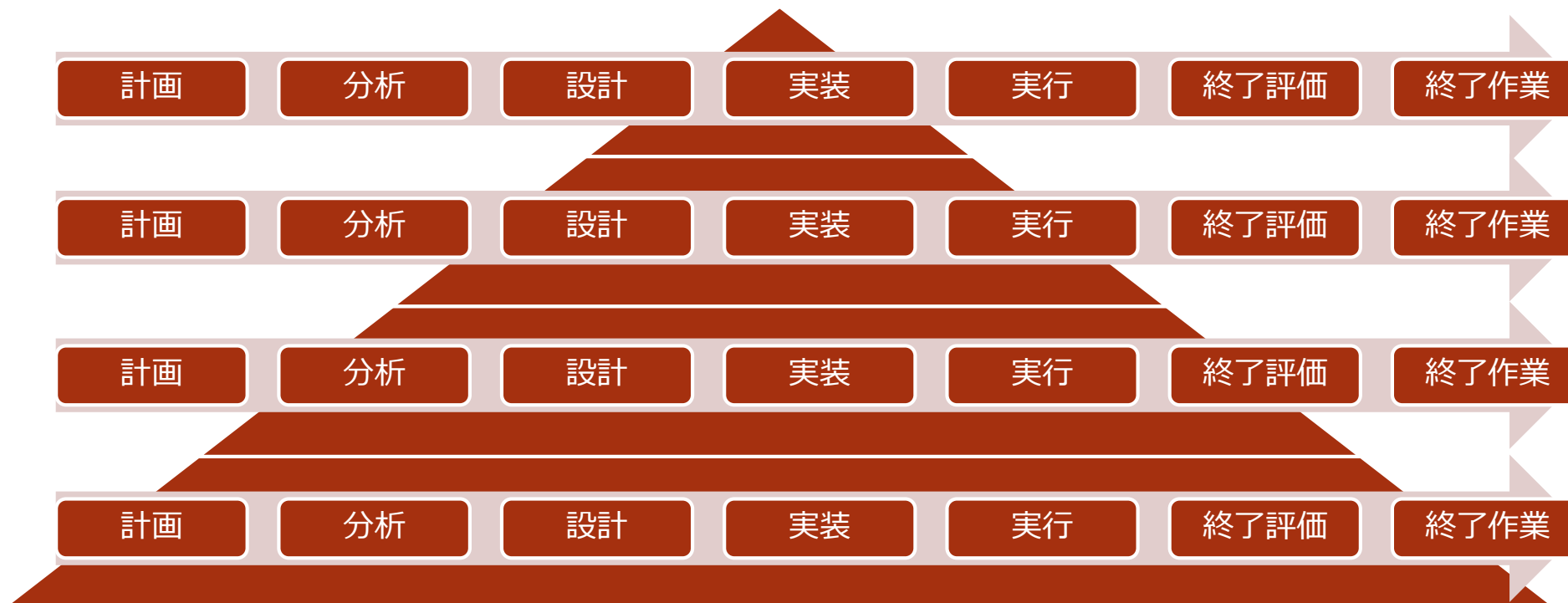
システムテスト

受け入れテスト

JSTQB用語集より引用

プロセス（process）：相互関係のある活動のセット。入力を出力に変換する。

テストプロセスとテストレベル



2014/7/2 ～納得しながら読む～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと59日

試験日

書きながら考えて納得する

5: 上田 和樹 2014/7/2(水) 22:21



1.6 テスト実行

<略>

このフェーズでのテストマネージャーの役割は

進捗モニタリング

コントロール

といった当たり前の活動。必要なものは「目的へのトレーサビリティ」。

つまり、当初目的を遂行するためのマネジメントを見失っちゃいけないってことだね。

テストケースを予定通り進めることじゃなくてテストを実施することによって果たされる目的を遂行するのがマネジメント。

ここ、実務で見失いがちですよ。

テスト実施するのが目的じゃないし、

バグを見つけることももしかしたら目的じゃないかもしれない。

<略>

原文引用：「テストを成功に導くため、コントロール活動を開始し実行する事である。そのために、テストマネージャは、テスト結果からテスト条件、テストベース、最後にはテスト目的へと遡るトレーサビリティ及びテスト目的からテスト結果へと進むトレーサビリティ、およびテスト目的からテスト結果へと進むトレーサビリティを使用できる。」

