

アジャイルブートキャンプ 1スプリントコース（10日間）

Archway Inc.
Consulting Services
<http://www.archway.co.jp/>

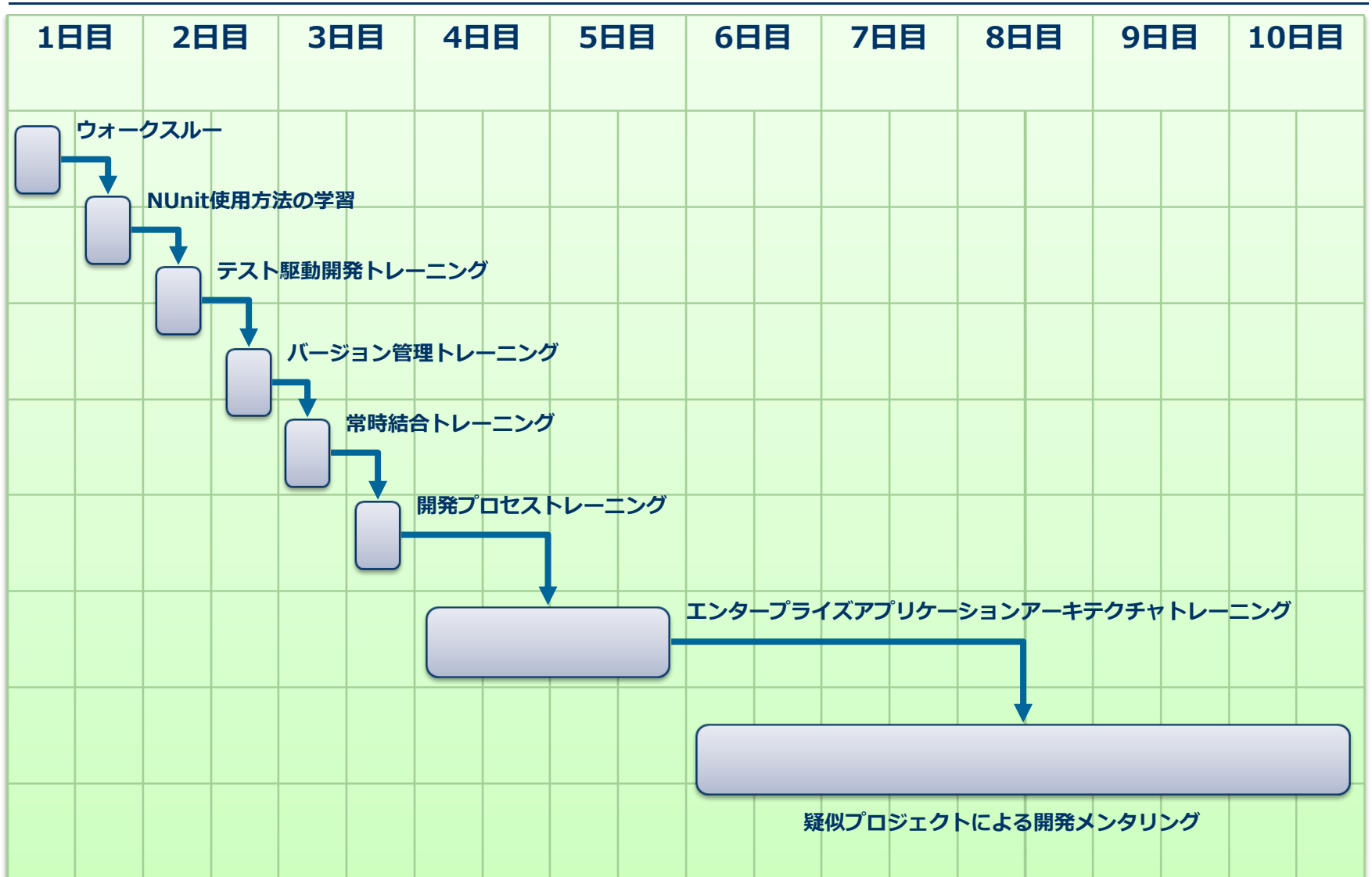
アジェンダ

- ブートキャンプの目標
- ブートキャンプのプラン
- ブートキャンプのメニュー

ブートキャンプの目標

- テスト駆動開発の演習、開発メンタリングを通じ、以下を理解する。
 - **テストフレームワークを使いこなす**
 - NUnitの使用方法を身につける
 - **テスト駆動開発に対する正しい理解を得る**
 - テスト技法ではなく設計手法
 - テスト駆動開発のリズムを身につける
 - レッド → グリーン → リファクタリング
 - **バージョン管理の必要性に対する理解を得る**
 - チーム開発の基盤となるバージョン管理の重要性を理解する
 - バージョン管理ツールの使用方法を身につける
 - **常時結合の必要性に対する理解を得る**
 - ローカル環境での自動ビルドの実施方法を身につける
 - サーバー環境での常時結合の実施方法を身につける
 - **実務で役立つエンタープライズアプリケーションアーキテクチャパターン及びそのテスト戦略を理解する**
 - データソースレイヤのテスト戦略
 - ドメイン（ビジネス）レイヤのテスト戦略
 - Webプレゼンテーションレイヤのテスト戦略
 - **疑似プロジェクト実施において、アジャイルな開発チームの習慣を体得する**
 - Scrumによるプロジェクト管理の実施
 - 予測と実測に基づいた要求管理
 - 自己組織化したチーム
 - デベロッパープラクティスの実施
 - テスト駆動開発
 - バージョン管理
 - 常時結合

ブートキャンプのプラン



ブートキャンプのメニュー

1. **ウォークスルー (0.5日間)**
 - ブートキャンプの概要説明と実施する内容およびゴールを明確にすることで、全体の学習効果を高めます。
2. **NUnit使用方法の学習 (0.5日間)**
 - NUnitのLearningTest (ツールやフレームワークの使用方法を学習するためのテストコード) を利用してのNUnitの使用方を学習して頂くことで、全体の学習効果を高めます。
3. **テスト駆動開発トレーニング (0.5日間)**
 - .NETのオープンソース・テストフレームワークであるNUnitを利用し、開発者が行うべきデベロッパーテスト手法 (テスト駆動開発) の基礎について学習します。
4. **バージョン管理トレーニング (0.5日間)**
 - オープンソースのバージョン管理ツールであるSubversionを利用し、開発者に必須のバージョン管理の基礎について学習します。
5. **常時結合トレーニング (0.5日間)**
 - オープンソースのビルドツールであるNAntを利用し、自動ビルドの基礎について学習します。
 - オープンソースの常時結合サーバーであるCCNETを利用し、常時結合の基礎について学習します。
