**\*\*Phase 1 & Phase 2 Statement of Work (SOW)\*\***

**\*\*Ellis Global Care - International Caregiver Booking Platform \*\***

---

*Please note that these are minimum requirements and not meant to be a constraints, I am always open to suggestions on modifications and better ways to achieve the goal.*

**Phase 1 – MVP Scope (2 Weeks)**

**Objective:** To create a functional (and mobile optimized) website with registration system to begin caregiver recruitment and early client registration before full launch.

---

**Features & Deliverables:**

- \*\*Landing Page\*\* with value proposition, how it works, FAQs, etc.

- \*\*Client Sign-Up\*\* form to register account and then create profile for care recipient

- \*\*Caregiver Application Pages\*\* Job description page, then actual application page with form and document upload functionality, video interview page.

- \*\*Admin Dashboard Access\*\* to review caregiver/client submissions

---

**### User Journey – Phase I**

**#### Client**

1. Visits the website landing page
2. Learns about the business and services
3. Clicks “Book a Caregiver” button
4. Registers with email/password (or plugin options like google account, fbk account, etc)
5. Creates care recipient profile
6. Creates/selects required schedule/dates for the service.
7. Confirms booking details and then proceeds to payment page
8. There will be no actual payment page yet at this MVP stage. When customer clicks pay, they will be taken to a page that thanks them for their expected service, that this was a proof-of-concept, lets them know our caregivers are currently in training, lets them know our launch date, and that they will be contacted when we launch and promises them discount when we launch. – *I will provide you with the message for this.*
9. Awaits further updates from admin

**#### Caregiver**

1. Visits landing page
2. Learns about the business and services
3. Goes to careers page, learns about becoming a caregiver with us and application process.
4. Clicks apply and is taken to job application page to fill details and upload documents.
5. Submits application and gets confirmation of submission
6. Awaits email for rejection or success to move on to second stage of hiring (link to complete a recorded video interview)
7. Video interview page (link sent to applicants via email) will enable applicants enter their details and answer questions by recording a video answer to each question.

**#### Admin**

1. Logs in via Dashboard
2. View site visitor activity
3. Views submissions (caregivers and clients)
4. Verifies documents and tags profiles
5. Exports data as needed
6. Coordinates offline onboarding as applicable.
7. Etc.

--- **Functional Workflow Breakdown – Phase I**

**#### Feature: Landing Page & Call to Action**

- Visitor lands on the home page

- Scrolls to learn about services and benefits

- Clicks CTA

- Redirected to appropriate registration form

**#### Feature: Client Sign-Up Form**

- User clicks “Book a Caregiver” CTA

- Fills in registration form: Name, Email, Password etc or signs up with plugin (google account, fbk account, etc)

- Redirected to create recipient profile (Number of recipients (2 max), Name, Age, Care Needs, Location, contact person)

- Selects care schedule

- Confirms booking details

- proceeds to payment page

- Receives confirmation message: “We’ll notify you at launch”

- Data stored in database and visible in admin dashboard

**#### Feature: Caregiver Application Form**

- User clicks “Become a Caregiver” CTA

- Fills in form: Regular job application form containing such as Name, Phone, Email, Address, preferred work location, job experienceuploads CV, cover letter, etc

- Uploads optional certification files (nurse/chef)

- Data submitted to backend - Admin receives notification

**#### Feature: Admin Dashboard**

- Logs in admin

- Views two custom submission tables: Clients and Caregivers - Exports entries (CSV)

- Can tag, sort, or flag applications manually

- Sends follow-up or onboarding communication as needed

- can send communication to multiple applicants