

CARTER CROFT

<https://www.linkedin.com/in/carter-croft-45621b210/> | carterncroft@gmail.com



Reservation System

FixIt4U

fsdp.croftcoding.com

June 16th, 2021

ABOUT MY CHOICE OF PROJECT:

This final project is the product of all that I have learned during my time at Centriq. I chose the reservation system and made the theme an instrument repair shop called FixIt4U. I did this because I enjoy playing guitar in my spare time and I (unfortunately) have experience with parts of my instrument working incorrectly, or not at all. While working on this project, I discovered a few things about my work process. It is extremely useful for me to create an outline of the things I need to work on throughout the development process. For this, I utilized Trello as a sort of checklist to complete to ensure I had completed every brief requirement and anything else I wanted to implement. After completing this project, I am much more confident in my abilities as a developer. I found that implementing different pieces of functionality and thinking through different approaches for the problem at hand is something that I really enjoy.

PROJECT REQUIREMENTS:

The requirements I had to meet during this project were as follows:

There are 3 levels of access: Users, Employees, and Administrators

- Users must have CRUD functionality for their assets.
- Users must be able to upload an image for each asset.
- Users must be able to place reservations for an asset.
- Users must be able to see the reservations they have placed and the assets they have created within the system.
- Each Location has a maximum number of reservations available on a certain date.

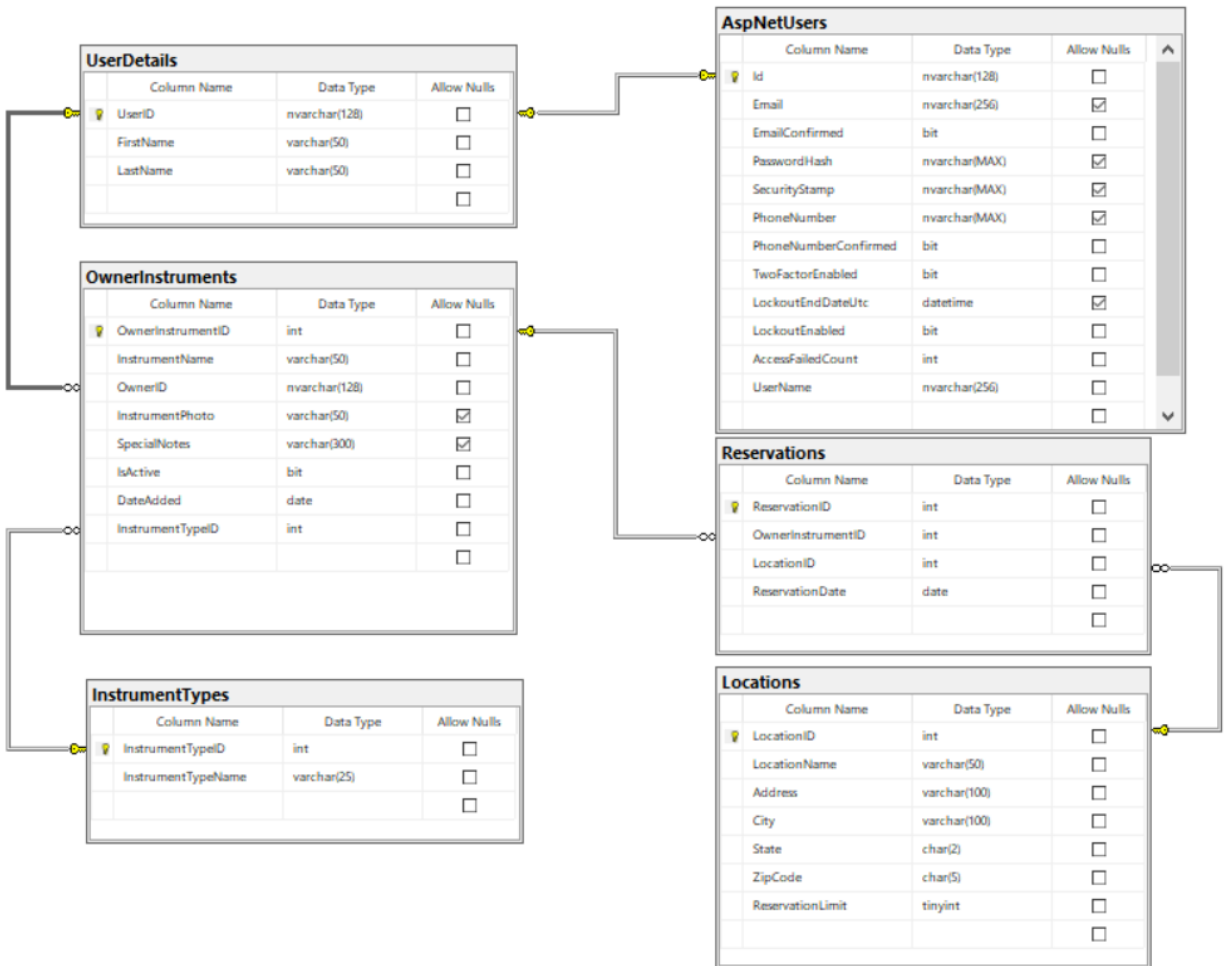
Employees:

