Learning Management System

Ryan Palmer

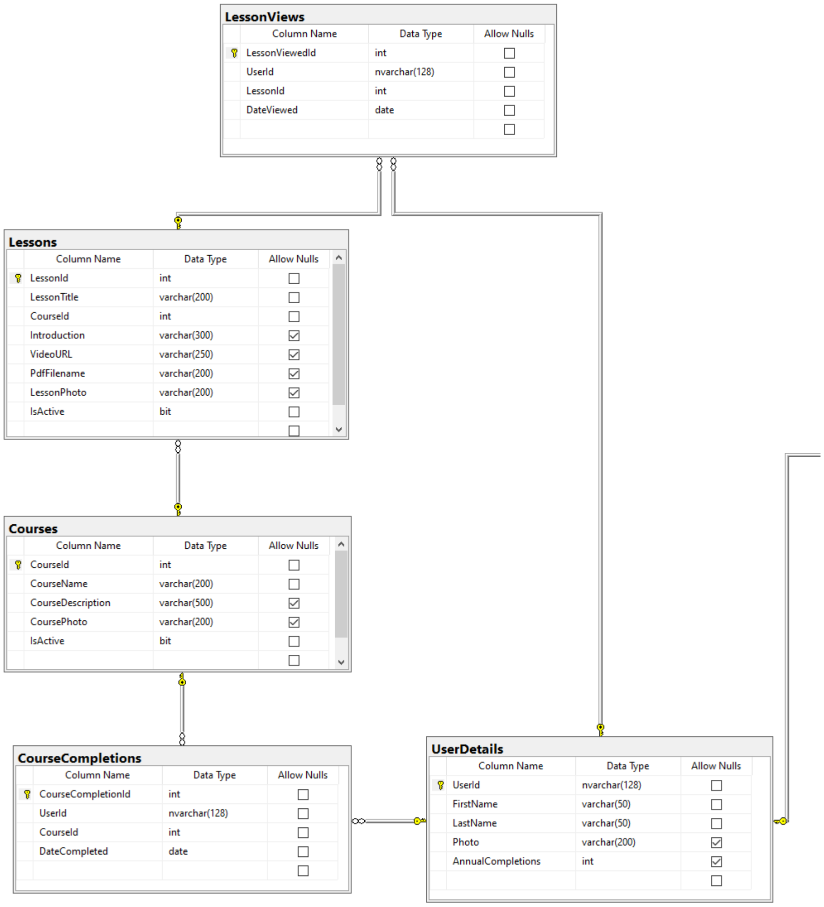
Presented on 8/19/2021

I've always had a heavy interest in technology, from editing videos to building computers. Naturally, I've always been intrigued by the process in how some of my favorite applications, games and websites are created. After taking the leap into web development in Centriq, I quickly found myself enjoying and excelling in the logical, mathematical backend work. And so, I wanted my final project to really challenge me in that aspect and let me build the individual systems to conjoin like a jigsaw puzzle.

For my final project, I decided to create a Learning Management System. This app is designed to provide a platform for employees to browse and take courses created by their manager; and track their progress.

Learning Management System (LMS) is an MVC web application with built in Role Security, User Registration, and a framework to add, edit, and delete entries on a remote repository.

The following pages will provide a brief description of the main controllers, followed by how we progressed through the building of the application.



LessonViews

	Column Name	Data Type	Allow Nulls
🔑	LessonViewedId	int	□
	UserId	nvarchar(128)	□
	LessonId	int	□
	DateViewed	date	□
			□

Lessons

	Column Name	Data Type	Allow Nulls
🔑	LessonId	int	□
	LessonTitle	varchar(200)	□
	CourseId	int	□
	Introduction	varchar(300)	☑
	VideoURL	varchar(250)	☑
	PdfFilename	varchar(200)	☑
	LessonPhoto	varchar(200)	☑
	IsActive	bit	□
			□

Courses

	Column Name	Data Type	Allow Nulls
🔑	CourseId	int	□
	CourseName	varchar(200)	□
	CourseDescription	varchar(500)	☑
	CoursePhoto	varchar(200)	☑
	IsActive	bit	□
			□

CourseCompletions

	Column Name	Data Type	Allow Nulls
🔑	CourseCompletionId	int	□
	UserId	nvarchar(128)	□
	CourseId	int	□
	DateCompleted	date	□
			□

