

FINAL  
WIREFRAMES & SCREEN LAYOUTS & DB ARCH  
FOR GREENVILLE LICENSE MANAGER (GEM.LM)



John S. Stritzinger  
Computer Science 242 – Wireframe/Page Layout Requirements  
Spring 2026

## Contents

SECTION 1. EXECUTIVE OVERVIEW .....	5
1.1    Definitions of Wireframes.....	5
1.2    Figma Design.....	5
1.3    Figma Make.....	5
1.4    Initial Wireframes Based on GitHub Project Management Theme for 1 <sup>st</sup> half of the course.....	5
1.5    Greenville License Manager as part of Greenville Enterprise Manager .....	5
SECTION 2. INITIAL WIREFRAME LAYOUTS -> GIT.IO THEME .....	6
2.1    Scope of Work.....	6
2.2    Scope Confusion.....	6
2.2    Current Github Site .....	7
2.3    Layout For The site.....	7
2.4    Layout of the Side Panel .....	8
2.5    Layout of the Home Main Page .....	8
2.6    Layout of the Personal Main Page .....	10
2.7    Layout of the Assignments Main Page.....	11
2.8    Layout of the Projects Main Page .....	11
2.9    Layout of the Render Page.....	12
2.10    Layout of the Github Page. ....	13
2.11 FEEDBACK FROM THE INSTRUCTOR.....	14
2.12 CHANGED SCOPE OF WORK.....	14
2.13 SECONDARY REVIEW WITH THE INSTRUCTOR ON 02/06/2026 JUST DAYS BEFORE DUE DATE....	14
3.0 RENDER APIs IN PRODUCTION.....	15
3.1 Executive Summary.....	15
3.2 Porting And New Code from Azure and Google Cloud to Render .....	15
3.3 SCHEMA DESIGN FOR THE PROJECT .....	15
3.4 SWAGGER AND EXAMPLE OUTPUT FOR THE PROJECT – USERS.....	17
3.5 DIRECT ACCESS TO THE API FROM AN APPLICATION RETURNS A JSON .....	17
SECTION 4.0 HOSTING.....	18
4.1 Render Hosting of Figma Make Output .....	18
4.2 Render Hosting of John Stritzinger's Enhanced Project Code – CG UI Design.....	18

4.3 Render Hosting of John Stritzinger's Enhanced Project Code – JSS Redesign .....	19
4.4 Live Database Queries Working:.....	20
4.5 Advantages.....	20
4.6 Target HTML/JS Site Up on licenses.greenvilleassociates.com .....	20

.....	20
<b>5.0 TARGET GREENVILLE LICENSE MANAGER WIREFRAMES – COMPLETE .....</b>	<b>21</b>
5.1 Executive Overview – Translating Scope of Work to Target Designs. ....	21
5.2 Competitive Analysis.....	21
5.3 Significant Improvements over Competing Designs .....	22
5.4 Structure of Final Workflow includes Up to 18 Pages All of Which are Designed and in near working form.....	23
5.5 Menu Options DropDown & Pages Accessible via Top Nav Bar .....	23
5.16 LOGIN(1).....	24
5.16 LOGOUT(2) .....	24
5.17 User Notices(3) .....	24
5.17 User Help(4) .....	25
5.18(5) .....	25
5.19(6) Welcome(SystemInfo)(6).....	26
5.20 Signup/Registration(7) .....	26
5.21 All Licenses(8) Legacy.....	27
5.22 Corporate Profile(9) .....	28

5.23 ProductReviews(10) .....	28
5.24 LicenseInformation(11).....	29
5.25 Home2(12) – Different View of Products – All Products in All conditions.....	29
5.26 LicenseLogs(13).....	29
5.27 Buy More(14) .....	30
5.28 Upgrade Licenses(15).....	31
5.29 Cartreview(16) .....	31
5.30 Payment Review(17) .....	32
5.31 Card Maintenance(18) .....	32
6.0 RAPID PROTOTYPING .....	32
6.1 Building the API first.....	32
6.2 Rendering From JSON Before Wireframing using Rapid Development is Advised.....	32
6.3 Licenses.GreenvilleAssociates.com completely Rendered story board in itself.....	33
7.0 STAGE WIREFRAME COMPLETE COMMENTS .....	33

## SECTION 1. EXECUTIVE OVERVIEW

### 1.1 Definitions of Wireframes

Wireframes are Screen Layouts spanning an entire application. Although there are multiple ways to do screen layouts for Analysis work, this class has discussed using Figma Design and Make Tools with a combination of HTML and React Pages. That was the requirement for this assignment.

### 1.2 Figma Design

Figma design is similar to other tools from Microsoft and Adobe software and allows a user to create a layer Visual on a canvas. Unlike some tools however Figma has “SnapTo” tools which are built based on a Rendered Device output allowing for developers to make tools for a single type of output.

### 1.3 Figma Make

Figma Make takes a “Figma Design” or any other HTML file, or Graphic that a LLM AI Tool can ingest and builds HTML, React, and other developer finished tools which can be compiled into real programs. Figma has partnered with ClaudeAI which is Anthropic’s AI tool.

### 1.4 Initial Wireframes Based on GitHub Project Management Theme for 1<sup>st</sup> half of the course.

We thought that the instructor had intended Wireframes just to be a demonstration of AI tools which we would later have to copy for our final project, but that initially it would be an extension of the website we have built to show our project states/project management stages. As such our initial submissions and our discussion in Section2 were based on this work product which was also completed as designed. The instructor asked us to rework our presentation before final.

### 1.5 Greenville License Manager as part of Greenville Enterprise Manager

We believe the Greenville License Manager is a sub-component of the Greenville Enterprise Manager Tool and will allow the real-time monitoring of Application Use across an Enterprise which is the first of its kind, using location processing on a per station basis, per region, and hosting shard which are definitions beyond the scope of this document.

## SECTION 2. INITIAL WIREFRAME LAYOUTS -> GIT.IO THEME

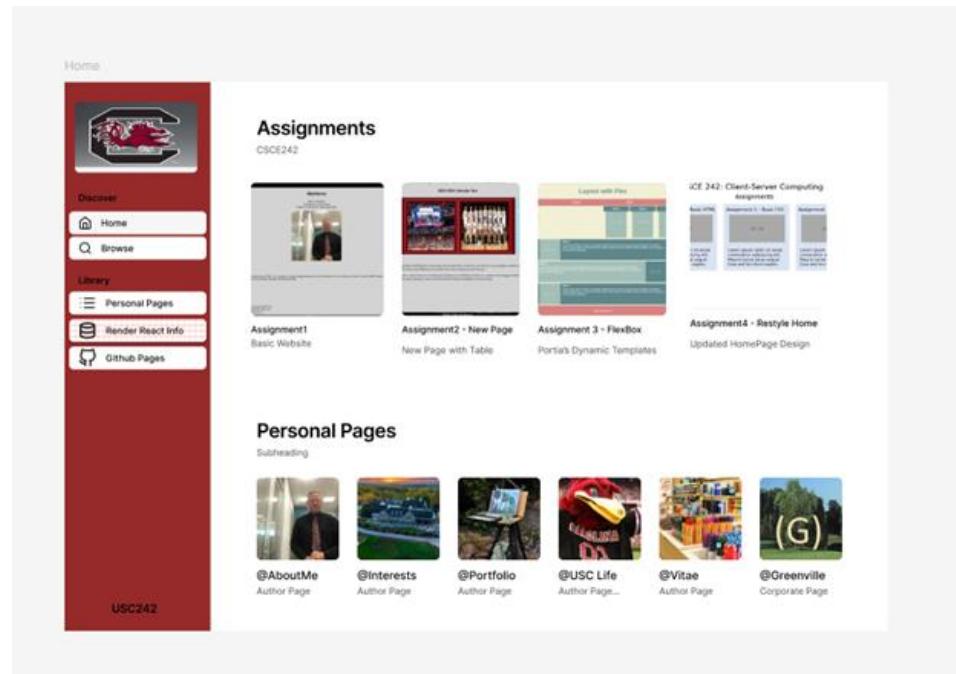


Figure 2.0.1

### 2.1 Scope of Work

Our initial plan for our wireframes was to build a React improvement for Project Management Professionals which would be a generic form of our class assignment.

#### GITHUB.IO – IMPROVEMENTS FOR PROJECT MANAGEMENT PROFESSIONALS

Our instructor told us she didn't believe that was consistent with our project after we had already effectively completed the work.

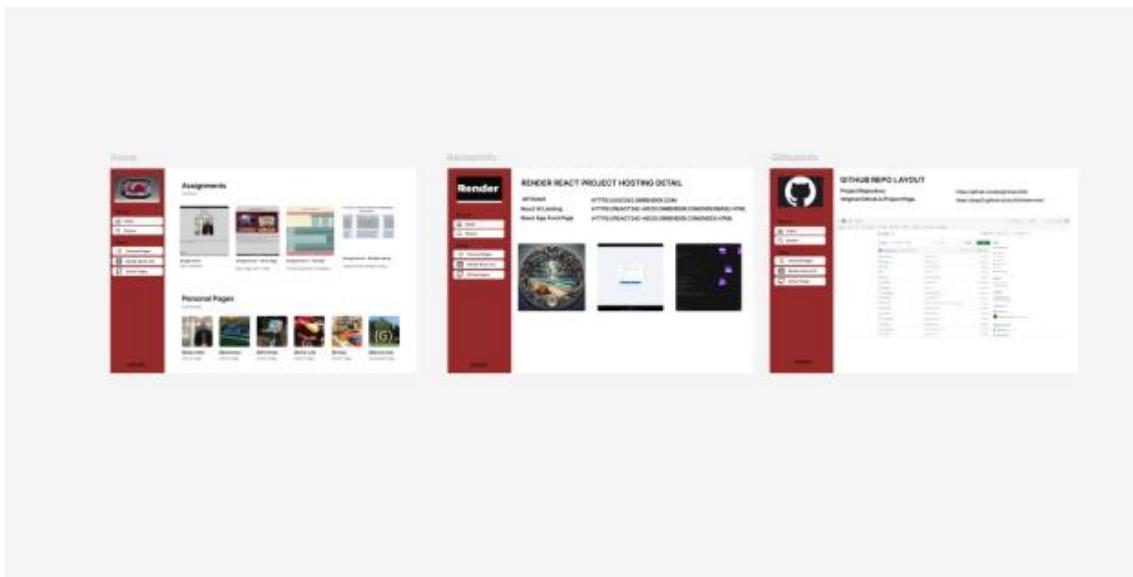
### 2.2 Scope Confusion

We believed that the Project plan we submitted for our project was going to be our End of Term Solution, and not related to the wireframe solution. As such our initial presentations from Figma were for the Github restyling project. We therefore were prepared to rebuild the entire project management site for our class based on the structure of the project on Github.io which does not support Javascript. We believed we could build a new version which would be a generic project tool which would be database driven with java script enabled on Render as the first part of our project. The instructor has said she didn't intend that to be part of the scope, but I believe its valuable anyway as I am a PMP and I think it would help me for other Consulting Deliverables. We have added two tables for Scopes, and Projects which are included in our APIs on Mongo DB which are working and in production.