- Employees must be able to edit notes on a user's asset.
- Employees must be able to see all reservations.

Admins:

- Admins must have CRUD functionality for Locations.
- Admins must be able to override the reservation limit at any location.
- Admins must be able to Add/Update notes for user's assets.

DATABASE SCHEMA:



CODE SNIPPETS:

```
#region AJAX Create
[HttpPost]
[ValidateAntiForgeryToken]
public JsonResult AjaxCreate(Reservation reservation)
{
    if (ModelState.IsValid)
    {
        var location = db.Locations.AsNoTracking().Where(l => l.LocationID == reservation.LocationID).FirstOrDefault();

        //Below if statement checks the number of reservations made on a specific day at a given location to ensure that the reservation limit is not exceeded on that day.
        if (location.Reservations.Where(r => r.ReservationDate == reservation.ReservationDate).ToList().Count >= location.ReservationLimit && !User.IsInRole("Admin"))
        {
            throw new Exception();
        }
        else
        {
            db.Reservations.Add(reservation);
            db.SaveChanges();
        }
    }

    return Json(new {
        Location = reservation.LocationID,
        Instrument = reservation.OwnerInstrumentID,
        Date = reservation.ReservationDate
    });
}
#endregion
```

The above code snippet is for creating a reservation. The branching logic checks to see if the number of reservations at a certain location on a certain date does not exceed the reservation limit at that location, the right side of the AND within the if statement allows Admins to override this limitation. Inside the if statement, I threw an exception which runs the error function within my AJAX code (Figure below, 1).

```
$.ajax({
    url: "/Reservations/AjaxCreate",
    type: "POST",
    data: formdata,
    datatype: "json",
    error: function (e) {
        $("#MessageContent").html("<div class = 'alert alert-danger'>Reservations full on that date at that location. Please try a different date or location.</div>");
    },
    success: function (s) {
        $("#MessageContent").html("<div class = 'alert alert-success'>Successfully added Reservation. Refreshing...</div>");
        // Reset the form
        $("#ReservationCreateForm")[0].reset();
        setInterval('location.reload()', 1000);
    }
});
```

- 1) This error function populates a div with a message that tells the user that the reservations are full on that day.

```
<td class="text-center">
    @Html.ActionLink("Edit", "Edit", new { id = item.ReservationID }, new { @class = "btn btn-sm btn-dark" })
    @Html.ActionLink("Details", "Details", new { id = item.ReservationID }, new { @class = "btn btn-sm btn-dark" })
    if (item.ReservationDate - DateTime.Now.Date <= TimeSpan.FromDays(7) && (User.IsInRole("Customer") || User.IsInRole("Employee")) && item.OwnerInstrument.OwnerID ==
        currentUser)
    {
        @Html.ActionLink("Delete", "Delete", new { id = item.ReservationID }, new { @class = "btn btn-sm btn-dark", disabled = "true" })
        <button class="btn btn-sm text-white deleteDisabled" title="You are unable to delete reservations within a week of the Reservation Date. Contact an admin if you
            really want to delete this reservation." data-toggle="tooltip" data-placement="top">Delete</button>
    }
    else if (User.IsInRole("Admin") || item.OwnerInstrument.OwnerID == currentUser)
    {
        @Html.ActionLink("Delete", "Delete", new { id = item.ReservationID }, new { @class = "btn btn-sm btn-dark" })
    }
    if (User.IsInRole("Employee"))
    {
        @Html.ActionLink("Edit Notes", "Edit", "OwnerInstruments", new { id = item.OwnerInstrumentID }, new { @class = "btn btn-sm btn-dark", id = "editButton" })
    }
    else if (User.IsInRole("Admin"))
    {
        @Html.ActionLink("Edit Instrument", "Edit", "OwnerInstruments", new { id = item.OwnerInstrumentID }, new { @class = "btn btn-sm btn-dark", id = "editButton" })
    }
</td>
```

One additional piece of functionality I added was restricting when a user could delete a reservation they have made. If they try and delete the reservation within a week of the day they have reserved, the button for deleting said reservation is greyed out and unclickable.