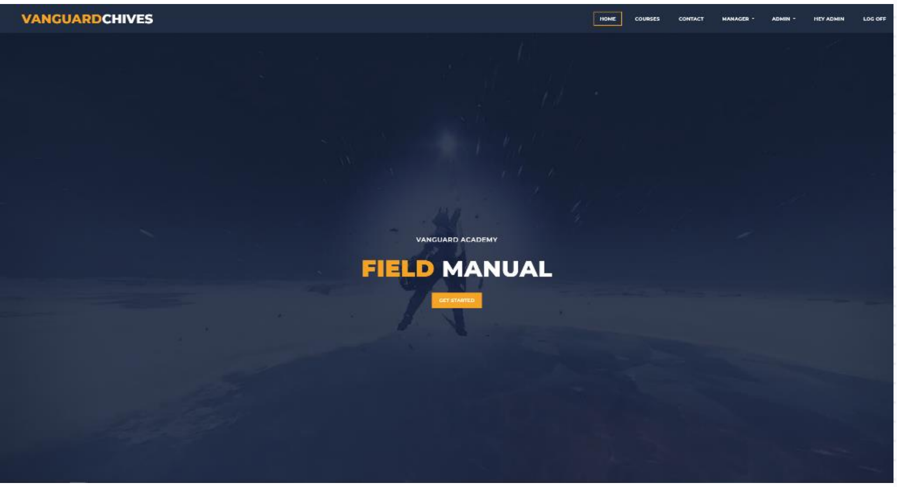
UserDetails

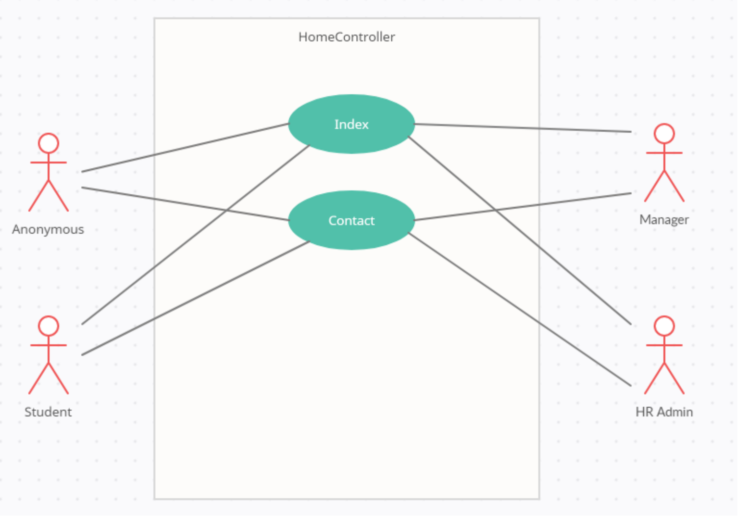
	Column Name	Data Type	Allow Nulls
🔑	UserId	nvarchar(128)	□
	FirstName	varchar(50)	□
	LastName	varchar(50)	□
	Photo	varchar(200)	☑
	AnnualCompletions	int	☑
			□

Data Overview

The database for this project comprises of 5 main tables:

1. Courses
    a. A target to attach lessons to, and the main tracker of progress for the User
2. Lessons
    a. Holds individual learning information for the user to consume, tracked to their parent courses through their ID
3. Lesson Views
    a. A log of each lesson a User has viewed, organized by the User’s Id
4. Course Completions
    a. A log of each course a User has completed, primarily used to track a User’s overall progress
5. User Details
    a. Additional information attached to the ID of a user, containing their Name and Photo