## 2.2 Current Github Site

The current Github site has approximately 7 personal pages with a a vitae, along with an Assignments Section, and Projects Section. The goal of this assignment was to have a unifying flex design for a simple page environment which would be mobile friendly, while the tasks themselves were in separate Github folders. The result is approximately what is available with a Sharepoint Site just with a bit better change control. The instructor using Git can see exactly whats been changed, and when which is an improvement over Sharepoint. Furthermore older versions can be restored.

## 2.3 Layout For The site



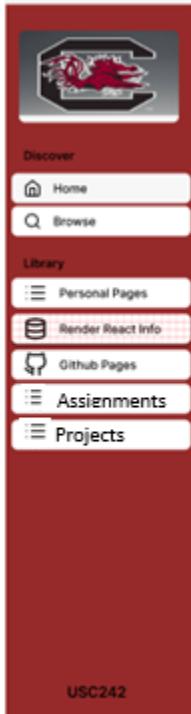
Our initial Wireframe as of above had:

- a) Home Page with Links to Personal Latest Assignments, and Project Stages just like are the requirements for the existing page at jssg33.github.io.
- b) A Side Navigation bar which allowed someone to switch from the Home Page to Personal Pages, to Assignment Pages, and Product Pages with a Sub-Home Page for Each subsection which is an improvement over the current design we have done in class.
- c) This effectively gives us 4 Main Navigation Pages (Home, Personal, Assignments, Projects), all of which we modeled in Figma Design and Two Infrastructure Pages (React, Github).
- d) We then gave Figma Make the Image of our home page, and gave it some guidance on the side nav pages, and it built out a React Structure with the 4 main navigation pages, and the approximately 30 sub pages which could only be reached from the main 4... in approximately 10 pages per sub page, **plus the home page which is a highlighter for the current task underway. This would allow us to move old tasks off the homepage when**

they are completed. This could be done by the database by only presenting on the dashboard tasks which are set to “underway” or not complete, and ordered by those sub-tasks.

e) The Completed First Pass of the Figma Design Board

## 2.4 Layout of the Side Panel



Other than the Home Page:

**There are three main Navigation Pages:**

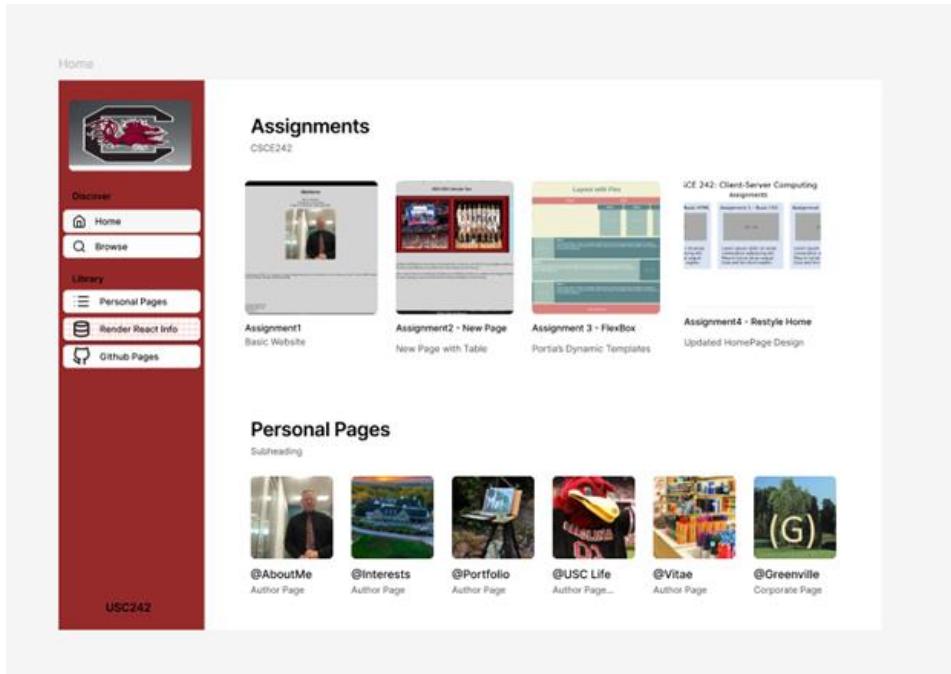
Personal Pages  
Assignments  
Projects

**And Two Infrastructure Pages:**

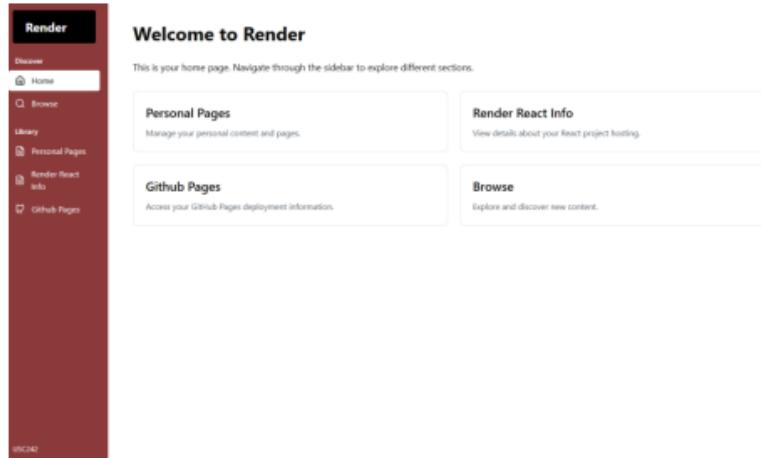
Render Hosting  
GitHub Devops

## 2.5 Layout of the Home Main Page

The envisioned home page would only have Active Assignments as seen in a Mongo DB, and all the personal pages.



AND THE AI VERSION OF THE HOME PAGE IN CLAUDE AFTER 1 PASS



The Home Page Layout in Figma with Claude Enhanced Instructions (Pass 2)

John S. Stritzinger  
Computer Science 242 – Wireframe/Page Layout Requirements  
Spring 2026

We ended with a Beautiful layout which has Personal Links and Quick Links which aren't quite on target, but with working Navigation and Sub Links on real code.

## 2.6 Layout of the Personal Main Page

We only developed the Personal Page after doing so in Figma Make and didn't go back to update the original board which is redundant. We had to prompt Claude to generate Icons which were attractive. However after we got this result we believe our original design was nicer than what Claude delivered from a design perspective. Claude's pages are however in React form and easy to fix.

## 2.7 Layout of the Assignments Main Page

Claude's Layout of the Assignment Page includes 10 Real Sub-Pages which used up our credits, but this assignment is mostly complete in React and we have working code to deploy very soon so it was a helpful corporate exercise. This site can be rebranded by simply changing the Nav Pages Icons, and changing the color of the side bar.

The screenshot shows a web application interface for 'CSCE242'. On the left is a dark red sidebar with a cartoon character icon at the top, followed by navigation links: 'Discover', 'Home', 'Browse', 'Library', 'Personal Pages', 'Assignments' (which is selected), 'Projects', 'Render Hosting', and 'Github Pages'. At the bottom of the sidebar is the text 'USC242'. The main content area is titled 'Assignments' and shows a grid of 10 assignment cards. Each card has a title, a brief description, and three small preview images. The assignments are:

- Assignment 1**: Basic HTML Structure. Description: Introduction to HTML fundamentals and page structure.
- Assignment 2**: CSS Styling Basics. Description: Learning CSS selectors, properties, and layout.
- Assignment 3**: Flexbox Layout. Description: Mastering flexbox for responsive layouts.
- Assignment 4**: JavaScript Fundamentals. Description: Introduction to JavaScript programming.
- Assignment 5**: DOM Manipulation. Description: Working with the Document Object Model.
- Assignment 6**: Event Handling. Description: Interactive web pages with JavaScript events.
- Assignment 7**: Form Validation. Description: Client-side form validation techniques.
- Assignment 8**: API Integration. Description: Fetching and displaying data from external APIs.

Each assignment card also includes 'Personal Pages' and 'Assignment' buttons with sub-options like 'Basic Website', 'New Page', 'New Page with Tables', 'New Page with Forms', 'Assignment 1 - Media', 'Assignment 2 - Media', 'Assignment 3 - Media', 'Assignment 4 - Media', 'Assignment 5 - Media', 'Assignment 6 - Media', 'Assignment 7 - Media', and 'Assignment 8 - Media'. There are also 'Assign' and 'Update' buttons.

## 2.8 Layout of the Projects Main Page

Claude did a much better design that we expected for the Project Phases, setting a completion status, and building a real-time dashboard below.