実務で行っていることを振り返ると、理解が深まる

実務での「あるある」

何を確認するために
テストしてるんだっ
け・・・？

今回のテスト目的は
「***」だ！

了解！目的を実現できる
テストを作ります！

テストケース

生産性を上げ
ないと

しかし、イン
シデントは見
つけないと

手順は正確に

シラバスを読むだけだと・・・

5: 上田 和樹 2014/7/2(水) 22:21



ふう、やっと第一章がおわった。。。。

全部で七章もあるんだね。

ただ、重いのが一章と二章なので、ここに関しては
頑張ろうかな。

自分は、こんな感じで勉強しています。

JSTQBのシラバスは、ただ読んでも

「ふーん」

って感じで頭に入らないので、できるだけ実務に照らしあわせて
考えるし、そこから派生する状況や問題などをイメージします。

- 過去の経験上、シラバスを読むだけだと頭に入らない
- 何となく知っている単語があると、理解したつもりになる
- 抽象的な内容も多いので、踏み込みが浅いまま読み進めてしまう

2014/7/3 ～自分なりにまとめる～

進捗

0%



50%

100%

日程

6/1

7/1



8/1

8/30

試験まであと58日

試験日

飲みながら育てながら学ぶ

6: 上田 和樹 2014/7/3(木) 22:11



さて、育児と晩酌の合間を縫って今日も勉強するか。

子供を風呂に入れて寝る支度をさせて、寝かしつけたあとの30分が一日の勉強時間。（そのあとの晩酌は欠かせない）

2.1 イントロダクション

テストマネージャーか。いい響きだよね。他のマネージャー（たとえば開発マネージャー）とはどこが違うんだろうか。そこがポイントなのかもしれない。

2.2 状況に応じたテストマネジメント

2.2.1 テストステークホルダについて

ステークホルダ = 利害関係者

ですね。とにかく、何かしらの関係をもつ人たちのことです。どんな人たちがいて、どんな役割をもつのかというと。。。。

シラバス上では、ステークホルダの種類と役割が列挙されていたので・・・

自分なりにまとめてみる（ざっくり）

自分の言葉でまとめる

<ステークホルダの、テストチームから欲しいものとやりたいことリスト>

=====

開発系の人たち → テスト結果を受け取り、バグをつぶして品質向上

アーキテクトな人たち → テスト結果を受けてアーキテクトを考え直すのか？

マーケットな人たち → 必要な機能や仕様などを分析して、その品質を決定。売れないといけないからね。

上級管理な人たち → こちらもテスト結果に基づいて意思決定。スポンサも含む。

プロマネな人たち → まあ、そのままですね。テストに関するすべての情報を受け取って、プロジェクトのかじ取りをします。お金を出すか出さないかの判断も。

サポートな人たち → 実は、一番影響を受けるのかも。仕様や品質によっては、サポート側での対応も必要だし、要望も上げると思います。

利用者たち → 実は一方的に影響を受けるだけだったりします。。。。

2014/7/8 ～気づきを得る～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと53日

試験日

「気づき」が大事。

7: 上田 和樹 2014/7/8(火) 22:30



一日10分の勉強法。「なんで、私がアドバンスレベルに・・・」

2.2.2 その他のソフトウェア開発ライフサイクル活動と成果物

要は、ほかのプロセス（テスト以外の）との関係ですね。

ところで、「プロセス」ってなんですかね。

確か、「入力と出力をもつ処理」「それを行うリソースすべて」だったと思います。「ヒト、モノ、場所」す

冒頭で言ってるのは、「テスト活動以外の成果物を評価する」・・・ここ意外と重要ですね。

テストは必ず【他のプロセスの成果物を評価する】活動なんですね。

あたりまえだけど、改めて認識しました。

<略>

原文引用：「ソフトウェアテストは、テスト活動以外で作成された1つ以上の成果物の品質を評価する活動である」

「気づき」があると、理解が深まるし、楽しい

開発ライフサイクルとテスト活動

要件定義

外部設計

内部設計

開発

その他

仕様書

仕様書

仕様書

ソースコード

マニュアル

レビュー結果

レビュー結果

レビュー結果

動作できるPG

改善事項

