This pretty detailed but the most important information is at the the top. It is flexible AND PLEASE DO NOT BE PUT OFF BY THE DETAIL it is actually pretty straighforward, i.e. Text,Image,Video, lots of choices with a mandatory default one that is required and a couple of selectors a for the first there pages leading to the main page.

"Storebuilder" and "Store Tow modes of the same thing.

THIS IS STOREBUILIDER FOR COLLECTING DATA ALMOST ALL FIELDS WILL BE FORMS WITH IDS. Most pages will be long to simplify

The structure is this few top level pages that link to many pages that are longish in nature. i.e. a broad tree. This simplifies things on my end

The site relies on POST requests

Structure

1 Login -> Legal -> "Cookie enabled options page that can be deselected on login on personal media or images -> Main Page
[
I. Every page on the following will have ** drop down menus, 1. A link to the main page. 2 4. A a link to a wiki page (You can help me develop that later)
3. A future link to a microphone icon for speaking (can be an icon for now. 5. An image search field whereby there is a way to some one an upload and image with a certain embedded submit button. (to be implemented) 6. a textbook for a search function (to be implemented)

So 4 basically dummy pages and 1 link THIS model can basically be Used on the STORE SITE TOO. It will be a similar template but instead of adding things we will be viewing things.

Every page in I is a separate page I is part of the future and should be there but doesn't have to be implemented now with links to some dummy static pages (these can be lumped in the same directory) i..e not implemented now.

]

**Every page will have a hidden field to provide the exact path to find the data
** Every page in the Main StoreBuilder page will have a stub path - the elements added

Main StoreBuilder Page
  — New Store page -> Goes back to Login and the process starts over

— Customer info page
— Business info page(if business)
—Catalog page (is recursive and calls itself to add product - so should look the same)

—Store Customization Page (this part is optional)
 (Optional stuff like customer levels
— A view current store page (which will basically go through the filesystems once the
 forms are submitted (once they are then it is done they are automatically saved) regurgitate all
the info
 ** A delete stuff page - this will be generated by myself using the names of the ids calls itself
eventually and is basically the same stuff as the main
page but instead of input forms there

I need in stages: Detailed description below. The developer(s)are free to use the freedom they
have to come up with solutions and their input is appreciated.

The first page will be a Log in Page. A customer can create an account using the following 4
piece of information (Most things come in fours and that is a pattern on this site). 1 of them must
be filled in for each but the other 3 are optional to simplify and reuse development. All of your
code should be as reusable as possible. No quick hacks and as little hard coding as possible

It is the first page the person visits the site when he/she is not logged in.

There will be a method to "recover login by phone or email"

Two factor will be enabled.

(It would help also to help get SSL certificates I will pay for this service.

The user emails of person (again only1) willl be passed around from page to page. The pages
will expire after a certain time then the person has to go back to the login page.

 i.e. 4 phone numbers, 4 contact information. 4 Images, 4 Videos, 4 Customer Levels (i.e. just
new here, want to learn even more, regular customer, special partner"
Each product in the catalog will have a s special price for each.

The catalog pages will be how every may items in The Stored all times

All this info will be saved in flat XML files. with the id's of the tags of the web page as
the levels in the XML files. I will take care of the parsing and the saving

The "number" field of the id always comes' last
Each page will have as it's <id> the store name i.e. store name version number

so it might look li
ie. in the following form. All have a date inserted at the end at it would be nice if you
guys to do it live using an "xhtml" request as can be found how tod do" i..e to get the
World Date. i.e.e UTC.
"StoreBuilder" "Storename", will be at the top of ever id and are separated by "-"'s
(dash) all converted to lowercase) numbers and symbols onlry

The form of the html documents would look like
<id storebuilder>
]
<./id>
and the "name fields will look like this : ""
<id ="
storebuilder-marcsstore-catalog-price-goodmember-type-textvbox`{-1date}>
>

Every name has to be hard coded initially (I know this will take time) but must include
all info necessary in lowercase letters and dashed "-" necessary to find the unique path and type
of data in the file system (My suggestion is to structure it according to the place its has in the
main document structure + a date appended)

<id
 I prefer that things be automatically generated as much as possible.
I will be able to provide you with the details.