**Project Phases**  
Development Lifecycle

Phase	Description	Status
Phase I	Project Initialization	Completed
Phase II	Requirements Gathering	Completed
Phase III	Design & Architecture	Completed
Phase IV	Frontend Development	Completed
Phase V	Backend Development	In Progress
Phase VI	Integration Testing	Pending
Phase VII	User Acceptance Testing	Pending
Phase VIII	Deployment Preparation	Pending
Phase IX	Production Deployment	Pending
Phase X	Maintenance & Support	Pending

Completed: 4  
In Progress: 1  
Pending: 5

## 2.9 Layout of the Render Page.

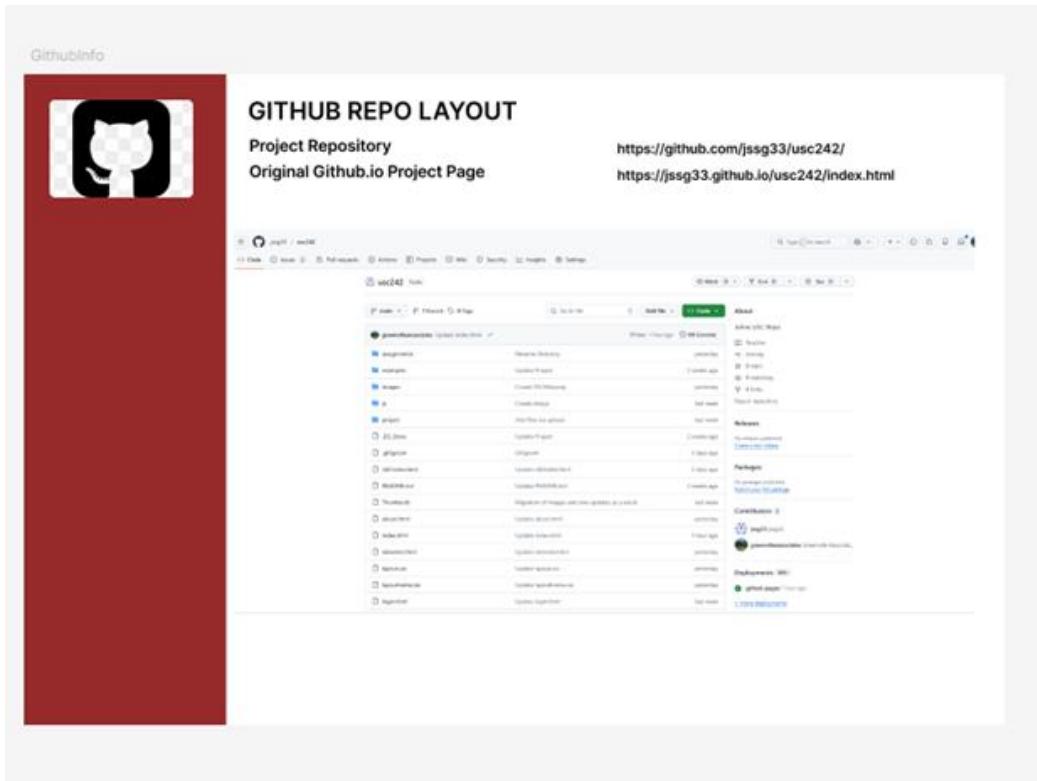
**RenderReact Project Hosting Detail**

**API Detail** <HTTPS://USC242.ONRENDER.COM>

**React UI Landing** <HTTPS://REACT242-H020.ONRENDER.COM/INDEXBASE.HTML>

**React App Front Page** <HTTPS://REACT242-H020.ONRENDER.COM/INDEX.HTML>

## 2.10 Layout of the Github Page.



AND THE AI VERSION OF THE GITHUB PAGE BELOW

The screenshot shows a simulated GitHub Pages interface. On the left, a sidebar titled 'Render' lists options: Discover, Home, Browse, Library, Personal Pages, Render React, and Github Pages (which is selected). The main content area is titled 'GitHub Pages' and includes sections for 'Portfolio Repository', 'Project Documentation', 'GitHub Profile', and 'GitHub Pages Deployment'. The 'Portfolio Repository' section shows a link to 'https://github.com/username/portfolio'. The 'Project Documentation' section shows a link to 'https://username.github.io/docs/'. The 'GitHub Profile' section shows a link to 'https://github.com/username'. The 'GitHub Pages Deployment' section shows 'Status: Active', 'Last Deployed: January 30, 2026', and 'Branch: main'. At the bottom left of the sidebar, it says 'USC242'.

## **2.11 FEEDBACK FROM THE INSTRUCTOR**

The instructor believed we had misunderstood and needed to submit a wireframe which matched our project description which we submitted in Phase I. This means most of the work we did for this section will be in excess of project requirements but it is useful in practice for Greenville Associates Consulting as a general tool which we need for most projects.

The submission made in Section 3 represents the changed requirements.

## **2.12 CHANGED SCOPE OF WORK**

At the completion of Section2, and After completing the production of most of the Code in react, we have moved to trying to complete the same thing for our original project in building a License Manager.

## **2.13 SECONDARY REVIEW WITH THE INSTRUCTOR ON 02/06/2026 JUST DAYS BEFORE DUE DATE**

The instructor reviewed our plan for the LicenseManager, and she believed it needed to include some sales data to be compliant with the scope of the project submitted. In other words, adding one or more pages of marketing collateral to enhanced the content seen in Section 4.0

## 3.0 RENDER APIs IN PRODUCTION

### 3.1 Executive Summary

In our scope of work for the project we discussed building a 30+ table backoffice as we believed that was the minimal function design after completing 547, 590, 567, 587, and 498(1)/498(2). This includes User Mode Functions, Security, Cart Services, Enterprise Tools, and the specific module under development (Parks, ITIL Service Manager, Solution X, Solution Y). A 30 Table Backoffice without any prior work in this area is a significant design and development burden in itself. It is another thing to put in production in C# APIs, Host the API in a responsible manner, and or build Express NodeJS equivalents.

### 3.2 Porting And New Code from Azure and Google Cloud to Render

The instructor determined her target for our environment was student accounts on Render, and MongoDB databases connected to Express. This environment is now complete and in production. Initially we built and checked our work without swagger/OpenAPI, then asked various AI tools if it is supported on Express, and since it is we felt great to get to more or less the same code design for our backoffice supportable on NodeJS as it is on Microsoft Hosting platforms as DotNet Executables. Unlike Microsoft Tools, The Render Environment can actually run Yarn, and Vite/NPM development scripts to build the source code and is superior from a production environment as it doesn't require 16GB of RAM to Run Vscode/Visual Studio. A Beautiful Solution.

GreenLicenseManager 1.1.0 OAS 3.0

API documentation for all legacy endpoints

**Users** User management

**Licenses** License management

Method	Endpoint	Description
GET	/users	Get all users
POST	/users	Create a new user
GET	/users/{id}	Get a user by ID
PUT	/users/{id}	Update a user
DELETE	/users/{id}	Delete a user
GET	/licenses	Get all licenses
POST	/licenses	Create a new license
GET	/licenses/{id}	Get a license by ID

### 3.3 SCHEMA DESIGN FOR THE PROJECT

The current schema which we believe encompasses all the requirements of this project is approximately 31-40 tables on completion which is large project for a single developer by itself. We believe this is only possible as we have been working on a target operating schema for more than 18 months in 6 other classes. We advised the instructor that a 5-6 page project is doable for a single developer in a 2XX class but is not anywhere near production quality deliverables of 547/590 built by teams.

Nevertheless here is our proposed solution.

```
// -----
// ROUTES
// -----



// Users & Accounts (4)
app.use("/users", require("./routes/userRoutes"));
app.use("/api/notices", require("./routes/userNoticeRoutes"));
app.use("/usercontacts", require("./routes/userContactRoutes"));
app.use("/userhelp", require("./routes/userHelpRoutes"));



// Companies & Branches (3)
app.use("/companies", require("./routes/companyRoutes"));
app.use("/branches", require("./routes/branchRoutes"));
app.use("/instances", require("./routes/instanceRoutes"));



// Logging & Security (5)
app.use("/api/apilog", require("./routes/apiLogRoutes"));
app.use("/api/adminlogs", require("./routes/adminLogRoutes"));
app.use("/downloadlogs", require("./routes/downloadLogRoutes"));
app.use("/licenselogs", require("./routes/licenseLogRoutes"));
app.use("/userlogs", require("./routes/userLogRoutes"));



// Licensing (1)
app.use("/licenses", require("./routes/licenseRoutes"));



// Parks (1)
app.use("/parks", require("./routes/parkRoutes"));



// DTOs (2)
app.use("/api/GCPARKS", require("./routes/gcParksRoutes"));
app.use("/api/gcparks", require("./routes/gcParksRoutes")); // lowercase alias
app.use("/api/CGCART", require("./routes/cgCartRoutes"));



// Batch Processing (1)
app.use("/api/batches", require("./routes/batchRoutes"));



// Commerce / Sales (7)
app.use("/products", require("./routes/productRoutes"));
app.use("/reviews", require("./routes/reviewRoutes"));
app.use("/salescatalogue", require("./routes/salesCatalogueRoutes"));
app.use("/invoices", require("./routes/invoiceRoutes"));
app.use("/invoicelineitems", require("./routes/invoiceLineItemRoutes"));
app.use("/payments", require("./routes/paymentRoutes"));
app.use("/refunds", require("./routes/refundRoutes"));



// Cart System (3)
app.use("/cart", require("./routes/cartRoutes"));
app.use("/cartmaster", require("./routes/cartMasterRoutes"));
app.use("/cartitems", require("./routes/cartItemRoutes"));



// Reservations (1)
app.use("/reservations", require("./routes/reservationRoutes"));
```

```

// Cards (1)
app.use("/cards", require("./routes/cardsRoutes"));

// Scopes & Project Tasks (2)
app.use("/scopes", require("./routes/scopeRoutes"));
app.use("/projecttasks", require("./routes/projectTaskRoutes"));

// -----
// TOTAL ROUTES: 31
// -----

```

THESE ALL SUPPORT GET, AND PUT, AND GET, PUT AND DELETE BY ID OUT OF THE BOX (5 SETS OF ENDPOINTS FOR EACH TABLE). This is basic SQL support, and does not support additional items like capacity checks, user notification, and backoffice services which may need additional tables, and lots more Endpoint logic none of which are in scope for this class.