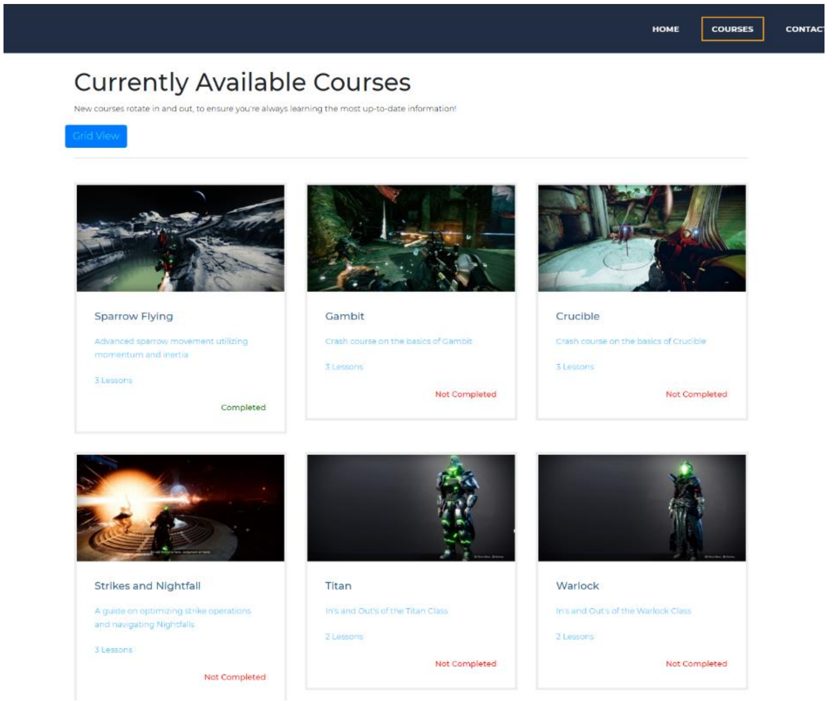
Home Controller

The home controller is the landing page for the site, and the main controller used by those who don’t already have an account.

The index plays a looping intro video I created in the background for an appealing first look at the site.

The nav bar will dynamically change to only show users what they have access to. The special access areas of the nav bar are contained within a dropdown menu for organization and to make the bar easier to navigate on phones, and the entire nav bar will collapse into a drop-down vertical menu on screens especially small.

The contact form would allow unsigned users to contact a manager or admin to help them register their role, but for the purposes of this project, users can instead register on the site.



Currently Available Courses

New courses rotate in and out, to ensure you're always learning the most up-to-date information!

Sparrow Flying
Advanced sparrow movement utilizing momentum and inertia
3 Lessons
Completed

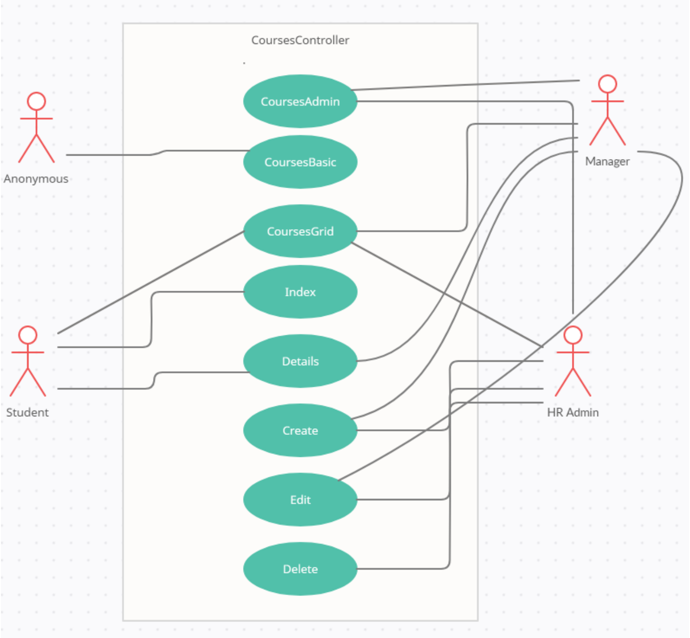
Gambit
Crash course on the basics of Gambit
3 Lessons
Not Completed

Crucible
Crash course on the basics of Crucible
3 Lessons
Not Completed

Strikes and Nightfall
A guide on optimizing strike operations and navigating Nightfalls
3 Lessons
Not Completed

Titan
In's and Out's of the Titan Class
2 Lessons
Not Completed

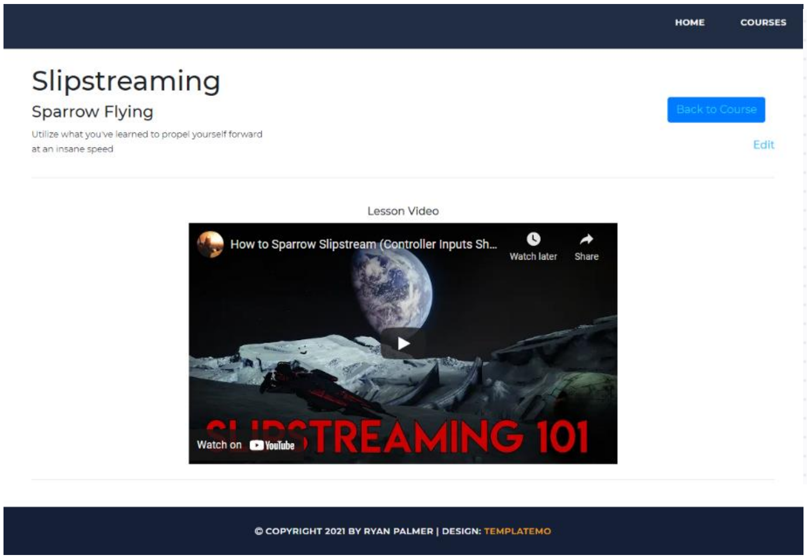
Warlock
In's and Out's of the Warlock Class
2 Lessons
Not Completed