You will have to hard code this initially you must do this to some sort of rule., but will

The result of each page will be stored in it's own separate directory according to the date
involved. Every <id> except for "StoreBuilder" and "Store name" have a date.

The first two pages will be stored in MySQL (Manga??) database (that is my problem)
The customer Login Page This is one of the only places where javascript is required to validate
information.

The second is a Legal page. (The page will disappear once the agreed
But a link will be available on every page.

These can even be served somewhere else on a separate server(i.e. networked database)

The 4th is a customer information page. They will be pretty long, to simplify the parsing and to
provide a place where a customer can enter all the information instead of figuring out menu's
and how to go back and forth and to navigate. T
i.e names

These can be line from above

Links to Please try to reuse only the data all change
Here they will provide the entering of data y…
4.a User's page (update info)
 b. A general info Page
4. b catalog page
4. c.

Payments page. Maybe different fro main use

A (total static) about us page

Akk payments will be embedded
[All payments will not store anything except for contact info

These can be all linked somehow. I want specifically avoid JavaScript except for links.

The Last is   a "customize security page"

Once that is entered the subsequent pages will be related to the store and will be linked to by a common "Storebuilder page" initially.

This page will be a page of promotions.

Common functions, which could be located in drop down menus on every page after the login and legal.

This has to be neat and tidy, so that if I wanted to create my own page I would be able to insert it easy. This is another place where cross-platform JavaScript must be available.

i.e. Chrome , Firefox, IE, Safari and edge (once you get this right it shouldn't be hard)

Note al lot of this code should be reusable for the Store too not just the Storebuilider

The function pages are

1.   Create a new Theme
   Under the following categories

a. Create new Product

The only identifier which will be the outer most <id> take will be the store name and there w
It will be stored a plain html files on a platform agnostic host. I have developed the "Store file
system" which is a nomenclature of how things are stored.Every name is unique and has a
date.It is ok for you to send me the html file and I will see how they look, but in the end I will
encapsulate them with print statements and change the extension so I can serve them so go
modular , no spaghetti code etc. When you send me the files therefore it can be in parts.

***All forms must specify the date as part of their id except the first two"

The Last for each page is is a "customer level" generation page of a "Customer Level page"
The custom will be able to specify that will list benefits and allow them to choose only one.

StoreBuilder and Store Project

This part deals with part 1. The store builder. It has a lot of input fields.The store will be pretty platform agnostic, that means all relevant information will be basically suing the host filesystem too read and write and execute. Again that is all my job, you job is to provide the html code to 1. Render the pages 2. Submit the information. The store it self will mostly display the data and I will provide a method for saving it and retrieving it. It should have a mobile friendly version of all of them (I hear there are converters, please use them to save time and money, if you cannot the mobile version will have to wait Your business should consider purchasing one it you don't have one). My method relies basically on having most of the information available for the store on a minimal of pages even if long. There are 3 special input pages. 1 The Login page (your suggestions will be helpful her, and I will be storing on a flat system

To simplify development things will generally come in optional 4's

The can be initially saved as static HTML files.
All pages will use the POST method of communicating data, (eventually plan is to use GET request top automatically log in) as the data will be large for each page.

This will be a static textfield that cannot be changed. If the terms are updated, then and only then will the new legal agreement be asked after login again (The file will have it's date changed on the system and Once the customer. The optional video page, where after login , or at
any other time, the customer will be able to optionally upload or update(1-4) Videos that they would like to feature on the main page of the store. This will be controlled by a cookie (Remember in the UK they have to ask to use cookies so A single ("I am ok with the site using cookies" ironically store in a cookie will be first presented to the user). So Once they submit a video they will have to goto a special "video" page on the top right of their site.
   Estimated time of completion, less than 3 months. Web store completion time of completion, less than 9 months to get a more polished product. The Project should preferably be done in HTML5, but HTML4 and css will be acceptable. I am looking for presentation. If the layout and the HTML code is provided, I will integrate it with my own dynamic code. Try to make it as modular and block oriented as possible (no spaghetti code) but rather modules.