

EXPORT_WORK_SUMMARY_2025-12-17.txt

SEAL Site – Consolidated Export
Date: 2025-12-17

Purpose

- Single-file reference of all work performed in the `V1` workspace to review later.

High-level summary

- Parsed live site and merged authoritative SEO/JSON-LD into `Index.html` (backup at `Index.original.html`).
- Built component library (`components.html` + `css/components.css`) and started replacing header/footer/hero a
- Downloaded many Wix/Gamma CDN images into `images/` and created a Canva originals folder at `images/canva/ori
- Created `scripts/webopt_canva.py` to produce JPEG + WebP optimized outputs from Canva originals; script exists
- Created pickled scaffold page `scaffold-pages/pickleball-passport.html` with responsive <picture> markup and (

Files created or modified (not exhaustive, key items)

- Index.merged.html (merge candidate)
- Index.original.html (backup)
- Index.html (merged authoritative SEO + content)
- components.html (component snippets)
- css/components.css (component styles)
- scripts/webopt_canva.py (Pillow-based image optimizer)
- images/ (many live assets downloaded from Wix/Gamma)
- images/canva/originals/ (Canva source files)
- images/canva/alt-texts.csv/alt-texts.csv (CSV with alt text for hero + pickleball images)
- scaffold-pages/pickleball-passport.html (patched with Gamma copy + responsive images)
- about.original.html (backup)
- about.html (patched to include canonical header/footer and link to css/components.css)
- contact.original.html (backup created just now)

Where the export lives

- File path: V1/EXPORT_WORK_SUMMARY_2025-12-17.txt (this file)

How to produce the web-optimized images (if you want to run locally)

- Requirements: Python 3.8+, pip, Pillow.
- From the `V1` folder, run:

PowerShell

```
```powershell
python -m pip install --user --upgrade pip
python -m pip install --user pillow
run the optimizer script (adjust path if needed)
python .\scripts\webopt_canva.py
```
```

Notes & expectations from the script

- Inputs: files under `images/canva/originals/` (script matches by substring to map filenames).
- Outputs: `images/canva/hero-home.jpg`, `hero-home.webp`, `hero-pickleball.jpg`, `hero-pickleball.webp`, `pick
- Quality: JPEG ~75, WebP created alongside JPEG.
- If any CDN-original assets returned AccessDenied, manually download via browser and place in `images/`.

Next recommended steps

1. Run `scripts/webopt_canva.py` to generate optimized images and verify they appear in `images/canva/`.
2. Continue replacing header/footer/hero on the remaining pages (create `.original.html` backups before each ov
3. Run accessibility and SEO checks (axe / Lighthouse) and address findings.

Contact points and authoritative data included

- The merged `Index.html` contains JSON-LD with Organization/localBusiness contact details (phone +1 (813) 951-

If you'd like

- I can (A) run the optimizer now and update pages to point to the generated images, or (B) produce a smaller P

End of export.

EXPORT_FULL_2025-12-17.txt

SEAL – Full Day Export
Date: 2025-12-17

Overview

This file consolidates all work performed earlier today in the `V1` workspace so you can review it tomorrow.

Key outcomes

- Merged authoritative SEO / JSON-LD into `Index.html` (backup at `Index.original.html`).
- Built component snippets and styles: `components.html` + `css/components.css`.
- Began replacing header/footer/hero across pages; created `.original.html` backups for edited pages (e.g., `about.original.html`).
- Downloaded many live Wix/Gamma CDN images into `images/` and collected Canva source exports at `images/canva/`.
- Created `scripts/webopt_canva.py` (Pillow-based) and a debug helper `scripts/webopt_canva_debug.py` to produce exports.
- Created scaffold page `scaffold-pages/pickleball-passport.html` with responsive `` markup, Calendly embed, and a hero image.
- Created an alt-text CSV at `images/canva/alt-texts.csv/alt-texts.csv` for the hero + pickleball images.
- Export files created today:
 - `V1/EXPORT_WORK_SUMMARY_2025-12-17.txt` (short summary)
 - `V1/EXPORT_FULL_2025-12-17.txt` (this file)

Files changed/created (key list)

- Index.html (overwritten with merged authoritative headers/JSON-LD)
- Index.original.html (backup)
- components.html (component snippets)
- css/components.css (component styles)
- about.html (patched to include canonical header/footer; backup: `about.original.html`)
- about.original.html
- contact.original.html (backup; small header snippet inserted)
- scaffold-pages/pickleball-passport.html
- images/ (many downloaded CDN images)
- images/canva/originals/ (Canva source files; see below file listing)
- images/canva/alt-texts.csv/alt-texts.csv
- scripts/webopt_canva.py (image optimizer; added fallback copy behavior)
- scripts/webopt_canva_debug.py (diagnostic script)

Canva originals (representative listing)

- Home Page cover image.png
- phuket beach.jpg
- phuket beach2.jpg
- pickleball 1.jpg
- pickleball 2.jpg
- pickleball 4.jpg
- mediterranean-1.jpg
- plus many other AVIF/WEBP/JPEG files under `images/canva/originals/`.

Script: `scripts/webopt_canva.py` (summary)

- Purpose: match source originals by substring, resize to target sizes, write `.jpg` and `.webp` outputs into `images/`.
- Mappings (source substring -> output base -> size):
 - "Home Page cover image" -> `hero-home` (1600x900)
 - "phuket beach2" -> `hero-pickleball` (1600x900)
 - "pickleball 1" -> `pickleball-1` (1200x800)
 - "pickleball 2" -> `pickleball-2` (1200x800)
 - "pickleball 4" -> `pickleball-3` (1200x800)
- Behavior: uses Pillow if available; if Pillow is missing or an error occurs, fallback copies the source file to `images/`.

How to run the optimizer locally

Requirements: Python 3.8+ and `pip`.

PowerShell commands (from `V1` folder):

```
```powershell
python -m pip install --user --upgrade pip
python -m pip install --user pillow
python .\scripts\webopt_canva.py
```
```

If you prefer a dry-run that prints found sources and attempts one test save, run:

```
```powershell
python .\scripts\webopt_canva_debug.py
```
```

Notes about earlier run

- I attempted to run `webopt_canva.py` earlier; the script executed but initially produced no outputs because the Pillow package was not installed.
- A few CDN image URLs previously returned AccessDenied – those require manual browser download into `images/`.

Backups

- Before any page overwrite I saved a `.original.html` backup in the same folder (e.g., `Index.original.html`, `about.original.html`).
- Next recommended steps

1. Run `scripts/webopt_canva.py` to generate optimized images into `images/canva/`.
2. Verify generated files exist and have expected dimensions; adjust mapping or sizes if necessary.
3. Update pages to reference the optimized local images (create `.original.html` backups before editing). Pages
4. Run accessibility/SEO checks (Lighthouse/axe) and fix issues.

If you want me to proceed now

- I can (A) run the optimizer and then update the site files to reference the new images, or (B) generate a PDF

End of export