全てテスト可能

2014/7/9 ～勉強法の弱点に気が付く～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと52日

試験日

この勉強法の弱点

8: 上田 和樹 2014/7/9(水) 23:16



さて、今日もやるか・・・

<略>

スパイラルモデル

プロトタイプ開発ですね。ちょっとずつ作っていったって、利用者の意見を取り入れながら開発していきます。なので、プロトタイプごとにテストを実施する必要がありますし、そのテスト結果については、プロダクトに反映されるべきです。

上記、開発モデルを理解しないと、当然ながらテストマネジメントはできないですね。

まあ、我々の業務のほとんどは現在はウォーターフォールですが、、、

<略>

しかし、この手法の弱点は、
経験がない事は理解できない
事・・・

2014/7/17~18 ~時間は作るもの~

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと44日

試験日

空き時間も有効活用

とにかく、時間のねん出が
難しい・・・

9: 上田 和樹 2014/7/17(木) 12:51



昼食後の10分間を使った勉強法。「偏差値5からのADVANCED合格！」

2.2.4 非機能テストのマネジメント

あえて、機能テストと分けているところがポイントかなと。非機能特有のマネジメントとはいかに???

ポイント①

非機能要因での不具合は、費用が掛かる→たとえば速度が遅い、使いにくい、、、なんて問題が出ると基本設計や要件定義から見直す必要があったりするし、修正箇所も広範囲にわたる可能性が。

ポイント②

非機能って難しい →たとえばセキュリティとか負荷とか、専門家に依頼する必要があります

<略>

二人目が生まれても勉強

10: 上田 和樹 2014/7/18(金) 12:46



二人目が生まれたことで戦争状態の我が家。

家で勉強する暇無しなので、昼カレー食べた後の時間を有効利用するカレー勉強法。

2.2.6 リスクベースドテストとその他の・・・・・・

この章は重要と思います。

テストマネジメントの大部分は、リスクマネジメントと優先管理だと思うので。

→極論いうと「やらない個所を決める」ことがマネジメントなんだなと思います

「リスク」=望ましくない結果をもたらす【可能性】

ですね。まだ起こってないけど「起こったらイヤだなあ」です。所を決める」ことがマネジメントなんだなと思います

前の週に二人目が誕生。妻と二人目が退院してバタバタ。。。



2014/7/24 ～発見と再考～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと37日

試験日

メタボ判定もなんのその。初めての発見。

11: 上田 和樹 2014/7/24(木) 12:53 

中性脂肪が三倍になってました。屈辱のメタボ判定にて、ダイエット実施開始。
なお、今日の昼ごはんは豚骨ラーメン。

2.3.1.2. リスクアセスメント

アセスメント＝客観評価ですね。

今はまだ「洗い出した」だけなので、それぞれに
適正な評価を与えます。「可能性」「影響」とかそういう評価。

で、リスクに「品質特性」を用いることもあるんですね。。。
初めて知りました。

リスクのタイプをタイプに分けて分類すると。 ★詳細は確認しないとい

<略>

ここはいまだによく理解できていません・・・

原文引用：「リスクの分類では、それぞれのリスクを、性能、信頼性、機能性などの適切なタイプに分ける。一部の組織では、ISO9126 [ISO9126]（ISO 25000 [ISO25000]に改訂中）の品質特性を使用してリスクを分類している」

普段使い慣れた言葉も、改めて考えてみる。

12: 上田 和樹 2014/7/24(木) 22:42



さて、今日はビールを冷やしたジョッキで飲みます。お供はスライスしたサラミとキュウリ。

2.3.1.3 リスク軽減

リスクが開発前にはっきりしているなら、テストフェーズを待つ必要はなく、レビューでの検出や組み込み時の注意喚起にてリスクは提言できますね。

・・・何となく「リスク」って日常用語なのでわかったつもりになりがちですが、正直リスクマネジメントとは非常に実践が難しく、実践できてる人は少ないと思います。これは自分も含めて。

