

# 授業支援型 e ラーニングシステム OpenCEAS の開発

— Ruby on Rails フレームワークに基づく再構築 —

宮崎誠<sup>†1</sup> 冬木正彦<sup>†1</sup> 三矢晴彦<sup>†2</sup> 栗原星史<sup>†2</sup> 奥田高広<sup>†2</sup> 植木泰博<sup>†3</sup>

**概要:** CEAS3 の UI をマルチデバイス・モダンブラウザに対応した CEAS10 の開発に続き、CEAS10 のバックエンド部を Ruby on Rails の開発フレームワークを使い、Ruby で書き直すことで実装を最新にした OpenCEAS を開発している。MVC モデルに基づいた設計を充分把握することで View の改修(CEAS10 の開発)、Ruby on Rails によるソースコードの書き換え(OpenCEAS の開発)の 2 段階に分けたモダナイゼーションを実行することができており、1 度に開発する場合と比較して失敗するリスクが少ないと考えるシステム開発手法をとることができた。また、Ruby on Rails の DRY 原則、CoC、REST 原則といった基本理念に沿って開発することでソースコードが最適化され、コード量を大幅に削減できた。また、OpenCEAS のソースコードは、将来オープンソースにすることでシステムが広く普及することを目指しており、公開にあたっては、自由度の高い MIT ライセンスを採用する予定である。

**キーワード:** 授業管理システム, 学習管理システム, モダナイゼーション, オープンソース, ソフトウェアライセンス

## OpenCEAS: An e-Learning System with Teaching-Support User Interface

- Re-written with Ruby on Rails Framework -

MAKOTO MIYAZAKI<sup>†1</sup> MASAHIKO FUYUKI<sup>†1</sup> HARUHIKO MITSUYA<sup>†2</sup>  
SEIJI KURIHARA<sup>†2</sup> TAKAHIRO OKUDA<sup>†2</sup> YASUHIRO UEKI<sup>†2</sup>

**Abstract:** Following the development of CEAS10 that supports the multi-device modern browser, we have developed OpenCEAS, which implemented the latest implementation by rewriting the backend part of CEAS10 with Ruby on Rails development framework and rewriting it in Ruby. By fully grasping the design based on the MVC model, it is possible to execute modernization divided into two stages of view renovation (development of CEAS 10) and source code rewriting by Ruby on Rails (development of OpenCEAS). We were able to adopt a system development method with less risk of failure compared to development at once. In addition, the basic idea of Ruby on Rails' DRY principle, CoC, REST principle has resulted in optimization of the source code, and the code amount can be greatly reduced. Also, by making OpenCEAS open source, we aim to spread the system widely. We plan to adopt MIT license in the future.

**Keywords:** CMS, LMS, Modernization, Open Source, Software License

### 1. はじめに

学生自身の所有端末を大学で使用する BYOD(Bring Your Own Device), スマートフォン, タブレット端末やモダンブラウザ[a]の普及, クラウドによるサービス提供など, 様々な要因によって大学における情報環境および教育体制が変わってきている. このような変化に対し, 大学の ICT 環境を整備し対応することは, 大学教育のサービスや品質にも直結するため, 非常に重要である.

畿央大学では, すべての授業科目で多様なアクティブ・ラーニングの実施を可能とするため, COPE (Corporate-Owned, Personally Enabled: 企業所有端末の個人利用) 方式

を採用し, 全学生が PC を所持することを実現している[1]. また, 2015 年度には, 大学キャンパス内に稼働していた約 40 台のオンプレミス・サーバのうち共有ファイルサーバおよび予算管理システム等の一部サーバを残し, 主要なほぼ全てのサーバをクラウド基盤[b]に移行した[2]. そして, 本学で運用している授業支援型 e ラーニングシステム CEAS[3]についても, CEAS3 系のプレゼンテーション層である UI を全面改修し, CEAS の特長である「授業支援型ユーザインターフェイス」[4]を継承しつつ, マルチデバイス・モダンブラウザ対応を行い[5], 近年の多様な利用デバイスやモダンブラウザに対応した CEAS10[c]を開発した. しかしながら, これまで長年に渡って機能強化や拡張がされて

†1 畿央大学 教育学習基盤センター  
Center for Teaching, Learning and Technology, Kio University

†2 ボウ・ネットシステムズ株式会社  
Bow Netsystems Corporation

†3 ニュータイプシステムズ株式会社  
NewType Systems Inc.

a W3C などが定義している Web 標準に充分準拠したブラウザ.

b 本学では, クラウドサービスにマイクロソフト社の Azure を利用している.

c CEAS3 系の後継のバージョンは, CEAS10 とした.