6. **開発プロセストレーニング (0.5日間)**
 - 明確なプロジェクトマネジメントと要求管理(Requirements Management)を重視するアジャイル開発手法であるScrumの概要について学習します。
7. **エンタープライズアプリケーションアーキテクチャトレーニング (2日間)**
 1. データソースレイヤのトレーニング (0.5日間)
 - データソースレイヤのパターン (TableDataGatewayパターン) を理解し、コードのテスト容易性に対する考察とテストの記述方法を学習します。
 2. ドメイン (ビジネス) レイヤのトレーニング (0.5日間)
 - ドメイン (ビジネス) レイヤのパターン (TableModuleパターン) を理解し、コードのテスト容易性に対する考察とテストの記述方法を学習します。
 3. Webプレゼンテーションレイヤのトレーニング (1日間)
 - Webプレゼンテーションレイヤのパターン (FrontControllerパターン、ApplicationControllerパターン) を理解し、コードのテスト容易性に対する考察とテストの記述方法を学習します。
8. **疑似プロジェクトによる開発メンタリング (5日間)**
 - アジャイル開発手法であるScrumを適応して、簡単なWebアプリケーションを構築を目的とした疑似プロジェクトを実施し、チーム開発全般をメンタリングします。
 - テスト駆動開発、バージョン管理、常時結合といったデベロッパープラクティスの実践をサポートします。
 - エンタープライズアプリケーションアーキテクチャパターンを実際に適用し、Webアプリケーションに最適なアーキテクチャへの理解を深めます。

1. ウォークスルー

- 1日目 (0.5日)
 - メニュー
 - テストの重要性に関する考察 (デモ)
 - 開発者のためのテスト手法
 - テスト駆動開発の概要説明
 - 期間
 - 0.5日間 (計3時間)
 - 9:00~12:00
 - 演習
 - なし
 - ゴール
 - テスト駆動開発とは何かに対する正しい理解
 - テスト技法ではなく設計手法
 - よい設計とは何かに対する理解
 - テストしやすい設計 = よい設計

2. NUnit使用方法の学習

- 1日目 (0.5日)
 - メニュー
 - NUnitのインストール
 - NUnitの実行
 - テストの記述方法
 - 従来のシンタックス
 - 新しい制約モデルのシンタックス
 - 期間
 - 0.5日間 (計 5 時間)
 - 13 : 00 ~ 18 : 00
 - 演習
 - あり
 - ゴール
 - テスティングフレームワーク利用方法の習得

3. テスト駆動開発トレーニング

- 2日目 (0.5日)
 - メニュー
 - Stackを利用したテスト駆動開発入門
 - 期間
 - 0.5日間 (計 5 時間)
 - 09 : 00~12 : 00
 - 13 : 00~15 : 00
 - 演習
 - あり
 - ゴール
 - デベロッパーテスト手法の習得
 - テスト駆動開発のリズム習得
 - テスト駆動開発に対するより深い理解