Courses Controller

The courses controller houses most of the main functionality of the application. Upon entering the Index for the controller, a check is ran to see the state of the user. If not signed in, users are redirected to a version of the courses page in which they cannot click on or interact with any of the content. Managers and Admins are shown a broader display, allowing them to more easily make changes. If the user is simply a Student, they'll be presented with a list of courses as well as pertinent information, and can click on each one to take their classes. In addition, all signed-in users have the option to switch the list into a grid format for easier visibility, and their most recent selection will be remembered for the duration of their session.

The course details page as been restructured into something like a lesson index for the lessons assigned under that class, displaying the status of each lesson and the overall course as they go. Upon completion of a course, an email is sent to notify the manager of the completion.



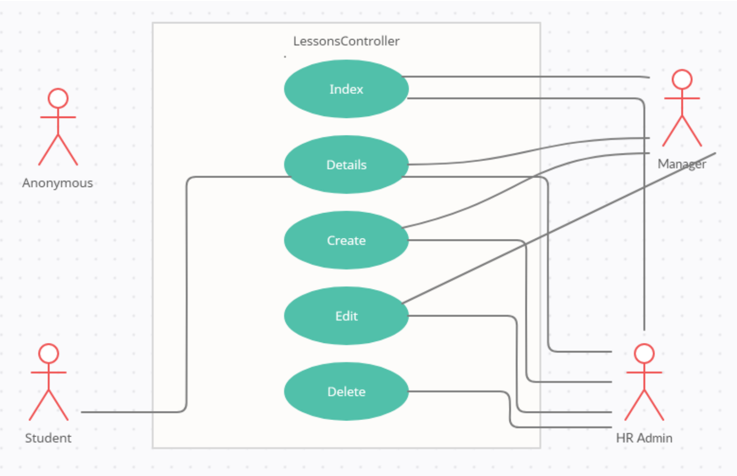
Slipstreaming

Sparrow Flying

Utilize what you've learned to propel yourself forward at an insane speed

How to Sparrow Slipstream (Controller Inputs Sh...

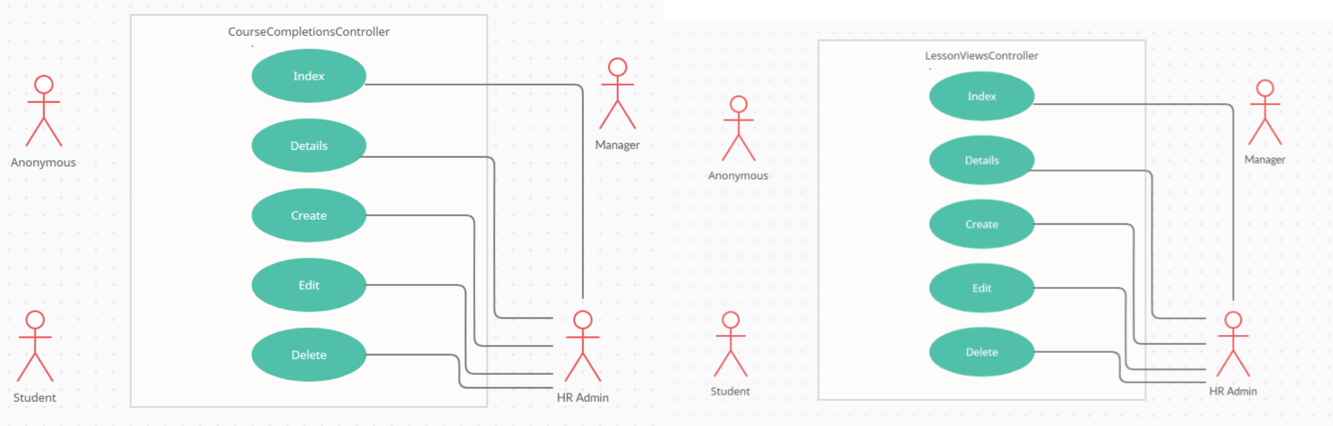
© COPYRIGHT 2021 BY RYAN PALMER | DESIGN: TEMPLATEMO