きたため、利用されている Java フレームワークやライブラリのバージョンが古く、セキュリティ面や最新技術による開発を考慮するとバックエンド部についても改修が必要な時期となっていた。

本論文では、CEAS10 の UI の Web デザインはそのままにバックエンド部の実装を最新にした OpenCEAS の開発について述べる。

## 2. OpenCEAS の開発

### 2.1 開発方針

CEAS10 におけるプレゼンテーション層の改修および OpenCEAS におけるバックエンド部の開発は、システム開発手法におけるモダナイゼーションである[6]。OpenCEAS の開発では、現在本学で利用中の業務要件、機能要件をそのままに CEAS の古い Java フレームワークの実装を Ruby on Rails を使い Ruby で書き直すことでシステム刷新、実装の最新化を行う[7][8]。システムリプレースや機能改修等とは違い、CEAS の授業支援型ユーザインターフェイスはそのままとすることによって、既存の CEAS10 ユーザにとっても新たな操作等になれる必要はなく、違和感なく OpenCEAS に移行できることを目指す。また、モダンなフレームワークを使って書き直すことによって、実運用で磨かれた実践的な機能を維持しながら、発展的ニーズへの機能拡張可能な基盤を再構築することが将来可能となる。

OpenCEAS の開発では、現在本学で利用中の CEAS10 と同じ授業支援型ユーザインターフェイスや機能を再現することを目標に Ruby on Rails の基本理念に沿いながら明確に次の方針で進めている。

#### 開発の方針

- CEAS10 と同じ UI 実装
- CEAS10 と同じ機能実装
- 使わなくなった機能廃止の検討
- 使わなくなった DB のテーブル削除の検討
- Ruby on Rails の DRY[d]原則に基づいた冗長なソースコード等の最適化
- CoC[e]「設定より規約」に準じた Active レコード、DB テーブル命名規約等の変更
- REST[f]原則によってすべてのリソースに一意的識別子 (URI) に対して、動作を決めてアクセス可能とする
- 利用ライブラリ、モジュール等のライセンス管理

#### 開発の除外事項

- 新規機能追加

d 「Don't Repeat Yourself」の略語。同じことを繰り返さない、コード(情報)を重複させないという Ruby on Rails における基本理念に基づいた開発原則のひとつ。

e 「Convention over Configuration」の略語。従来のフレームワークのように設定を大量に記述するのではなく、規約にのっとってプログラミングす

- UI および機能改修
- データベーススキーマの変更や最適化

なお、今回のモダナイゼーションにあたって Java 言語のまま最新の実装にアップデートし、最新技術に対応することも検討したが、最新のフレームワークやライブラリ等へのバージョンアップとなるため既存のソースコードに対しても相当数の修正が予想され、リライト(Ruby on Rails による Ruby への書き換え)しても開発工数はほぼ変わらないという見通しをえたため、リライトを選択した。

### 2.2 アーキテクチャ

今回の CEAS10 のリライトによる OpenCEAS の開発は、CEAS3 の UI をマルチデバイス、モダンブラウザに対応した CEAS10 の開発に続く改修である。CEAS3 は、基本アーキテクチャに Layer パターンを採用しており、Web アプリケーション全体を 3 層の機能層、プレゼンテーション層、ビジネスロジック層、インテグレーション層に分けて構成した MVC モデルである。一方、OpenCEAS の開発フレームワークとして採用した Ruby on Rails も MVC モデルで構成しており、これにより第 1 段階の CEAS10 の開発では、まず優先度の高い View の改修を実施し、第 2 段階の OpenCEAS の開発では、バックエンド部の改修の方向性を検討した後、CEAS10 の View の Web デザインはそのまま利用し、Model, Controller の書き換えを実施するという段階に分けた開発アプローチが可能となった。