## 3.4 SWAGGER AND EXAMPLE OUTPUT FOR THE PROJECT – USERS

Swagger is returning the User Table Below

No parameters

Responses

Curl

```
curl -X 'GET' \
  'https://api242.onrender.com/users' \
  -H 'accept: application/json'
```

Request URL

<https://api242.onrender.com/users>

Server response

Code	Details
200	Response body

```
[{"_id": "696ce40678b37cfc3677c4ad", "fullname": "John S. Stritzinger", "email": "stritzj@email.sc.edu", "plainpassword": "test12345", "companyId": "V2FED001", "corporateuser": "True", "password": "test1234", "phone": "751-123-4567", "role": "superuser", "status": "active", "updatedAt": "2026-02-02T18:15:31.716Z"}, {"_id": "696ce48e78b37cfc3677c4ae", "fullname": "Portia Planté", "email": "plantec@cse.sc.edu", "plainpassword": "test12345", "companyId": "V2FED001", "corporateuser": "True", "password": "test1234", "phone": "983-867-5309", "role": "superuser", "status": "active", "updatedAt": "2026-02-02T18:10:52.276Z"}, {"_id": "696ce51d78b37cfc3677c4af", "fullname": "Joey Yip", "email": "yip@cse.sc.edu", "plainpassword": "test1234", "companyId": "U02T18:10:52.276Z"}, {"_id": "696ce51d78b37cfc3677c4b0", "fullname": "Brian Hipp", "email": "hipp@cse.sc.edu", "plainpassword": "test12345"}]
```

Download

Response headers

```
access-control-allow-origin: *
alt-svc: h3=":443"; ma=6400
```

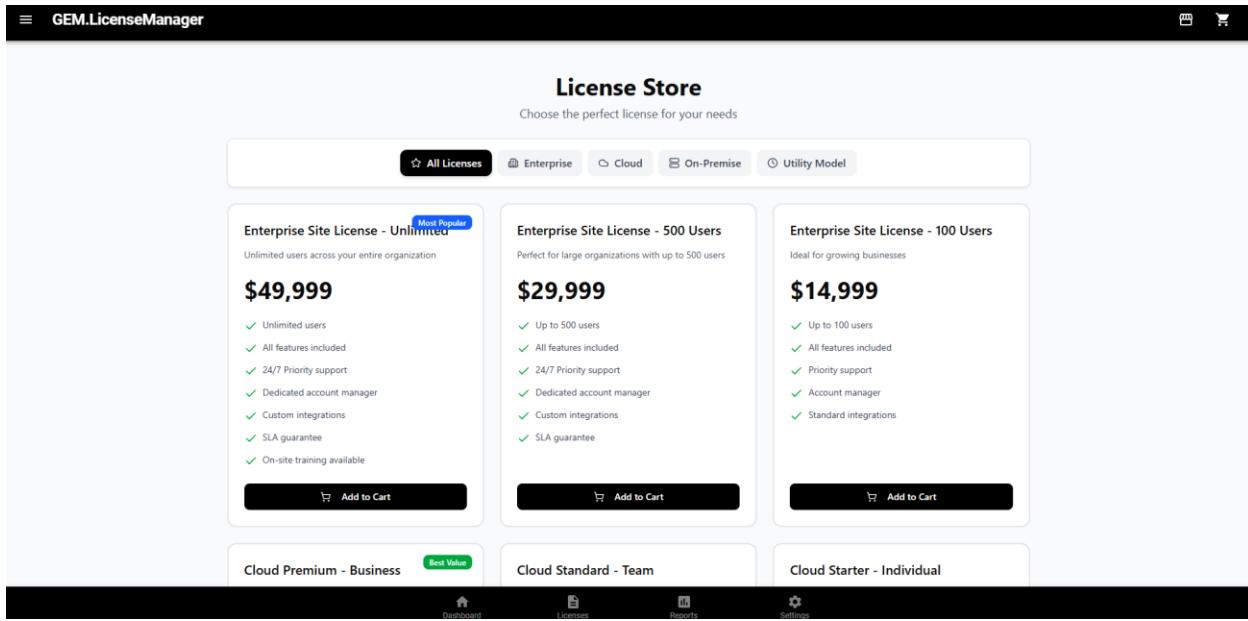
## 3.5 DIRECT ACCESS TO THE API FROM AN APPLICATION RETURNS A JSON

```
[{"_id": "696ce40678b37cfc3677c4ad", "fullname": "John S. Stritzinger", "email": "stritzj@email.sc.edu", "plainpassword": "test12345", "companyId": "V2FED001", "corporateuser": "True", "password": "test1234", "phone": "751-123-4567", "role": "superuser", "status": "active", "updatedAt": "2026-02-02T18:15:31.716Z"}, {"_id": "696ce48e78b37cfc3677c4ae", "fullname": "Portia Planté", "email": "plantec@cse.sc.edu", "plainpassword": "test12345", "companyId": "V2FED001", "corporateuser": "True", "password": "test1234", "phone": "983-867-5309", "role": "superuser", "status": "active", "updatedAt": "2026-02-02T18:10:52.276Z"}, {"_id": "696ce51d78b37cfc3677c4af", "fullname": "Joey Yip", "email": "yip@cse.sc.edu", "plainpassword": "test1234", "companyId": "U02T18:10:52.276Z"}, {"_id": "696ce59978b37cfc3677c4b0", "fullname": "Brian Hipp", "email": "hipp@cse.sc.edu", "plainpassword": "test12345"}]
```

## SECTION 4.0 HOSTING

We are hosting live applications on Render, and on Figma Make for the project. Links are on the Github.io page for the class.

### 4.1 Render Hosting of Figma Make Output



The screenshot shows the 'License Store' section of the GEM License Manager. At the top, there are tabs for 'All Licenses', 'Enterprise', 'Cloud', 'On-Premise', and 'Utility Model'. Below the tabs, there are three main license categories displayed in boxes:

- Enterprise Site License - Unlimited** (Most Popular)  
Unlimited users across your entire organization  
**\$49,999**
  - ✓ Unlimited users
  - ✓ All features included
  - ✓ 24/7 Priority support
  - ✓ Dedicated account manager
  - ✓ Custom integrations
  - ✓ SLA guarantee
  - ✓ On-site training available**Add to Cart**
- Enterprise Site License - 500 Users**  
Perfect for large organizations with up to 500 users  
**\$29,999**
  - ✓ Up to 500 users
  - ✓ All features included
  - ✓ 24/7 Priority support
  - ✓ Dedicated account manager
  - ✓ Custom integrations
  - ✓ SLA guarantee**Add to Cart**
- Enterprise Site License - 100 Users**  
Ideal for growing businesses  
**\$14,999**
  - ✓ Up to 100 users
  - ✓ All features included
  - ✓ Priority support
  - ✓ Account manager
  - ✓ Standard integrations**Add to Cart**

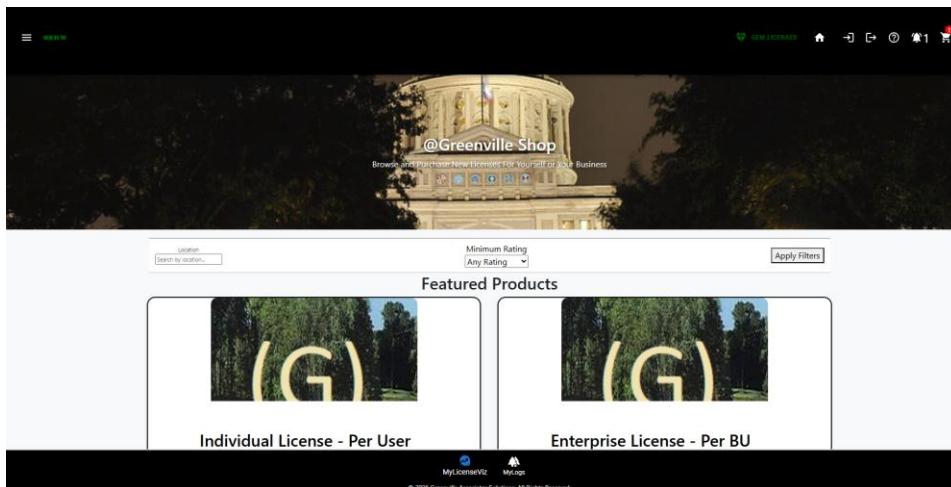
Below these boxes, there are three smaller boxes representing Cloud options:  
**Cloud Premium - Business** (Best Value)  
**Cloud Standard - Team**  
**Cloud Starter - Individual**

At the bottom of the screen, there is a navigation bar with icons for Dashboard, Licenses, Reports, and Settings.

### 4.2 Render Hosting of John Stritzinger's Enhanced Project Code – CG UI Design

We have built a Target Environment on React Material UI pages which is also hosted and nearing completion within a few weeks more than two months in advance. The layout of this page was designed by Capgemini Consulting

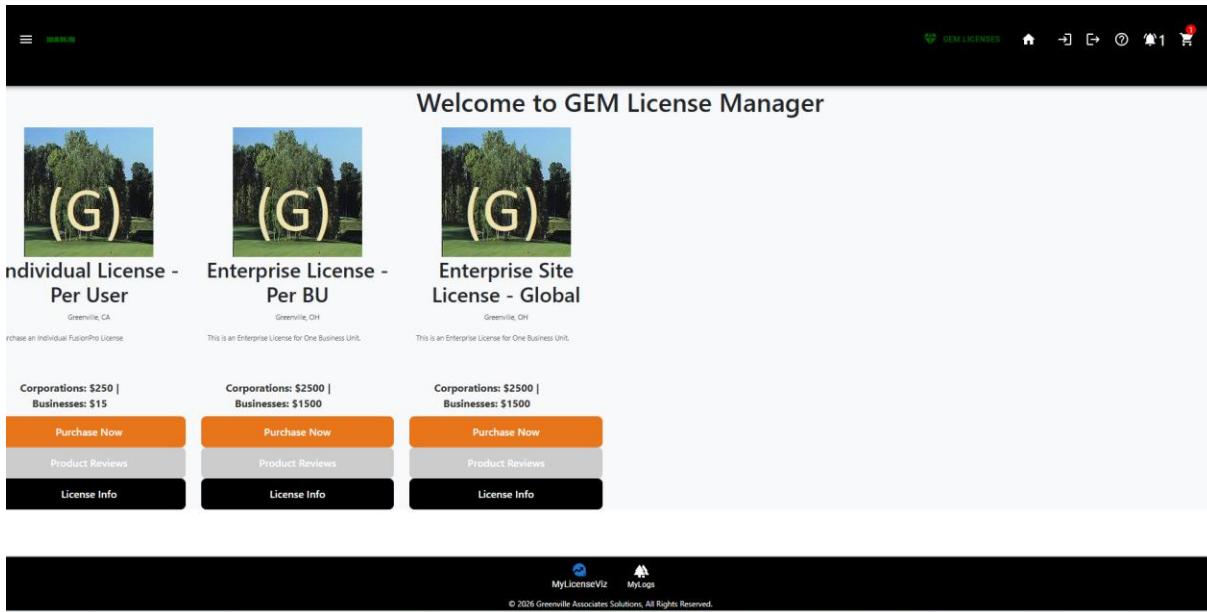
Enclosed is view of the home page.