Lessons Controller

The lessons controller is largely not used by those below manager status, apart from the details page, which is accessed by clicking into one of the lessons on Course Details. Lesson pages will dynamically display either a YouTube Video, PDF File, both, or neither depending on what was supplied to the lesson upon its creation.

For both Courses and Lessons, Managers can create and edit entries, but must contact an admin to delete one, and should instead simply mark a course as inactive if it's not in use.

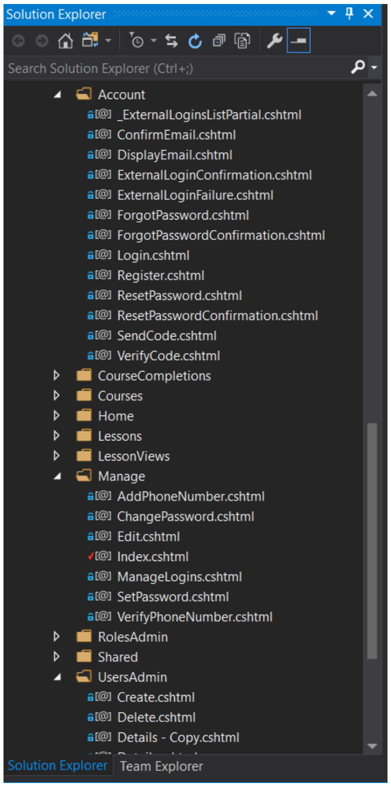


Course Completions and Lesson Views Controller

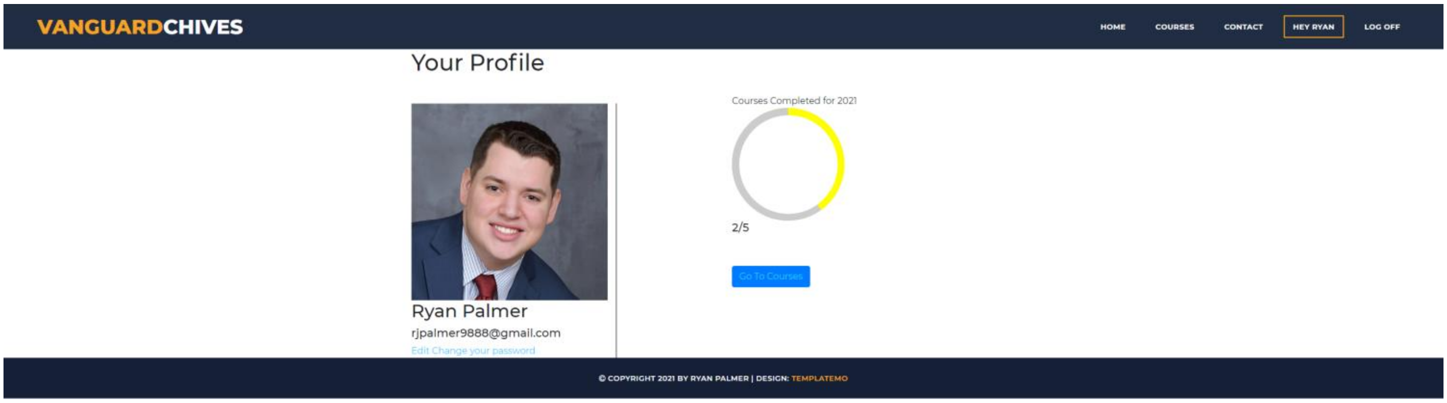
Course completions and Lesson Views are a log of all completions by users. This is only visible to HR, and managers can instead view student progress on their User Page, detailed below.