モダナイゼーションにおいて、開発を 2 段階に分けることには、次のメリットとデメリットがあると考えられる。

#### メリット

- 着実に開発を進めることができるため 1 度に開発する場合と比較して失敗するリスクが少ない
- 開発を分けることで予算計画が立てやすい
- 優先度の高い項目(本開発では UI)からリリースできる
- 段階に応じて開発チームのメンバーと役割の割り当てを最適化できる

#### デメリット

- 1 度に開発する場合と比較して開発コストが高くなる

CEAS のシステム規模を次節で示すが、開発対象となるシステム規模が大きいほど複雑性が高くなるため、上記メリットの恩恵を享受することができるものと思われる。

ることで、余計なプログラミングや設定を省く。

f 「Representational State Transfer」の略語。Web アプリケーションなどにおける設計概念の一つ。

表 1 実装機能一覧

共通	ログイン・ログアウト, 公開 FAQ, お知らせ, 公開科目一覧, 個人情報変更, レポート確認
履修環境管理者	ユーザ管理, 科目管理, ユーザリスト読込, 科目リスト読込, 科目担任リスト読込, 科目担任リスト読込, 履修情報リスト読込, システム利用状況, 告知機能, アクセスログ
担当者	授業資料, 教材割付, 科目独自ページ, 複合式テスト作成, 記号入力式テスト作成, レポート課題作成, アンケート作成, 複合式テスト管理, 記号入力式テスト管理, レポート管理, アンケート管理, 出席管理, システム情報, 科目環境設定, 連結評価一覧表, お知らせ/メール, FAQ, 回答登録, 教材一括更新
学生	複合式テスト実施, 記号入力式テスト実施, レポート提出, アンケート実施, テスト結果表示, お知らせ, FAQ, 出席確認, 出席情報収集, 出席表, グループフォルダ

表 2 CEAS3 の規模

	総行数				
	Interface	class	JSP	Java	JSP
プレゼンテーション	-	143	202	39,172	32,678
ビジネスロジック	29	67	-	39,995	-
インテグレーション	34	68	-	15,048	-
合計	63	278	202	94,215	32,678

表 3 OpenCEAS の規模

項目名	クラス	メソッド	総行数	
			コメント含む	コメント除く
View	-	-	29,910	-
Controllers	57	506	12,143	9,457
Helpers	0	70	939	825
Models	42	188	2,753	2,246
Mailers	5	5	44	41
Javascripts	0	112	1,336	896
Libraries	1	1	6	6
Total	105	882	47,131	13,471

## 2.3 CEAS3 との機能比較と変更点

CEAS3 および OpenCEAS が実装している機能を表 1 に示すこれらの機能は商用の CMS/LMS(授業管理システム/学習管理システム)が提供している機能と同様であるが, CEAS ではこれらの機能が「授業支援型ユーザインターフェイス」[4]の原則にしたがって提供されている。また, CEAS3 および OpenCEAS のシステム規模をそれぞれ表 2, 表 3 に階層別に示す。表 2 の interface, class 欄はそれぞれ Java のユーザインターフェイスとクラスの数を示し, JSP と表記された欄はプレゼンテーション層にあたる View の JSP ファイルの総行数を示している。表 3 の OpenCEAS の View の行数と比較すると約 8.5%の減少率であった。一方, CEAS3 の Java ファイルのソースコード行数と OpenCEAS の Ruby によるソースコードの View を除いた総行数(コメント含む)とを比較すると約 81.7%の削減率を達成することができた。一部機能の廃止・未実装もあること, Java と Ruby で単純に比較はできないこともあるものの View に関しては表示ページ毎に対応したファイルが必要であるためコード削減効果は高くはなかったが, その他の Controller, Model 等のソースコードの削減率は非常に高く, Ruby on Rails でリライトした効果は充分にあったと言える。なお, OpenCEAS の開発過程でほとんど利用しなくなったとして廃止した機能, 現在未実装だがリリースまでに実装予定の機能は次の通りである。