これが完全にできるなら、品質問題なんて起きるはずがないのです。

次から難解ゾーンに入りそう。。

やべえ、まじでALは難しい！

「リスク」は日常用語ですが、説明させると、みんな曖昧だったりしますね。

「リスク」はJSTQB ALにおいて、非常に重要なキーワードです。

リスクとは・・・（JSTQB用語集より）

- ▶ リスク（risk）
 - ▶ 将来、否定的な結果を生む要素。通常、影響度と発生可能性として表現する。
- ▶ プロジェクトリスク（project risk）
 - ▶ （テスト）プロジェクトのマネジメントとコントロールに関係するリスク。たとえば、スタッフの不足、厳しい締め切り、変更管理などがこれに当たる。
- ▶ プロダクトリスク（product risk）
 - ▶ テスト対象に直接関係するリスク。

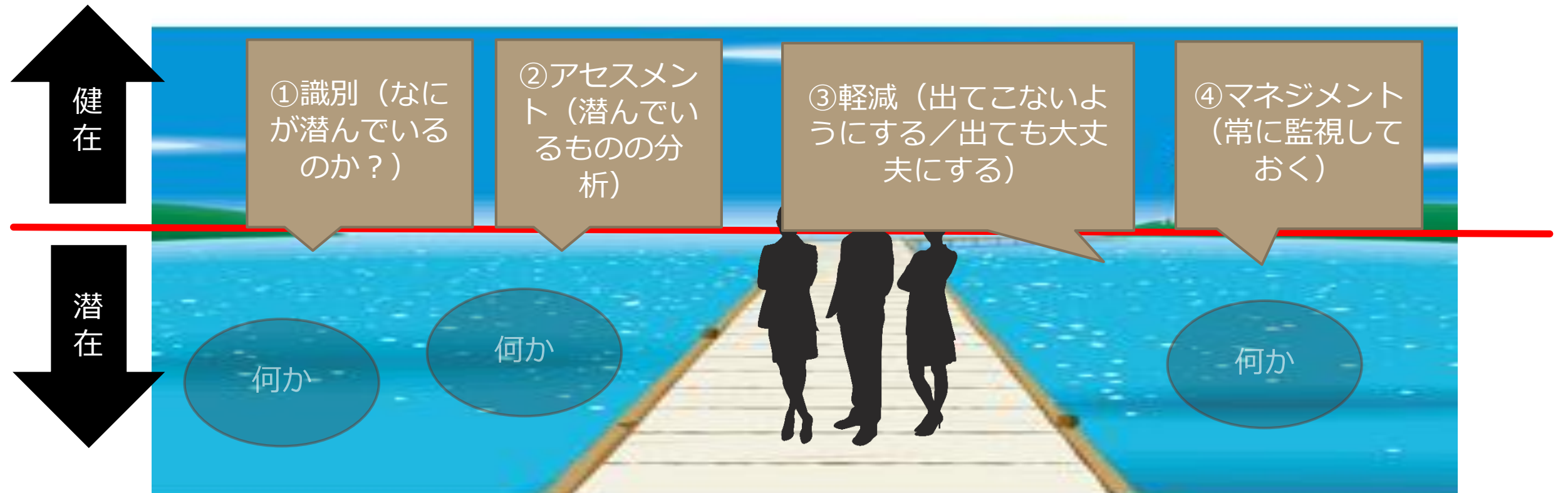
リスクとは・・・



リスクが顕在化してからでは・・・



リスクベースドテストの活動



2014/8/1 ~悩む~

進捗

0%

50%

100%

試験まであと29日

日程

6/1

7/1

8/1

8/30

試験日

ヤマにぶちあたる

15: 上田 和樹 2014/8/1(金) 23:10



2.3.2 リスクベースドテストの技法

おおおお、ここは。。。難しい！

たぶん、テストのヤマなんじゃないかと。

結局のところ、マネジメントの難しさはリスクコントロールだし、
テスト戦略ってのは「やる／やらない」「優先度」のコントロールだ。

時間が無限にあって、すべてのテストが実施できるなら

マネジメントなんていない

少しずつ、読み解いてみるか・・・

私の主観です

悩む

①リスクベースドは進化する

②クロスファンクショナル → 企業で、部門ごとに存在する知識や手法などを横断的に流通させ、組織全体の機能を強化すること。リスクの識別は、技術的側面だけでなく、あらゆる立場の人の意見を聞けてきな感じ。要は「影響度」の評価にかかわる部分ですね。