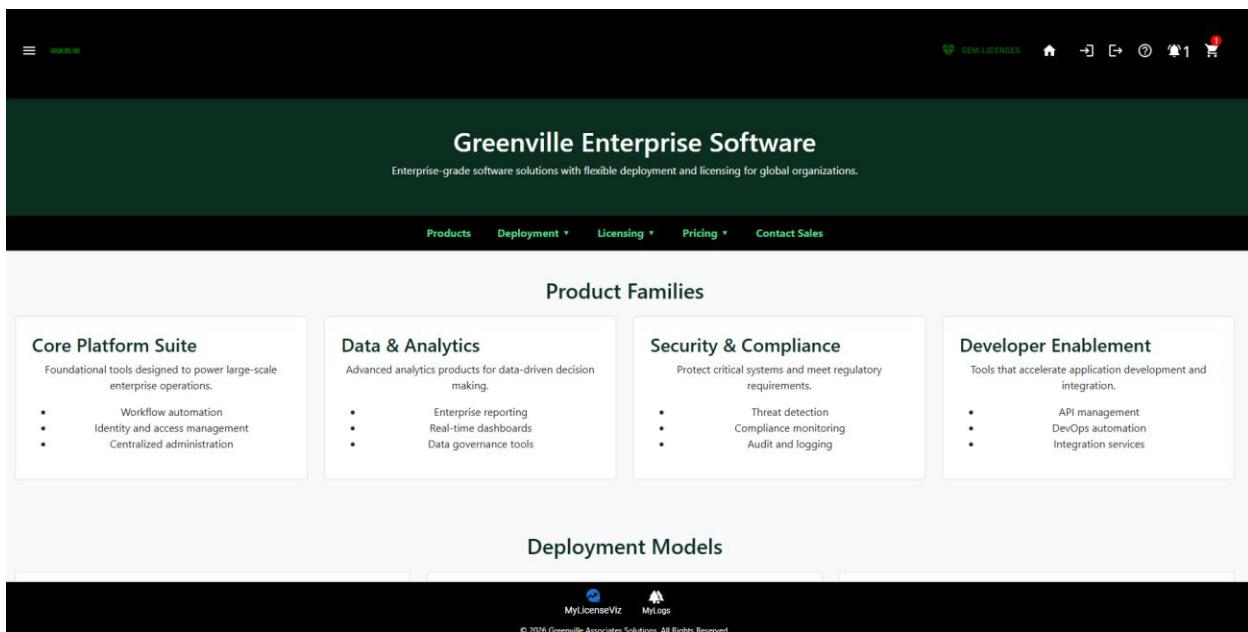
The screenshot shows the Greenville Shop homepage. The top features a large image of a classical building with the text '@Greenville Shop' and 'Browse and Purchase New Licenses For Your SaaS or SaaS Business'. Below the image are search filters for 'Location' (Search by location...), 'Minimum Rating' (Any Rating), and 'Apply Filters'. A 'Featured Products' section displays two items:  
**Individual License - Per User** (represented by a thumbnail of a landscape with a large letter 'G')  
**Enterprise License - Per BU** (represented by a thumbnail of a landscape with a large letter 'G')  
At the bottom of the page, there is a footer with links for 'MyLicenseViz' and 'MyLogi', and the text '© 2026 Greenville Associates Solutions, All Rights Reserved.'

## 4.3 Render Hosting of John Stritzinger's Enhanced Project Code – JSS Redesign

At the present time, we believe the CapGemeni Site Layout is more attractive than ours in either potential variant sent below. However we are moving to Normalizing the design between the sites as both are deficient now, as the Search Bar, and Featured Products pages are breaking the mobile environment on smaller screens.



Enclosed is the Enhanced Layout after Instructor Feedback...but it is not quite right either... completing the design to collective happiness will take a bit longer.



John S. Stritzinger  
Computer Science 242 – Wireframe/Page Layout Requirements  
Spring 2026

## 4.4 Live Database Queries Working:

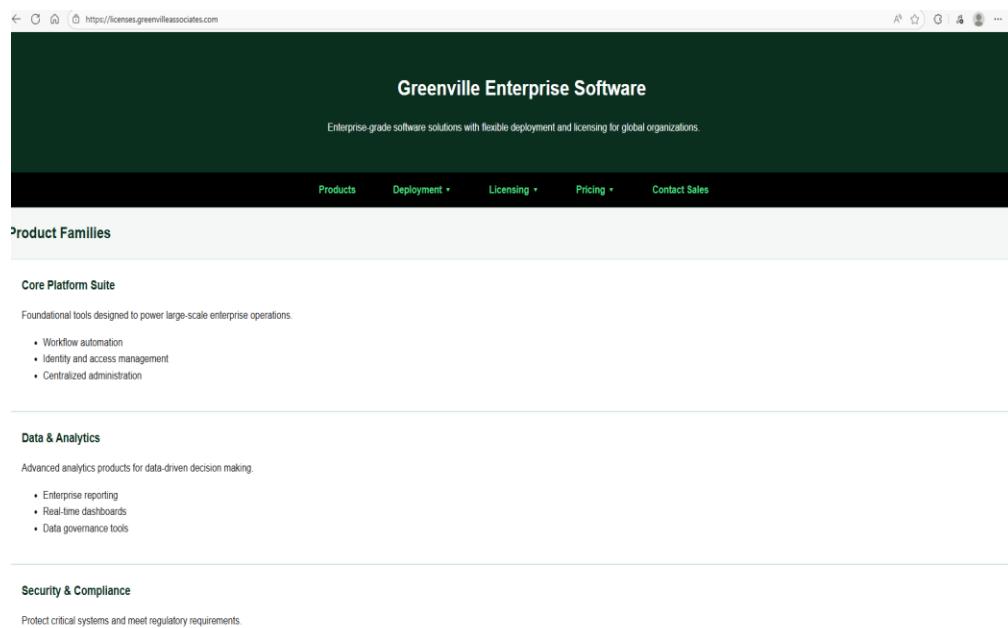
- a) Product Pricing
- b) License Types
- c) Sales Information
- d) User SSO
- e) User Notices
- f) User Logs
- g) User Notices
- h) User Help Tickets.

We have working Queries for most of the logical requirements for the program. All of this code is new code.

## 4.5 Advantages

The previous project with CapGemeni Delivered a New Cart, But with AI assistance we were able to build a new model which works up to the post. We think with some new guidance we can get the Post to work with AI in a few steps. But it requires more than 10 parameters in an AI session including working target JSONs. It's a non-trivial exercise with AI, and without AI we have two other working cart architectures in production we can port to this assignment. This includes the three tiered cart model we already presented.

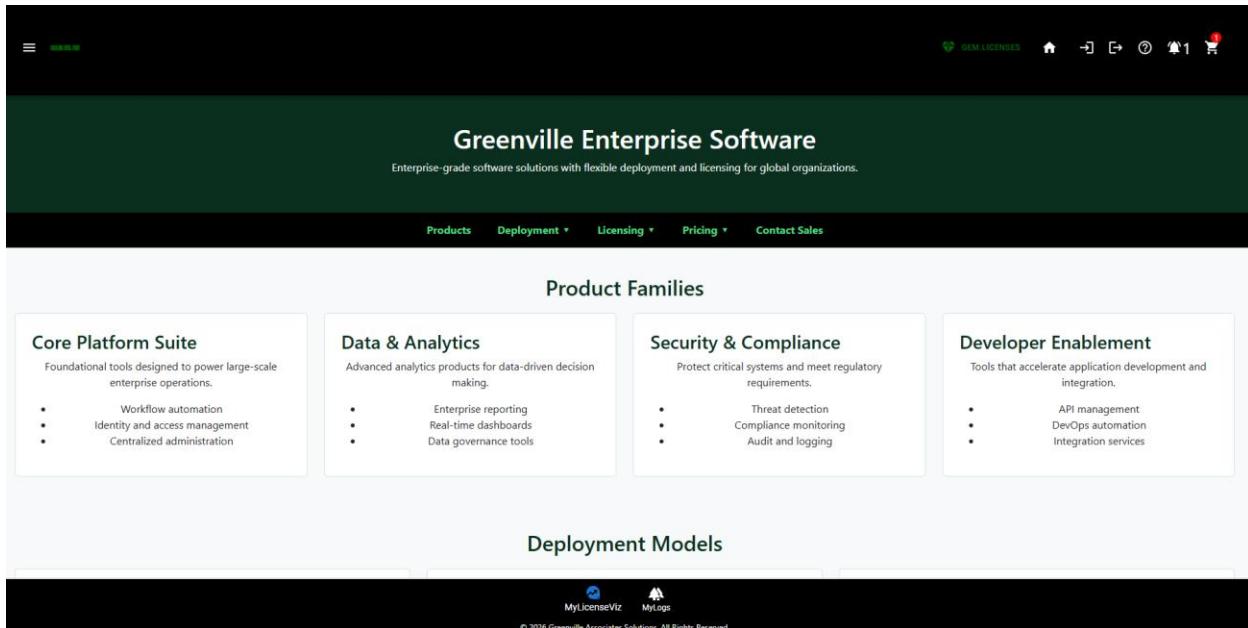
## 4.6 Target HTML/JS Site Up on [licenses.greenvilleassociates.com](https://licenses.greenvilleassociates.com)



The screenshot shows the homepage of the Greenville Enterprise Software website. The header features the company name "Greenville Enterprise Software" and a tagline "Enterprise-grade software solutions with flexible deployment and licensing for global organizations". A navigation bar at the top includes links for Products, Deployment, Licensing, Pricing, and Contact Sales. Below the header, a section titled "Product Families" lists three categories: "Core Platform Suite", "Data & Analytics", and "Security & Compliance". Each category has a brief description and a bulleted list of features.

- Core Platform Suite**  
Foundational tools designed to power large-scale enterprise operations.
  - Workflow automation
  - Identity and access management
  - Centralized administration
- Data & Analytics**  
Advanced analytics products for data-driven decision making.
  - Enterprise reporting
  - Real-time dashboards
  - Data governance tools
- Security & Compliance**  
Protect critical systems and meet regulatory requirements.