Additional Controller Details

1. User Details
   a. Users have their own custom name and photo when they sign up through the Register action in the AccountController
   b. This can be seen by clicking on the Users name in the nav bar, taking them to a rebuilt Index in the ManageController
   c. In addition to showing their user info, a dynamic jQuery progress bar will display their annual progress in course completion
2. Users Admin
   a. Managers have access to an index of all currently signed-up users. Checking the details of any user will display a similar page to the Manage Index, displaying the user's info and current progress.
   b. Admins can make changes to users if need as well, just as users can make changes to their own profiles



```
Account
    _ExternalLoginsListPartial.cshtml
    ConfirmEmail.cshtml
    DisplayEmail.cshtml
    ExternalLoginConfirmation.cshtml
    ExternalLoginFailure.cshtml
    ForgotPassword.cshtml
    ForgotPasswordConfirmation.cshtml
    Login.cshtml
    Register.cshtml
    ResetPassword.cshtml
    ResetPasswordConfirmation.cshtml
    SendCode.cshtml
    VerifyCode.cshtml
CourseCompletions
Courses
Home
Lessons
LessonViews
Manage
    AddPhoneNumber.cshtml
    ChangePassword.cshtml
    Edit.cshtml
    Index.cshtml
    ManageLogins.cshtml
    SetPassword.cshtml
    VerifyPhoneNumber.cshtml
RolesAdmin
Shared
UsersAdmin
    Create.cshtml
    Delete.cshtml
    Details - Copy.cshtml
```



Your Profile

Ryan Palmer
rjpalmer9888@gmail.com
Edit Change your password

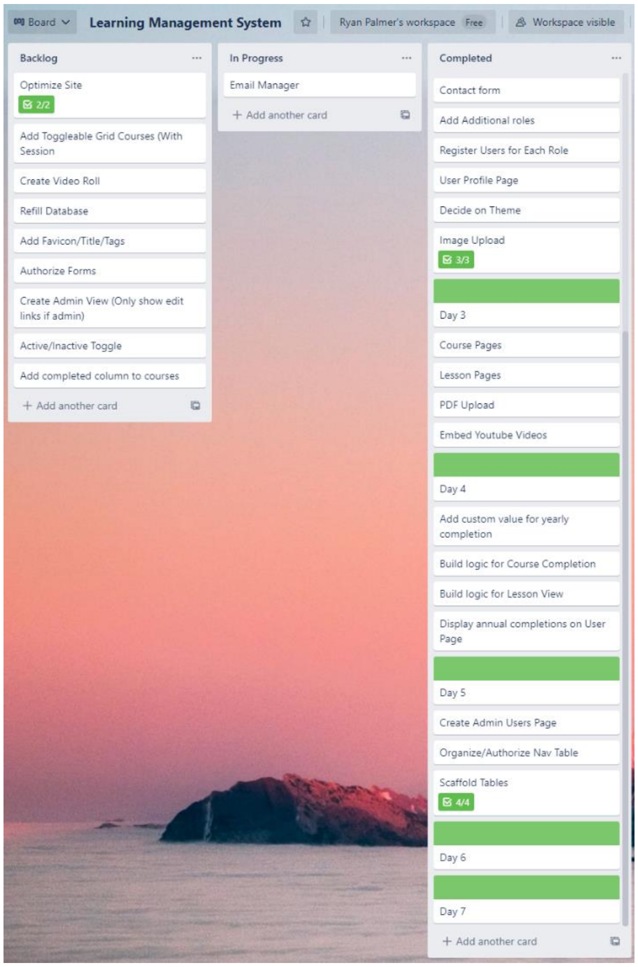
Courses Completed for 2021
2/5

© COPYRIGHT 2021 BY RYAN PALMER | DESIGN: TEMPLATEMO

Sprint Workflow

I managed my goals and tasks through a Trello board, prioritizing more base level features to allow me to build on more advanced components.