③リスク管理は上流でやれ

④リスク分析などの活動の結果は、テストの実務に生かせ

⑤残存リスクは、きちんと関係者に見える化してね

・・・くらいの感じなんだろうか。なんかぼんやりだ。あとで、じっくり読み返そう。そして。。。

SST = Systematic Software Testing = 構造的ソフトウェアテスト ???

これわからん。しかし、この技法には要求仕様が必須と書いてあるので、要求からリスクを分析する手法なんだろうか。

あとの記述を見ると、ここはどちらかという「機能仕様」に着目しているんだろうな。

つまり、「影響度」より「バグの出やすさ」をメインにするのかな ??? ★教えて！えらいひと！！！！

難しい・・・

メンバーの「いいね」で奮えたつ

こつに外機能開発に集中しつつ。

たとえば「ボタンを押した時の反応速度が、利用者にどれだけ影響を与えるか」の評価みたいな感じかな。

最後の一文は面白いですね。

優先度付けされた要求は、リスクレベルと相関がある

まあ、そりゃそうだ。

重要な要件に紐づくテストは、問題があったらヤバいのでリスクレベルは高くなる。

しかし、こうとも言える。

★要件（要求開発）の段階で、優先付けを適切に行えばプロジェクト全体のリスクが下がる

さて、今日はピールが私を呼んでいるのでここまで。
良い週末を！！！！

いいね！ - 返信する



少ないけど、日々の「いいね」があると勇気づけられる。

2014/8/11 ～2章を駆け抜ける～

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと19日

試験日

2章を走り抜けた

16: 上田 和樹 2014/8/11(月) 22:28



残り2章 一気にいくぜ。

今日は休み取って、赤子と二人っきりで育児しながらJSTQB勉強。なかなか寝ないん

<中略>

2.7

テストの価値。コスト換算してみてもいいかな。

じゃないと、いつまでもテストが「間接部門」だし「デッドコスト」になってしまって、予算

<中略>

★お知らせ★

予告通り、2章は走り抜けました。以降は、まとめ作業はしません。

なぜなら、、、もう時間がないので、、、まずは一通り読んで理解しようと思います。

まあ、こんな感じで、いろいろと具体化して実務と結び付けて学習していくのが良いのでは、、、という提案です。

自分が合格しないとまったく説得力がなくなるので、ぎりぎりまで頑張ります。

とりあえずひと段落。（時間が無くなって焦ってくる）

2014/8/18 ～3章を駆け抜ける～

進捗

0%

50%

100%

試験まであと12日

日程

6/1

7/1

8/1

8/30

試験日

以降、サッと読んでいく

17: 上田 和樹 2014/8/18(月) 12:50



夏休みが終わってしまった。。。。休んだような遊んだような、それでいて疲れているのはなぜだろう、

3章 レビュー

ざっくり。

基本的な内容はFLと同じだけど、レビューの対象が

「プロジェクト」に拡大していますね。

FLは「プロダクト」のレビューだけだったのに。

契約レビューとか、マネジメントレビューとか。

ここで、このシラバスは「テストして、ソフトのバグを報告するヒト」から

「プロジェクト全体の品質（プロジェクト品質込み）に責任を持つヒト」

に拡張されました。これはすごく重要なことだと思います。

2014/8/22 ～4章、5章、6章を駆け抜ける～

進捗

0%

50%

100%

試験まであと8日

日程

6/1

7/1

8/1

試験日

8/30

段々と雑になってゆく

18: 上田 和樹 2014/8/22(金) 22:13



というわけで、あと一週間！！！！残り、一気に行くぜ！！！！

第四章

欠陥のマネジメントです。

欠陥や故障を検出するだけじゃダメなんですね。AIは。

欠陥のワークフローは、まあよくわかる内容。

ポイントは、やはりプロセス改善の材料となる欠陥マネジメントだと思います。

とするとポイントは4. 4。

欠陥情報から、プロセス改善を図る内容です。