## 5.0 TARGET GREENVILLE LICENSE MANAGER WIREFRAMES – COMPLETE



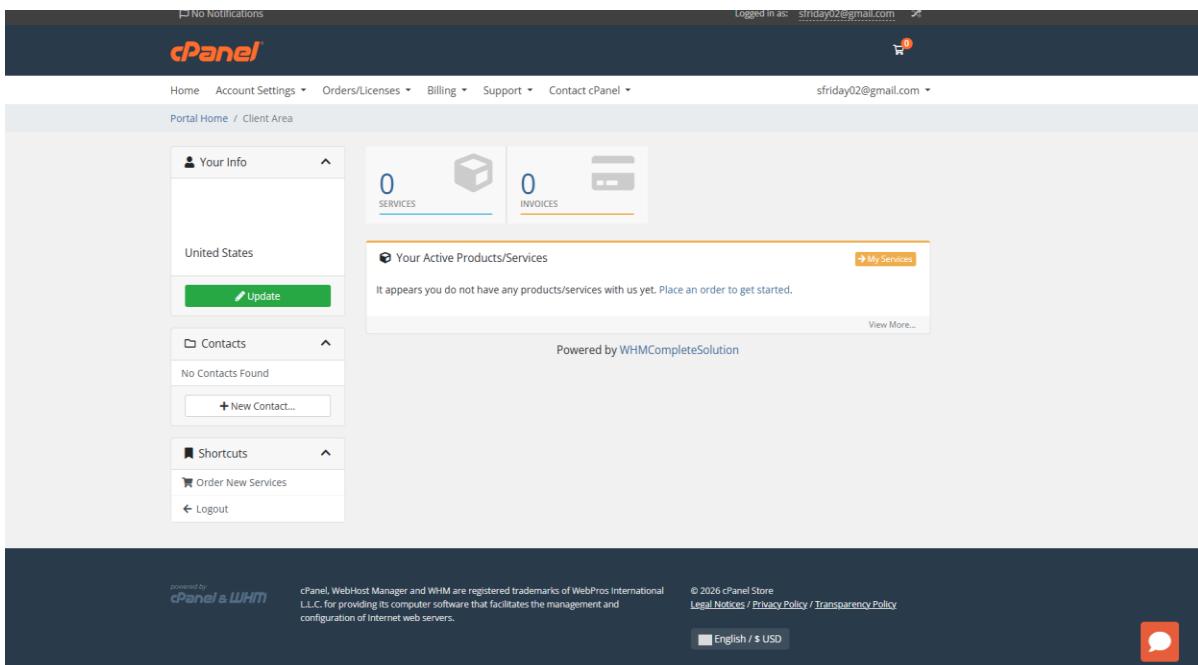
### 5.1 Executive Overview – Translating Scope of Work to Target Designs.

We have sought to build a Landing page for our LicenseManager which discusses what types of licenses we sell in the Greenville Enterprise Software(GES) Division of our company. The idea is after we logged in we would only see domain specific information

### 5.2 Competitive Analysis

Other competing products from 3CX VOIP tools, and cPanel which we think both do an excellent job handling Cloud Licenses and we look at their license managers as the gold standard which we are not trying to copy, but we think we can improve it substantially in the following ways:

- a) Our competitors tool only allows us to view our current licenses, add more and upgrade existing ones. This is significant code.
- b) Our competitors tools allows us to manage our business accounts, and purchase history. This is previous sales history. This is similar to restaurant companies like Sonic and Ihop restaurants which show your previous transactions online.



### 5.3 Significant Improvements over Competing Designs

The four competitors software tools we mentioned only show a single user or companies licenses. They have no view of enterprise licenses, or in a wholesale model the usage of agents. Furthermore these tools have no Administrative equivalents which show dashboard, and usage of platforms, although we think that our competitors have this information externally for investor purposes and for understanding the usage of their products in more detail.

We intend to provide aggregated information to Hosting Providers who have 5, 10, 100, or hundreds of containers of our software across multiple hosting regions, grids, and hybrid clouds.

We intend to use advanced logging and security tools to extend our SSO framework to a global platform and capability across our Global Managed Grid Infrastructure.

For the same reason they have no ability to provide any enterprise usage reports for their customers without professional services rollouts. We had intended to mirror these very successful companies deployments of license management which is superior even to Cisco or Microsoft in our opinion but she told us she thought it wasn't very good.

5.4 Structure of Final Workflow includes Up to 18 Pages All of Which are Designed and in near working form.

Major structures of our document include a TopBar, a BottomBar, and interior Mid-Page Navigation as seen in the diagram in 5.0 using Material-UI Tool bars. These engage the React-Router.

## 5.5 Menu Options DropDown & Pages Accessible via Top Nav Bar

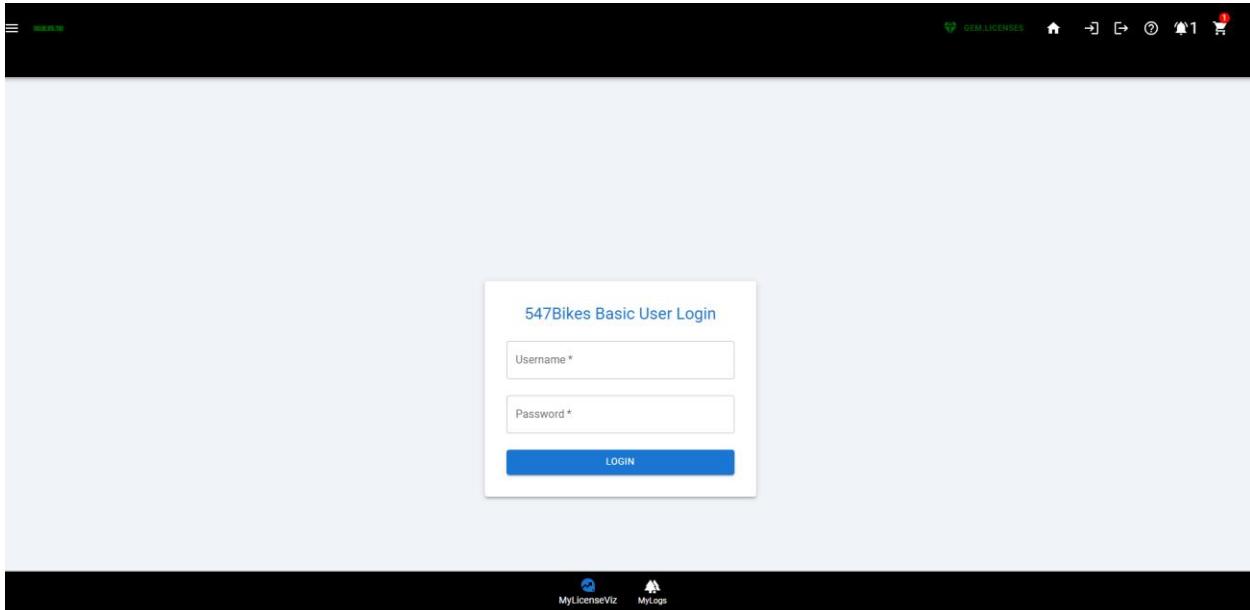
We intend to load the Sales Site as our Landing Page for the License Page with a General overview of the License Types available and where they can be used. Furthermore, the Login(1), Logout(2), UserNotices(3), UserHelp(4), And Cart(5) can be accessible from the Main Nav Bar and sized to work on all devices. The Menu Bar Supports additional information Welcome(SystemInfo)(6), Signup/Registration(7), All Licenses(8), Corporate Profile(9), ProductReviews(10), LicenseInformation(11), Home2(12), LicenseLogs(13).

The Bottom Nav Bar for End-Users is Operational. It shows Logs, and Current Licenses. On the Current Licenses Tool You can Buy More(14), or Upgrade Licenses(15) with default pricing level loaded. Then there is a cartreview(16), paymentreview(17), and cardmaintenance(18) page environment which is more than 3X the project requirements for the class, and with a much larger backoffice on Mongo DB our project is significantly more sophisticated.

These Screen #'s are included below for more information.



## 5.16 LOGIN(1)



## 5.16 LOGOUT(2)

Processed in the background with a spinner.

## 5.17 User Notices(3)

Email gateway failure	Account Balance. Please Review Soon.
<b>Error</b> <b>SMTP</b> User: user42 Email: user42@example.com 2/4/2026, 12:00:31 PM	<b>Account</b> <b>SMTP</b> User: user42 Email: user42@example.com 2/4/2026, 12:00:31 PM

## 5.17 User Help(4)

The screenshot shows a web application interface with a central 'Customer Contact' form overlay. The form contains fields for Phone, Email, Address 1, Address 2, City, State, Zip, Country, and Fax. A 'Submit' button is at the bottom. The background shows navigation icons like 'Submit Trouble Ticket' and 'Customer Support'.

Customer Contact

Phone

Email

Address 1

Address 2

City

State

Zip

Country

Fax

Submit

© 2026 Greenville Associates Solutions. All Rights Reserved.

## 5.18(5)

The screenshot shows a shopping cart summary page. It displays a single item: '10CAL - FusionShell Professional V6.01' with a price of '\$179.99'. Below this, it shows Subtotal '\$179.99', Tax (8%) '\$14.40', and Total '\$194.39'. Buttons for 'Clear Cart' and 'Proceed to Checkout' are at the bottom.