### 廃止した機能

- 複合式テストの相互採点 (他人の答案を採点) 機能
- 複合式テストの XML ファイルによる問題出力機能
- 複合式テストの XML ファイルによる問題作成機能
- アンケートの XML ファイルによる質問作成機能
- アンケートの XML ファイルによる質問出力機能

### リリースまでに実装予定の機能

- CAS 認証によるログイン

DB に関しては, CEAS10 では CEAS3 および CEAS/Sakai の DB をそのまま利用できたが, OpenCEAS では, Ruby on Rails の規約に基づいたテーブル名の変更などがあるためマイグレーションする必要がある。

## 2.4 ユーザテスト

現在, OpenCEAS ベータ版が完成し, 実際の授業で試用を開始し, ユーザテストを実施している。今期の授業を通して発見されたバグや未実装の機能などについて今後対応することで, 安定版を公開できる予定である。

## 3. OpenCEAS のオープンソース化

### 3.1 オープンソース化の目的

当初より CEAS は, 日本の授業に特化した開発を行っており, 「授業と学習(予習・復習)のサイクル形成」に必要な支援機能を備えた授業支援型の CMS(Course Management System)として担当者と学生の授業や予習・復習に関する諸活動について, 毎回の授業実施を単位として扱える特長を有している[9]。例えば, 授業科目は最初から 15 回に分けて作成され, ユーザインターフェイスにおいても授業前, 授業中, 授業後および常時の利用される段階に応じて機能がまとまっているなど学生, 担当者および履修管理者の視点に立った UI 仕様に関しての工夫をしている。一方で, CEAS を導入する大学によっては, 独自の機能追加や削除,



図 1 OpenCEAS 担任者 Top ページ

画面に表示するラベルやメッセージの変更等といった要望があがっていた。これまでの CEAS は、オープンソースとして公開していたものの、利用に関するライセンスが明確でなかったためにカスタマイズしての利用実績はほとんどなかった。そこで OpenCEAS では、将来、完成したシステムのソースコードをオープンソースにすることでシステムが広く普及することを目指したいと考えている。具体的には、ソースコードをフリーで公開し、自由度の高いライセンスを採用することで、利用する大学は、各大学に必要な機能の追加やデザイン変更などのカスタマイズができるようにする。また、特定の事業者依存せずに開発が可能となることで、日本の大学におけるカスタマイズ等のニーズを満たすことが可能となる。

### 3.2 オープンソースライセンスの検討

前節でも述べたようにシステムが普及することを最大の目的としてオープンソース化する予定である。表 4 にオープンソースの代表的な LMS とライセンスの一覧を示す。Open Source Initiative (OSI)によるオープンソースライセンスの定義では、誰でも自由に入手でき、自由にカスタマイズできること、特定の個人やグループに属さないことなどに配慮したライセンスを採用することが挙げられており、できるだけ自由度の高いライセンスを採用することが重要だと考えた。また、大学の学部学科や全学などの規模で導入する場合には、システム導入や運用保守を業務委託できるサポートベンダーの協力が不可欠である。そのため、OpenCEAS をビジネスとして取扱うことの容易さも重要なポイントだと考えた。これらの点を踏まえ、OpenCEAS のライセンスには、代表的な LMS のライセンスに比べより自由度の高い MIT ライセンスを採用することとした[g]。