ここはとても大事。

<略>

どんどんザックリに。。。

18: 上田 和樹 2014/8/22(金) 22:13



第六章

投資効果ROIは重要とみた。

ツールの導入の失敗は、ここの考慮ができていないことが多いから。

自動化とかもそうだけど、自動化すればコストは下がる。。。と簡単にいうけれど、導入コストやメンテナンスコストを考えると、たいていはコストオーバーになります。

むりやり終わらせた

18: 上田 和樹 2014/8/22(金) 22:13



7章

7. 4のテストチームの分類と、メリットとデメリットを抑えればよし。

模擬試験をうけまくる

- ▶ スクエアリング・サービス
 - ▶ <http://squaring.jp/>
 - ▶ 無償と有償のコースあり
- ▶ 現在のところ、ADVANCED LEVELでの模擬試験サイトや書籍はあまり無い
 - ▶ Foundation Levelであれば、いくつかのサイトや書籍はある

2014/8/26 ~追い込み~

進捗

0%

50%

100%

日程

6/1

7/1

8/1

8/30

試験まであと4日

試験日

まとめ作業に入る

19: 上田 和樹 2014/8/26(火) 22:25 

重要ポイントまとめ ～第二章・ライフサイクルモデル～

逐次型(シーケンシャル)

- ウォーターフォールモデル
- V字モデル
- Wモデル
- PSP: Personal Software Process

繰り返し型(イテレーティブ-インクリメンタル)

□イテレーティブ(反復型)

- RAD: Rapid Application Development (※プロトタイピングの応用型)
- RUP: Rational Unified Process (※オブジェクト指向型)
- スパイラルモデル (※Plan-Do-Check-Actのデミングサイクルが多い)

<略>

ひたすらまとめ

21: 上田 和樹 2014/8/28(木) 18:26



重要ポイントまとめ ～5.3 プロセス改善モデル～

-
- テスト成熟度モデル(TMMI) ※5段階の成熟度レベル。段階表現。
 - テストプロセス改善 (TPI) ※4つの観点で20個のキー領域を設定
 - CTP ※クリティカルテストプロセス。12個のプロセス領域。
 - STEP ※体系的テスト評価プロセス。
 - テスト改善モデル(TIM) ※リスクマネジメントとコスト効果に着目
 - テスト組織成熟度(TOM) ※技術面ではなく組織面の課題に着目

やることはやった！

22: 上田 和樹 2014/8/29(金) 23:54



いよいよ明日ですね！AL,FL受験する
みなさん、健闘をいのります！

2014/8/30 (土)



出張から当日の朝に帰還。
札幌駅から会場直行。

試験終了

24: 上田 和樹 2014/8/30(土) 13:47



おわたー。一応、大体の問題は確信が得られた気がするけど、どうかなあ、、、。
あとは神頼みだな。

みなさん、どうでしたかね？

最後の計算問題、開発コストを除外するのを忘れてた、、、痛恨、、、

サンプル問題をやってみましょう！

- ▶ サンプル問題が公開されています
 - ▶ http://www.jstqb.jp/dl/JSTQB_Sample_Questions_ALT_M_V2012_1_02_J01.pdf
- ▶ 実際に解いてみましょう！
 - ▶ 既に解いたことのある方も、今一度じっくりと「なぜ正解なのか」「なぜ不正解なのか」を考えてみてください

感想

32: 上田 和樹 2014/9/1(月) 22:29



忘れないうちに、試験のこと書いておこうと思います。

内容としては、「基本問題」と「応用問題」と「計算問題」の三つに分かれていると思います。

基本問題は、いわゆる4択問題。

「正しいものを1つ選ぶ」方式がほとんどでした。複数選択はなかったと思います。薄く広く出ている印象で、模擬試験サイトにあるようなたとえばテストプロセス改善について、深く追及するような内容はなかったなと。