GEM.LicenseManager

Shopping Cart 1

10CAL - FusionShell Professional V6.01

SKU: FSFROV601-10CAL  
Vendor: Greenville Associates

VP Discount (10%)

\$199.99 **\$179.99**

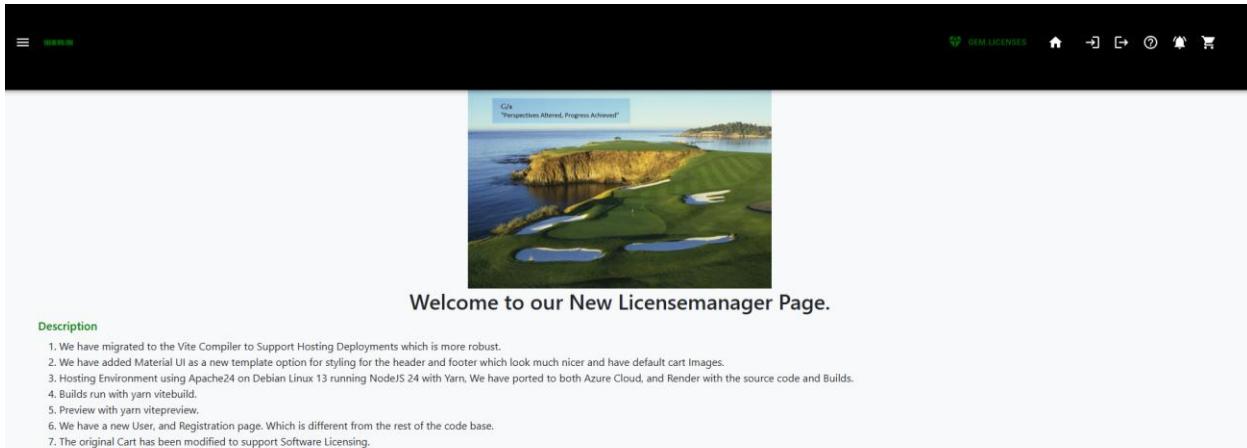
Subtotal: \$179.99

Tax (8%): \$14.40

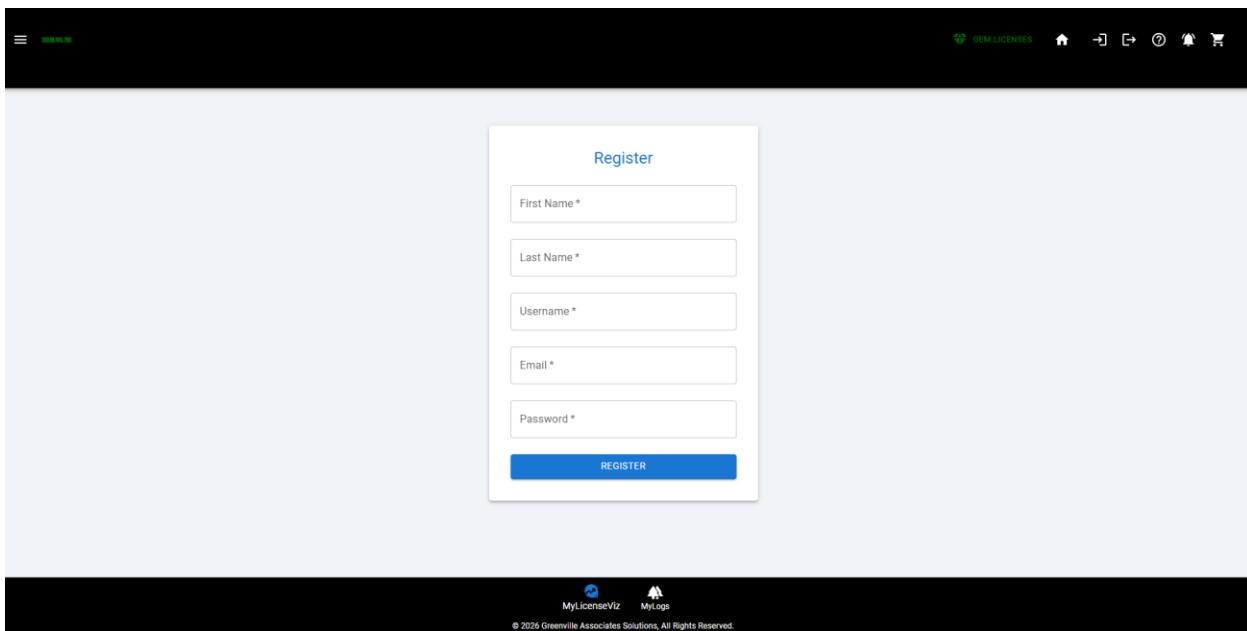
Total: **\$194.39**

Clear Cart Proceed to Checkout

## 5.19(6) Welcome(SystemInfo)(6)



## 5.20 Signup/Registration(7)



## 5.21 All Licenses(8) Legacy

The screenshot shows a grid of five license cards. Each card contains the following information:

License ID	Software	Version	Install Date	End Date	Customer
ABC1234	Open25	Open25	2025-01-01	2026-01-01	ClearviewSoftware211
ABC1234	Open25	Open25	2025-01-01	2026-01-01	COMP-001
LIC-001	Open25	1.0.0	2025-01-01	2026-01-01	COMP-001
LIC-001	Open25	1.0.0	2025-01-01	2026-01-01	COMP-001
LIC-331	Open25	1.1.0	2025-01-01	2026-01-01	VZF001

Below the cards are two buttons: "BUY MORE LICENSES" and "BUY ENTERPRISE LICENSE".

## Legacy Licenses – Figma Design(Improvement)

The screenshot shows a modernized license management interface. At the top, there are navigation icons for "MyLicenseViz" and "MyLogs". Below the header, the title "License Manager" is displayed with the subtitle "View and manage your software licenses".

On the right side of the header are three buttons: "+ Add License", "Upgrade License", and "Logout".

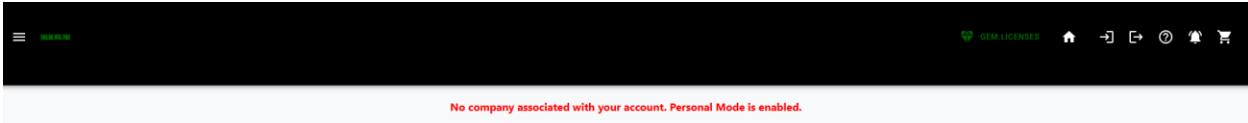
The main content area is titled "Active Licenses (5)" and displays a table of license details:

Software	Version	License Key	Seats	Expiry Date	Status
N/A	Open25	N/A	N/A	N/A	Active
N/A	Open25	N/A	N/A	N/A	Active
N/A	1.0.0	N/A	N/A	N/A	Active
N/A	1.0.0	N/A	N/A	N/A	Active
N/A	1.1.0	N/A	N/A	N/A	Active

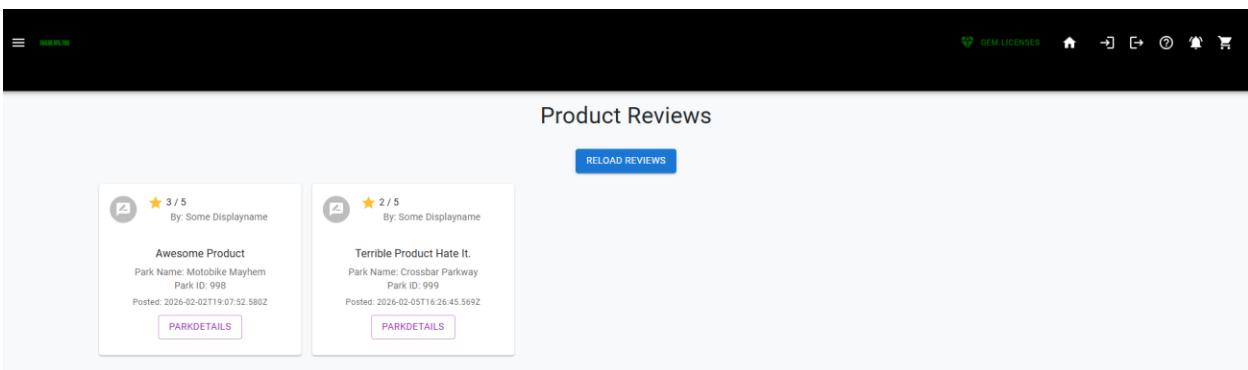
At the bottom of the interface are four navigation icons: "Dashboard", "Licenses", "Reports", and "Settings".

## 5.22 Corporate Profile(9)

This page is not in a good tangible form....Because I don't have a JOIN which works in the database so I can debug the result yet.... IE UserProfile to Corporate Account has to work.



## 5.23 ProductReviews(10)



## 5.24 LicenseInformation(11)

The screenshot shows the GEM License Manager interface. At the top, there are three categories: "Software On-Premise" (Installed and operated within your own data centers for maximum control.), "Software in the Cloud" (Fully managed cloud-based solutions with elastic scalability.), and "Hybrid Software" (Combine on-premise infrastructure with cloud-based services for flexible deployment.). Below these, a section titled "Enterprise Licensing" is shown, stating "Greenville Associates offers flexible enterprise licensing models designed to support organizations of all sizes and structures." It lists four models: "Software Based on Utility Hours" (Pay only for the compute or usage hours consumed by your organization.), "Named Users" (Licenses assigned to individual users for predictable cost and access control.), "Business Unit Enterprise Licenses" (Licensing tailored to specific divisions or departments within your organization.), and "Enterprise License Types" (Organization-wide licensing with unlimited usage across all business units.). Further down, a "License Store" section is shown with the sub-section "Enterprise Licenses". At the bottom of the page, there are navigation icons for "MyLicenseViz" and "MyLogs".

## 5.25 Home2(12) – Different View of Products – All Products in All conditions.

This is a Full List of Products while the home page is supposed to be just a subset of key products.

The screenshot shows the GEM License Manager interface with a "Welcome to GEM License Manager" message. It displays three product categories: "Individual License - Per User" (Greenville, CA), "Enterprise License - Per BU" (Greenville, OH), and "Enterprise Site License - Global" (Greenville, OH). Each category includes a thumbnail image of a landscape with a large letter "G", a brief description, price information (Corporations: \$250 | Businesses: \$15), and "Purchase Now" and "Product Reviews" buttons. At the bottom, there are "License Info" buttons. Navigation icons for "MyLicenseViz" and "MyLogs" are at the very bottom.

## 5.26 LicenseLogs(13).

User License Logs			
License ID	User	Status	Accessed
somelicenseid	john@glocation.info	active	1/18/2026, 12:00:00 AM
string	string	active	Invalid Date
string	string	active	Invalid Date
string	string	active	Invalid Date
string	string	active	Invalid Date
string	string	active	Invalid Date

 MyLicenseViz 
  MyLogs

© 2026 Greenville Associates Solutions. All Rights Reserved.

## 5.27 Buy More(14)

≡ ... GEM-LICENSES Home → ? Notification Cart

Search by location...