Tasks were cataloged day by day to better organize the pace of progress

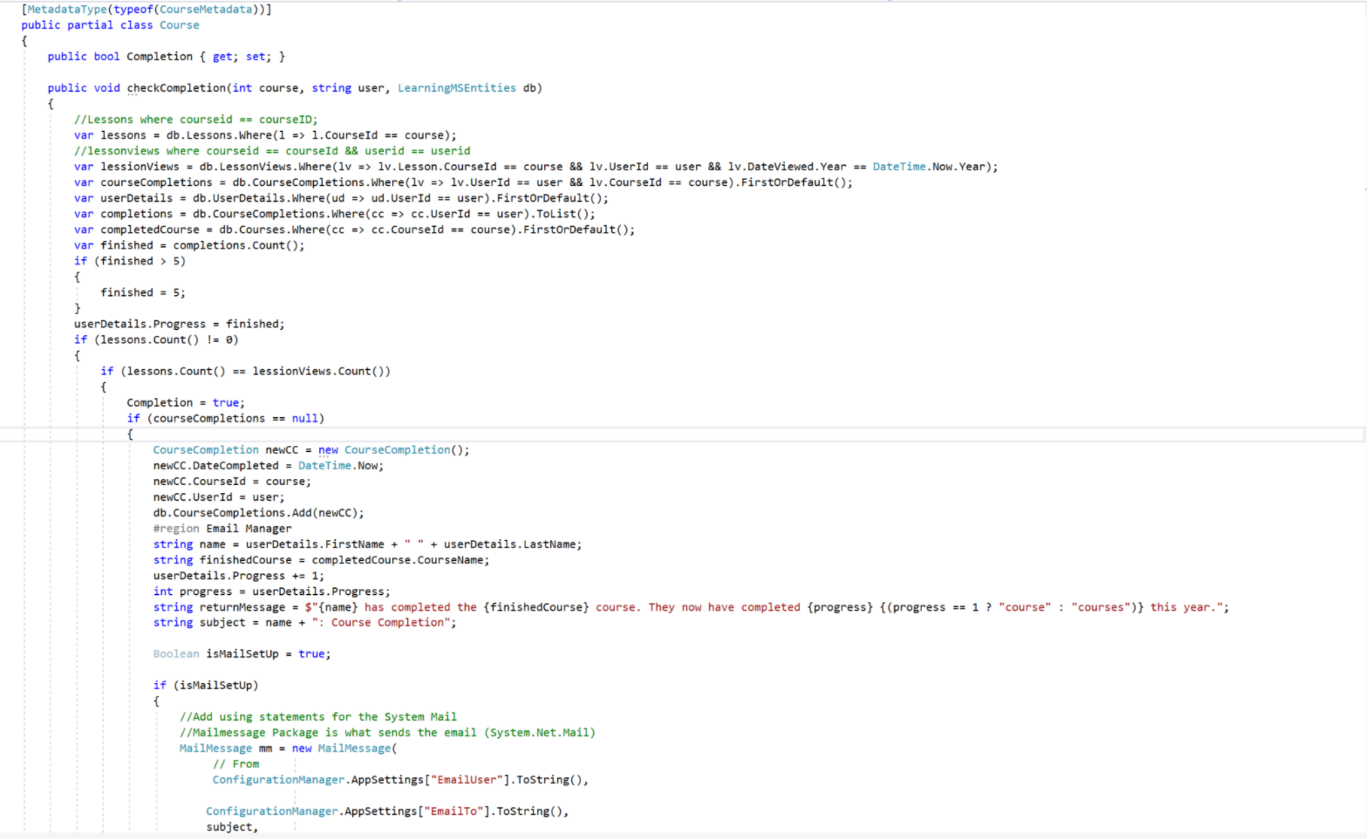


Learning Management System

Backlog
- Optimize Site ☑ 2/2
- Add Toggleable Grid Courses (With Session
- Create Video Roll
- Refill Database
- Add Favicon/Title/Tags
- Authorize Forms
- Create Admin View (Only show edit links if admin)
- Active/Inactive Toggle
- Add completed column to courses