感想

たとえば、、、、

IEEE829がドキュメントの標準を定めていることを知っていれば、

「テストレベルの定義をIEEE829をもとに実施」などの選択肢が間違いとわかるはず。

要は、知識レベルを「一段」だけ活用するくらいの問題が出ると思います。

<中略>

感想

全体的に、問題はやはりリスクに関する問題が多かったと思います。

シラバス一読しても、よくわからないことが多いので、やはり自分の実務と照らし合わせて考えるのが一番と思いますが、それだと自分の経験していない業務はわからないことになるので、弱点は何かの方法で補う必要があると感じました。

そして2か月後



結果発表！

33: 上田 和樹 2014/10/27(月) 15:55



FL&ALの試験結果が出ています！

JSTQB認定テスト技術者資格Advanced Level第4回 試験結果

<http://juse-sqip.jp/jstqb/result.html>

さあ、どうでしたでしょうか、、、

ちなみに、上田はAL受かりました。。（一安心、、、）

これでようやく「テストマネージャー」を公言できます（笑）

一緒に受けて合格したメンバA池氏の感想

37: A池 S一 2014/10/27(月) 23:09



上田さんのおかげで A L 受かりました！
ありがとうございます(*'▽`)/

一応受かりましたので、こんな感じで勉強しましたというのをまとめておきます。

まず、このスレに記事が投稿される度に内容は

確認していました。でもその時は何となくしか理解できてなかったです。。

で、、、試験 1 週間前。。。

さすがにちゃんと勉強しないと受からないだろうと思い、模擬試験サイトでの学習を始めました。

一緒に受けて合格したメンバーの感想



ここからは時間との勝負でした。
残りは月～金の5日間。演習は全部で269問。
片っ端から解いて、わからないものは覚えるまで解く。
この状態で再度このスレを読み返してみると、
最初に読んだときに理解できなかったことも
理解できるようになっていることに気付きました。

<中略>

自分への効果

- ▶ 手を動かしながら読むことで、「読み流し」を防ぐ
 - ▶ 疑問などは、その時点で書き出して、あとで調べる
 - ▶ あとで読み返して、理解を深める
- ▶ 実務を意識することで、具体的なイメージを結び付けておく
 - ▶ 後の実業務に役立てている
- ▶ モチベーションの維持
 - ▶ 楽しみながらの勉強

メンバーへの効果

- ▶ SNSのまとめ投稿を見ることで、理解を深める助けになった
- ▶ 勉強の仕方や、模擬試験の紹介などの情報共有にも役立った
- ▶ 上司が取り組んでいる姿をみて、奮い立った



・・・ハズです

うまくいかなかった事

- ▶ SNSで仲間と一緒に取り組むこと。。。が「双方向」ではうまくいかなかった
 - ▶ 普段のコミュニケーションが大事
 - ▶ 普段から「一緒にやろうぜ！」の雰囲気にしておかないといけない
 - ▶ SNSの運用
 - ▶ メンバが書き込みづらくならないようにする必要があったか
- ▶ 走り抜けた後半部分の理解が足りなかった
 - ▶ この勉強法を実践するなら、もっと前から取り組む必要があった

まとめ

- ▶ 手を動かしながら、悩みながら、気づきながら勉強すると理解しやすい
 - ▶ 思考の過程をSNSなどに残してみるのは有効
 - ▶ 流し読みを防ぐ
- ▶ 一人より、みんなで取り組む方が楽しい
 - ▶ SNS利用することで、みんなで取り組んでいる感覚になれる
- ▶ 実務と結び付ける
 - ▶ 内容の理解がしやすい
 - ▶ 資格取得を業務に役立てることができる

しかし、一番の効果は . . .

- ▶ 自分へのプレッシャー
 - ▶ ここまで展開しておいて「不合格」だと死ぬほど恥ずかしい
- ▶ 絶対に合格しないといけない気分になる
 - ▶ すごく頑張れる

結果にコミット



引用・参照

- ▶ ISTQBテスト技術者資格制度
Advanced Level シラバス 日本語版 テストマネージャ Version2012.J03
- ▶ ソフトウェアテスト標準用語集 日本語版 Version 2.3.J02
 - ▶ <http://www.jstqb.jp/syllabus.html>

ありがとうございました