表 4 LMS と採用ライセンス

名称	ライセンス	ビジネスでの取扱い容易さ
Moodle	GPL v3	△
Sakai	ECL-2.0	○
Canvas LMS	AGPL v3	○

## 4. まとめ

本論文では、CEAS10 の UI はそのままにバックエンド部の実装を最新にした OpenCEAS の開発について述べた。OpenCEAS はベータ版として現在、ユーザテストを兼ねて一部の授業で実際に試用している。CEAS3 の UI を改修してマルチデバイス、モダンブラウザに対応した CEAS10 の開発に続き、今回 CEAS の古い Java のフレームワークを Ruby on Rails でリライトすることで、2 段階に分けたシステム開発手法におけるモダナイゼーションが完了できる見込みとなってきた。1 度に UI とバックエンドを改修するよりも 2 段階に分けたことで着実に開発を進めることができるため失敗するリスクを少なくすることができたと考えている。さらに、開発を 2 段階に分けることで予算計画を立てやすいというメリットもあった。現在、世の中の情報技術の進展に伴い、大学における教育体制も変わってきており、長年使い続けているシステムの設計や機能が古くなることは、いつの時代も大学のみならず起こり得る。今回のような MVC モデルに基づいた設計を充分把握したモダナイゼーションは、システム更新の選択肢の一つとして検討する価値がある。

OpenCEAS の開発では、Ruby on Rails を使って Ruby で書き直すことによって、現在本学で利用中の CEAS10 と同じ授業支援型ユーザインターフェイスや機能を再現することを明確な目標とし、開発を進めてきた。OpenCEAS のリリース後には、多様なユーザ認証機能や WebAPI, LTI 対応などによる他のシステムとの連携機能の充実を予定している。

**謝辞** 最後に、今回の OpenCEAS の開発プロジェクトにおいて、ユーザインターフェイスデザインと実装およびテストを担当いただいたボウ・ネットシステムズ (株) の石川有紀氏の多大な貢献に謝意を表します。

g MIT ライセンスを採用し続けていくには、Ruby on Rails に導入する追加モジュール等に GPL 系ライセンスのものが入らないようにすることにも気をつける必要があり、OpenCEAS には、オープンソースのツールを使っ

たライセンスチェック機能を内蔵し、MIT ライセンスによる公開の際に問題とならないよう開発に役立てている。

## 参考文献

- [1] 福森貢, 宮崎誠, 冬木正彦, 大山章博, 関大治郎, 植木泰博, 「畿央大学におけるアクティブ・ラーニング環境の整備-授業支援環境下での全学演習科目と連動する COPE 方式実践の仕組み-」, 情報処理学会, 第 21 回 CLE 研究会, Vol.2017-CLE-21 No.5, 2017.
- [2] 大山章博, 福森貢, 冬木正彦, 宮崎誠, 関大治郎, 植木泰博, 2017 年度完成年度を迎える学内主要サービスのクラウド移行と全学生へのパソコン貸与の取り組みについて, 大学 ICT 推進協議会 2016 年度年次大会, 2016.
- [3] 冬木正彦, 辻昌之, 植木泰博, 荒川雅裕, 北村裕, Web 型自発学習促進クラス授業支援システム CEAS の開発, 教育システム情報学会論文誌, 21(4), 343-354, 2004.
- [4] 植木泰博, 冬木正彦, コース管理システム CEAS の授業支援型ユーザインターフェイス, 教育システム情報学会誌, 27(1), 5-13, 2010.
- [5] 宮崎誠, 冬木正彦, 植木泰博, 日本の教育環境への適合を目指す授業支援型 e ラーニングシステム CEAS の発展 — プレゼンテーション層変更によるモダンブラウザ・マルチデバイス対応と次世代 CEAS —, 情報処理学会第 20 回 CLE 研究会, Vol.2015-CLE-17 No.32, 2016.
- [6] 岡 大勝, モダナイゼーションの九つの手法 動機によって対策は変わる 手法の違いを理解しよう (特集 どうする?塩漬けシステム : 鍵はモダナイゼーションにあり), 日経 systems (260), 43-47, 2014-12.
- [7] Ruby on Rails,  
<http://rubyonrails.org/>
- [8] Ruby on Rails – Wikipedia  
[https://ja.wikipedia.org/wiki/Ruby\\_on\\_Rails](https://ja.wikipedia.org/wiki/Ruby_on_Rails)
- [9] 児山享弘, 小山和倫, 植木泰博, 荒川雅裕, 冬木正彦, 日本の教育環境への適合を目指す授業支援型 e ラーニングシステム CEAS ~Java フレームワークに基づく再構築~, 情報処理学会 研究報告教育学習支援情報システム研究グループ 第 1 回 CMS 研究会, 43-47, 2005.