In Progress
- Email Manager

Completed
- Contact form
- Add Additional roles
- Register Users for Each Role
- User Profile Page
- Decide on Theme
- Image Upload ☑ 3/3
- Day 3
- Course Pages
- Lesson Pages
- PDF Upload
- Embed Youtube Videos
- Day 4
- Add custom value for yearly completion
- Build logic for Course Completion
- Build logic for Lesson View
- Display annual completions on User Page
- Day 5
- Create Admin Users Page
- Organize/Authorize Nav Table
- Scaffold Tables ☑ 4/4
- Day 6
- Day 7



```
[MetadataType(typeof(CourseMetadata))]
public partial class Course
{
    public bool Completion { get; set; }

    public void checkCompletion(int course, string user, LearningMSEntities db)
    {
        //Lessons where courseid == courseID;
        var lessons = db.Lessons.Where(l => l.CourseId == course);
        //lessonviews where courseid == courseId && userid == userid
        var lessionViews = db.LessonViews.Where(lv => lv.Lesson.CourseId == course && lv.UserId == user && lv.DateViewed.Year == DateTime.Now.Year);
        var courseCompletions = db.CourseCompletions.Where(lv => lv.UserId == user && lv.CourseId == course).FirstOrDefault();
        var userDetails = db.UserDetails.Where(ud => ud.UserId == user).FirstOrDefault();
        var completions = db.CourseCompletions.Where(cc => cc.UserId == user).ToList();
        var completedCourse = db.Courses.Where(cc => cc.CourseId == course).FirstOrDefault();
        var finished = completions.Count();
        if (finished > 5)
        {
            finished = 5;
        }
        userDetails.Progress = finished;
        if (lessons.Count() != 0)
        {
            if (lessons.Count() == lessionViews.Count())
            {
                Completion = true;
                if (courseCompletions == null)
                {
                    CourseCompletion newCC = new CourseCompletion();
                    newCC.DateCompleted = DateTime.Now;
                    newCC.CourseId = course;
                    newCC.UserId = user;
                    db.CourseCompletions.Add(newCC);
                    #region Email Manager
                    string name = userDetails.FirstName + " " + userDetails.LastName;
                    string finishedCourse = completedCourse.CourseName;
                    userDetails.Progress += 1;
                    int progress = userDetails.Progress;
                    string returnMessage = $"{name} has completed the {finishedCourse} course. They now have completed {progress} {(progress == 1 ? "course" : "courses")} this year.";
                    string subject = name + ": Course Completion";

                    Boolean isMailSetUp = true;

                    if (isMailSetUp)
                    {
                        //Add using statements for the System Mail
                        //Mailmessage Package is what sends the email (System.Net.Mail)
                        MailMessage mm = new MailMessage(
                             // From
                             ConfigurationManager.AppSettings["EmailUser"].ToString(),

                            ConfigurationManager.AppSettings["EmailTo"].ToString(),
                            subject,
```

Code Snippet: Completion Check

Above is the code used to run a check on the User’s current state of Course Completion. This is a method placed in critical points in controllers to update the status of each course. Upon being called, the method will grab the ID of the course being looked at, get a count of how many lessons are in that course, and check that count against how many lessons have been viewed by the User in that course. If the numbers match, the course is marked as completed, a log is created, and an email is sent to notify the manager.

Thanks for your interest in my site! I’m really pleased with how it came out and it was a fun challenge to build it. For my next project, I’d like to take a run at implementing foreign API from Bungie to create an application that interacts with you while you play the game Destiny. On the following page is my resume, if anything here has caught your interest, I’d love the opportunity to speak with you!

Ryan J. Palmer

ryanjpalmer.net

Full Stack Web Developer

6 Campus Edge Drive, Edwardsville IL 62025
618-531-2417/rjpalmer9888@gmail.com

Bright technologist who quickly adapts knowledge into performance blended with innovative approaches. Consistently driven to complete tasks beyond the expectations of my leaders. New to the industry, but my innate aptitude & pace of learning exceeds most others.

Technical Experience

Full Stack Web Development	Centriq, LaunchCode
GitHub	Centriq, LaunchCode
.NET Framework	Centriq
Database Management	Centriq

Secondary Skills

Problem Solving, Communications, Leadership, Resource Management, MS Office/G-Suite

Education

College coursework – University of Illinois (2017)

CS105 Intro Computing: Non-Tech
CM101 Public Speaking
CM102 Introduction to Communication
AFAS 102 Leadership Laboratory

LaunchCode – LC101 full stack web developer course (2019)

eLearning – PluralSight 25 hours (2020)

Additional self-paced learning in courses such as: JavaScript, Core Python: Organizing Larger Programs, Python Best Practices for Code Quality, Full Stack Web Development with Python, Managing Python Packages and Virtual Environments, Python: The Big Picture, Program w/C++

Centriq – Full Stack Web Development (Current)

A Full-Stack Web Development course dedicated to not only arming students with the technical knowledge of the field, but also the knowhow to find a career. Taught in HTML, CSS, C#, JavaScript, and SQL, in addition to teaching coding practices such as Agile

Work Experience

Amazon Picking/Amazon Delivery (2020 - 2021)

- Utilized company software to accurately and quickly manage products
- Dynamically optimized efficiency in a self-paced daily task

AGP Audio Visual Events (2019 - 2020)

- Managed high-end technical gear configuration, budget, schedules, & small team
- Consulted clients to assess needs, set expectations & assist event attendees

Best Buy – Sales (2018 - 2019)

- Maintained an evolving knowledge of technologies and products
- Gained & understood customer’s needs to support effective tech applications