USE CASE DIAGRAM:

	Anonymous	User	Employee	Admin						
Home										
Index	x	x	x	x						
Contact	x	x	x	x						
Owner Instruments										
Index		x	x	x						
Inactive Index			x	x						
Create		x	x	x						
Edit		x	o	x						
Details		x	x	x						
Deactivate		x	x	x						
Reservations										
Index		o	x	x						
Create		o	o	x						
Edit		x	o	x						
Details		x	x	x						
Delete		x	x	x						
Locations										
Index		x	x	x						
Create				x						
Edit				x						
Details		x	x	x						
Delete				x						
Instrument Types										
Index				x						
Create				x						
Edit				x						
Details				x						
Delete				x						

Legend
 Blank means no access to View
 x denotes full access
 o denotes limited access

Limited Functionality:

Employees:

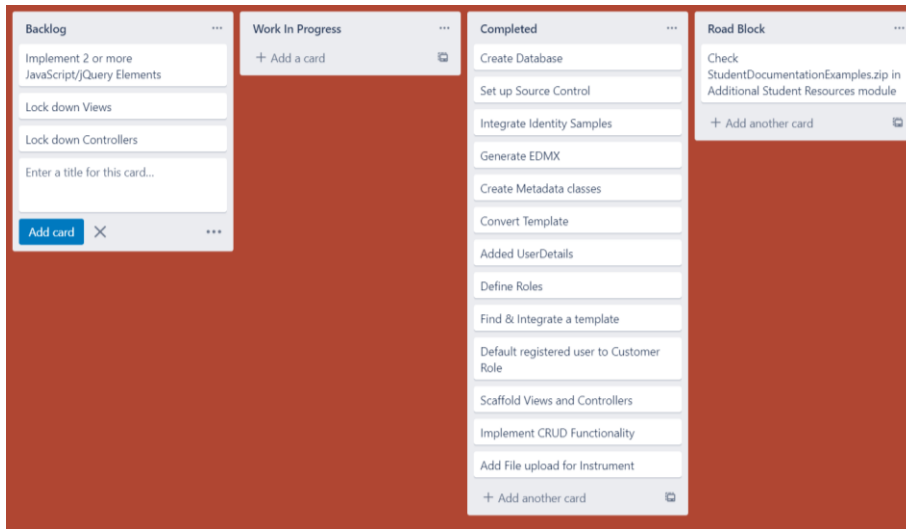
- Employees have access to all CRUD functionality on any instruments that belong to them. Otherwise, they can only edit the notes, view instrument details, and deactivate instruments.
- Employees are able to create/edit reservations as long as the reservation limit isn't exceeded.
- Employees are only able to delete their reservation as long as the reservation date is further than a week away.

Users:

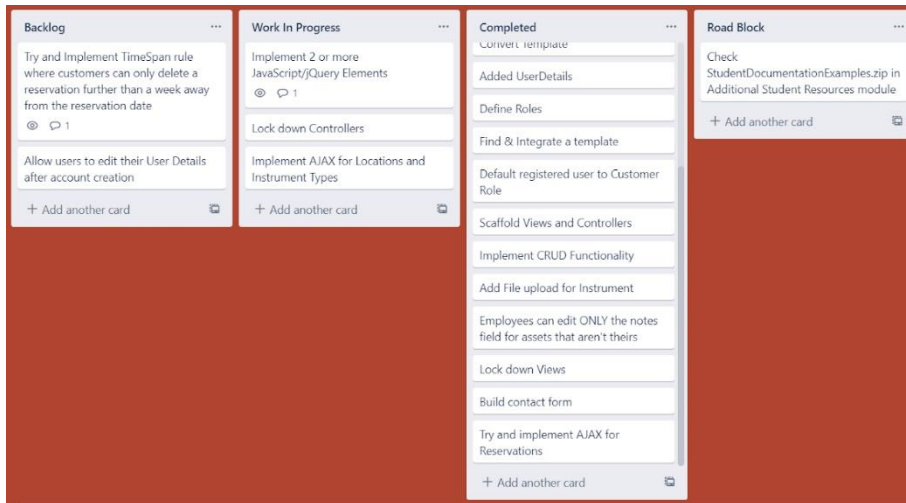
- Users are able to see only their reservations on the reservation index.
- Users are only able to create a reservation as long as the reservation limit isn't exceeded.
- Users are only able to delete their reservation as long as the reservation date is further than a week away.