4. バージョン管理トレーニング

- 2日目 (0.5日)
 - メニュー
 - Subversionを利用したバージョン管理入門
 - 期間
 - 0.5日間 (計 3 時間)
 - 15 : 00 ~ 18 : 00
 - 演習
 - あり
 - ゴール
 - バージョン管理ツールの使用方法の習得
 - 適切なリポジトリ設計に対する理解
 - バージョン管理ツールがチーム開発に必須であることに対する理解

5. 常時結合トレーニング

- 3日目 (0.5日)
 - メニュー
 - NAntを利用したローカル環境での自動ビルド
 - CCNETを利用したサーバー環境での常時結合
 - 期間
 - 0.5日間 (計 5 時間)
 - 09 : 00 ~ 12 : 00
 - 13 : 00 ~ 15 : 00
 - 演習
 - あり
 - ゴール
 - 自動ビルドツールの使用方法の習得
 - 常時結合サーバーの使用方法の習得
 - 動作するアプリケーションを常に得ることの重要性に対する理解

6. 開発プロセストレーニング

- 3日目 (0.5日)
 - メニュー
 - アジャイル開発手法であるScrumの概要
 - Scrumのルール
 - Scrumのセレモニー
 - Scrumのアーティファクト
 - 期間
 - 0.5日間 (計 3 時間)
 - 15 : 00 ~ 18 : 00
 - 演習
 - なし
 - ゴール
 - Scrumの正しい理解
 - 要求管理の重要性に対する理解
 - 自己組織化されたチームの重要性に対する理解

7-1. エンタープライズアプリケーション アーキテクチャトレーニング

- 4日目 (0.5日)
 - メニュー
 - データソースレイヤのパターンの理解
 - データソースレイヤのテスト容易性の考慮
 - データソースレイヤのデベロッパーテスト記述方法
 - 期間
 - 1日間 (計4時間)
 - 09:00~12:00
 - 13:00~14:00
 - 演習
 - あり
 - ゴール
 - データソースレイヤのエンタープライズアプリケーションアーキテクチャパターン (TableDataGatewayパターン) に対する理解
 - データアクセスロジックのテストに関する問題点の把握
 - データアクセスロジックのテスト戦略の習得

7-2. エンタープライズアプリケーション アーキテクチャトレーニング

- 4日目 (0.5日)
 - メニュー
 - ドメイン (ビジネス) レイヤのパターンの理解
 - ドメイン (ビジネス) レイヤのテスト容易性の考慮
 - ドメイン (ビジネス) レイヤのデベロッパーテスト記述方法
 - 期間
 - 1日間 (計4時間)
 - 14:00~18:00
 - 演習
 - あり
 - ゴール
 - ドメイン (ビジネス) レイヤのエンタープライズアプリケーションアーキテクチャパターン (TableModuleパターン) に対する理解
 - データアクセスと分離されたドメイン (ビジネス) ロジックのテストを行う必要性に対する理解
 - データアクセスと分離されたドメイン (ビジネス) ロジックのテスト戦略の習得

7 - 3. エンタープライズアプリケーション アーキテクチャトレーニング

- 5日目 (1日)
 - メニュー
 - Webプレゼンテーションレイヤのパターンの理解
 - Webプレゼンテーションレイヤのテスト容易性の考慮
 - Webプレゼンテーションレイヤのデベロッパーテスト・カスタマーテスト記述方法
 - 期間
 - 1日間 (計8時間)
 - 09:00~12:00
 - 13:00~18:00
 - 演習
 - あり
 - ゴール
 - Webプレゼンテーションレイヤのエンタープライズアプリケーションアーキテクチャパターン (FrontControllerパターン、ApplicationControllerパターン) に対する理解
 - ASP.NETランタイムに依存するクラスのテストに関する問題点の把握
 - ASP.NETランタイムに依存するクラスのテスト戦略の習得

8. 疑似プロジェクトによる開発メンタリング

- 5日目～10日目（5日）
 - メニュー
 - スプリント計画ミーティング（0.5日）
 - 1週間のスプリントで実装する要求の理解とタスクの洗い出し
 - 環境構築（0.5日）
 - ビルドサーバー、開発クライアントの環境構築
 - スプリント（3日）
 - スプリントレビューミーティング（1日）
 - 完成したプロダクトのレビュー
 - スプリントレトロスペクティブ（ふりかえり）
 - 期間
 - 1日間（計8時間）× 5
 - 09:00～12:00
 - 13:00～18:00
 - 演習
 - あり
 - ゴール
 - アジャイル開発手法であるScrumの実践による開発プロセスの体得
 - デベロッパープラクティス（テスト駆動開発・バージョン管理・常時結合）の重要性の理解
 - エンタープライズアプリケーションアーキテクチャパターンを実際に適用することによる、より深い理解

Archway Consulting Service