Any Rating

Apply Filters

### Featured Products



#### Individual License - Per User

④ Greenville, CA

★★★★★  
 5.00 (3 Reviews)

Purchase an Individual FusionPro License

[Purchase](#)

[License Options](#)

[Product Reviews](#)



#### Enterprise License - Per BU

④ Greenville, OH

★★★★★  
 3.00 (3 Reviews)

This is an Enterprise License for One Business Unit.

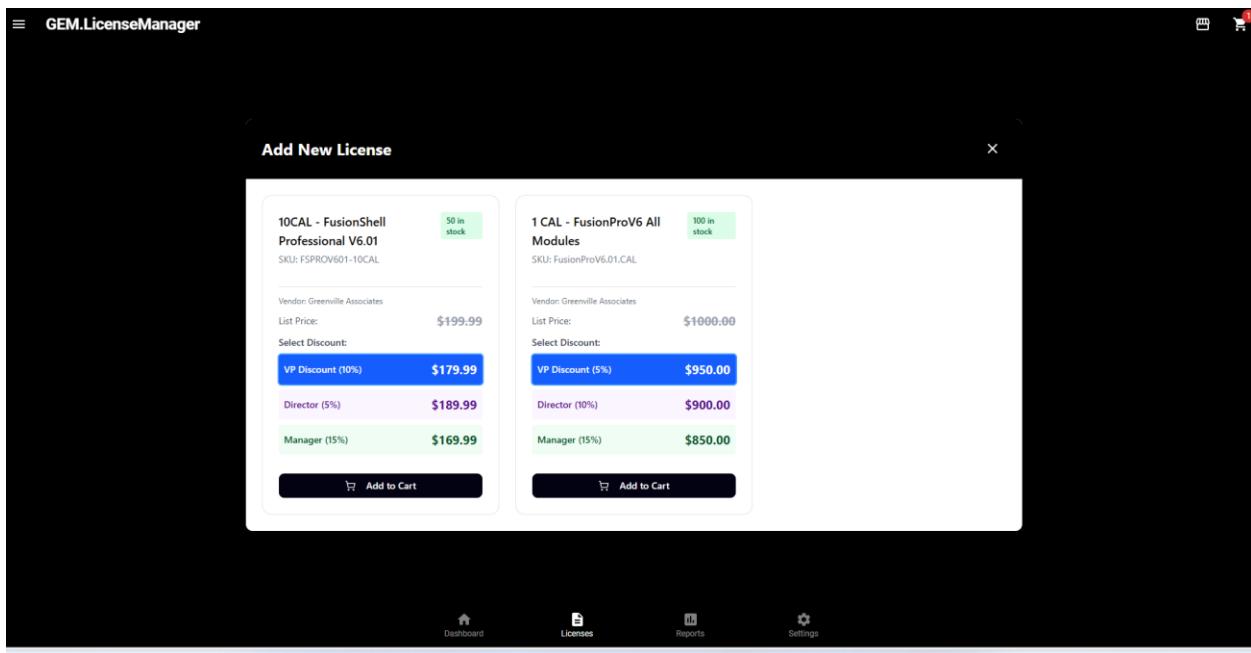
[Purchase](#)

[License Options](#)

[Product Reviews](#)

 MyLicenseViz 
  MyLogs

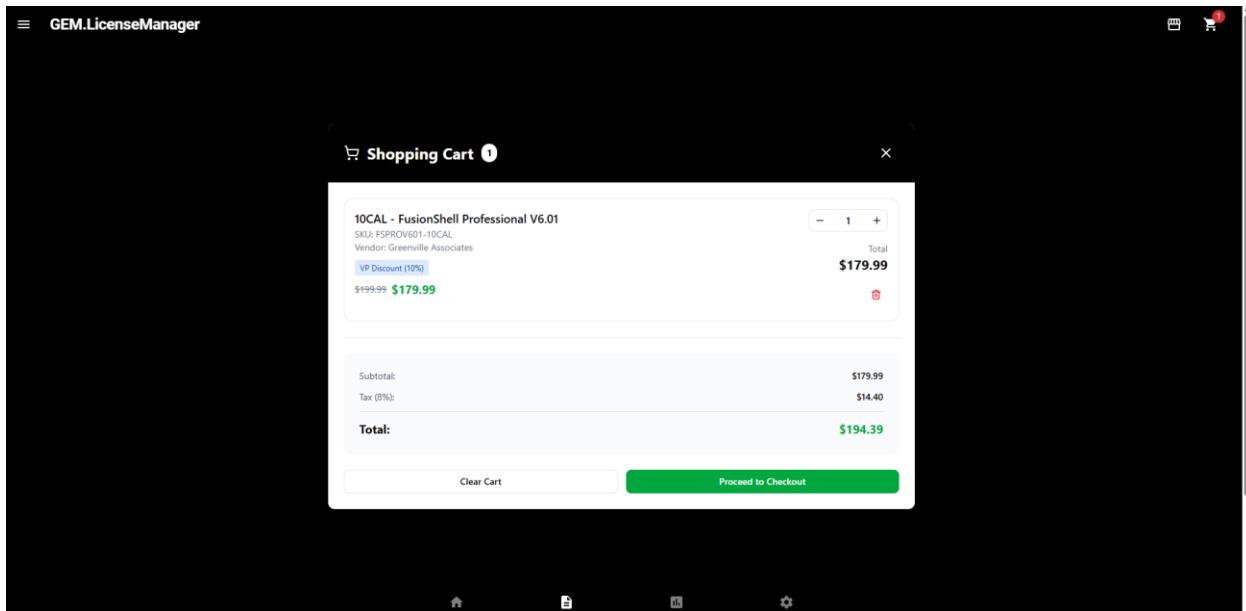
Inside the License Manager



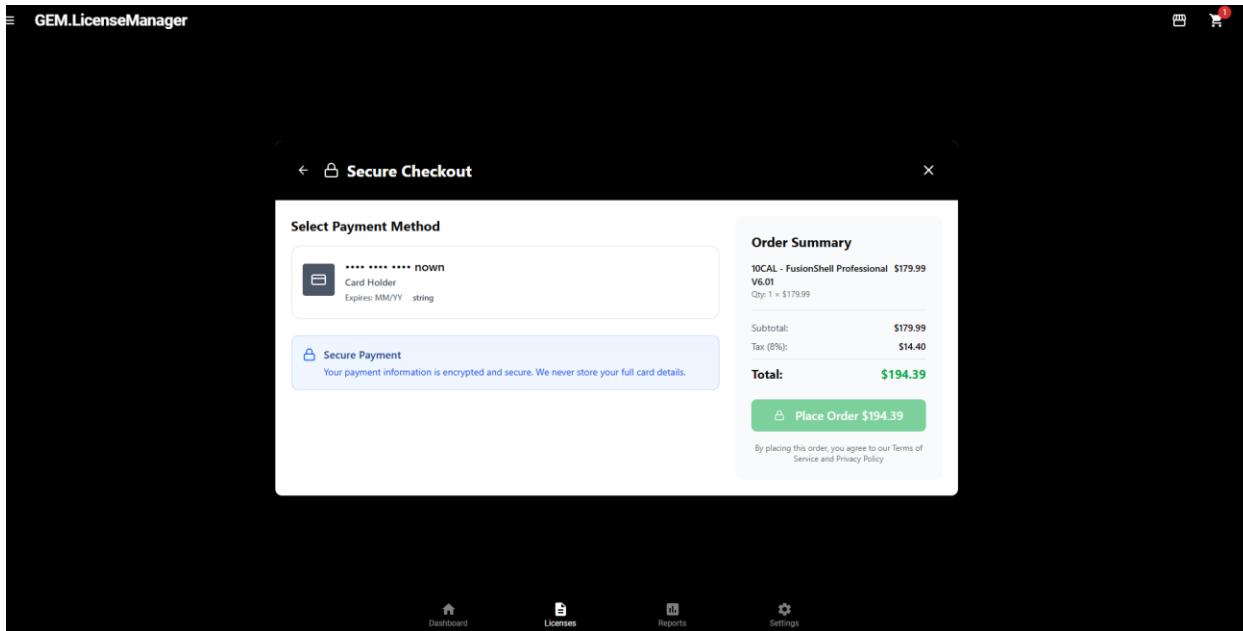
## 5.28 Upgrade Licenses(15)

Same as Above right now... will eventually filter for only products which can be upgraded.

## 5.29 Cartreview(16)



## 5.30 Payment Review(17)



## 5.31 Card Maintenance(18)

This is implemented in the CG Interface using custom Greenville Components. This button is not implemented in the Figma AI model as we ran out of credits.

## 6.0 RAPID PROTOTYPING

In this class we have been talking about using Figma to take a Wireframe, and build out code via a Publish Method. We agree that this works, however we can do the same thing using our own code which we pull from a combination of sources which isn't automated by Figma but is automated via production Rapid Prototyping which usually identifies most of the challenges needed in the backoffice.

### 6.1 Building the API first

When you build the API first in rapid prototyping a JSON is produced from your handiwork as previously demonstrated. Most AI tools can build entire screens in .ASPX, DotNet, C++, HTML/JS, or JAVA from this json alone. Building Wireframes before databases therefore is of questionable value except for learning programming languages.

### 6.2 Rendering From JSON Before Wireframing using Rapid Development is Advised

When you render pages from rapid development you take the time to look at flows of pages and how they are accessed. You identify tables and data you need to complete your screen which then usually adds layers to setup fields, and data modeling excercises. We also get to the point in some rendering sections that specific pages look terrible... etc and are going to need wireframes and better art production.

Claude, Grok, and Microsoft AI tools are greater than 95% accurate now with building simple web pages, especially using the predominate frameworks like Bootstrap.

6.3 Licenses.GreenvilleAssociates.com completely Rendered story board in itself.

## 7.0 STAGE WIREFRAME COMPLETE COMMENTS

This project as layed out is many orders of magnitude bigger than all existing commercial products in the same regard (See 3cx, cPanel who are Internet Leaders in this space) and is as large or equivalent to the project just completed in 547 as a Graduate student with teams as large as five.

Using Figma Design versus building pages is many times more complicated and slower than just building individual pages in your AI tool of choice. I am not sure it adds value. When you wireframe in Visual Studio you can make the screens work by dragging grid database objects on them.