STAYING ORGANIZED:

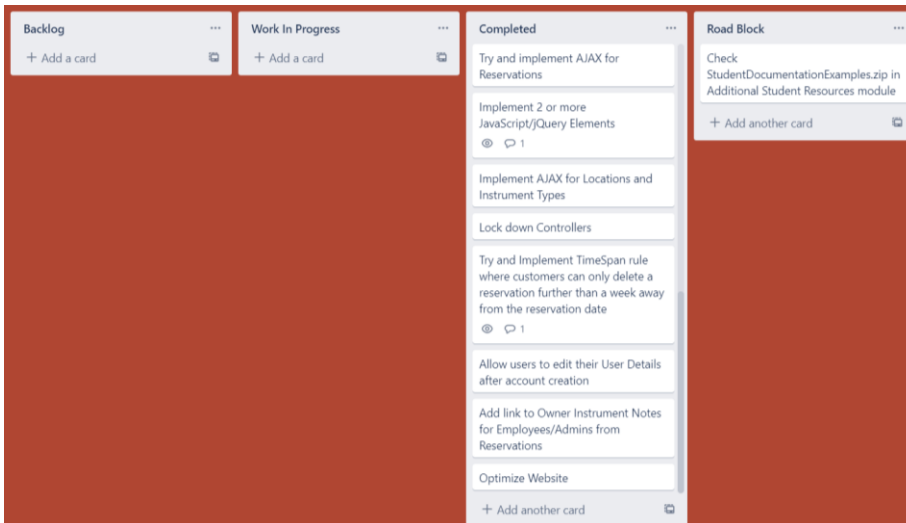
At the start:



Midway through:



Finished Up:



CARTER CROFT

carterncroft@gmail.com · 913-749-9474 · <http://croftcoding.com/>

LinkedIn: <https://www.linkedin.com/in/carter-croft-45621b210/>

FULL STACK DEVELOPER

.NET Framework

Full Stack Web Application Design

C# Fundamentals

Database Management

C++ Fundamentals

Unit Testing

Software Development Lifecycle

Agile Scrum Methodology

Continued Growth in IT Knowledge

Pair Programming

TECHNICAL AND PROFESSIONAL SKILLS

Front End: HTML5, JavaScript, jQuery, jQueryUI, CSS3, Responsive/Mobile Web Development, Bootstrap, ReactJS

Middle Tier: Visual Studio, C#.NET, ASP.NET, LINQ, MVC, EF

Back End: ADO.NET, SQL, SQL Server, SSMSE

Professional Skills: Troubleshooting, Critical Thinking, Communication, Project Management Fundamentals, Teamwork, Leadership

INDEPENDENT DEVELOPMENT PROJECTS

- **Personal Site:** Created a personal website used to showcase my work, resumé. In order to create this website, I converted a website template and used HTML5, CSS3, Bootstrap and JavaScript to customize my site.
- **Small Company Startup:** Created a mockup website based on a small startup company's project brief. This project used basic HTML and CSS.
- **Dungeon Application:** Created a console application using C# which is a text-based game. The player is in a "dungeon" where they fight a variety of monsters with different combat abilities.
- **StoreFront:** Created a secure application for managing product data. Application is built to simulate an online store front with a shopping cart. Administrators are able to manage product, category and vendor data.
- **S.A.T. Scheduling Administration Tool:** Created a secure application for managing student enrollment data. Application is built to simulate an online class scheduling system. Administrators will have the ability to manage students, courses, scheduled classes, and enrollments.
- **Final Project:** Created a secure data-driven ASP.NET MVC application from design through deployment for managing the tracking and organization of reservations within an instrument repair company. Administrators have the ability to manage employee, location data and all details relating to assigned reservations, instruments, and types of instruments.

TECHNICAL TRAINING

Full Stack Web Developer Program, Centriq Training
Kansas City, KS

Graduated 06/2021

- MVC Framework
- Trouble Shooting & Debugging
- Source Control
- Agile/Scrum (Created Team Project)
- Website Deployment
- Pair Programming
- Code Review
- .NET Framework

WORK EXPERIENCE

Shipping Manager, Dangerous Goods Association
(DGA) Ltd.
Overland Park, KS

12/2016 -- Present

- Oversee 8+ weekly incoming orders and their fulfillment
- Manage a database of hundreds of products
- Utilize a database to create packing lists
- Responsible for processing orders via UPS WorldShip

Landscaper, Garcia's Lawn and Landscape, LLC.
Kansas City, KS

03/2017 – 04/2018

- Worked with a team of six to accomplish a common goal
- Oversaw general appearance and health of gardens and trees
- Learned to use different motorized gardening and landscaping tools