Questions:

What does the above project represent in development/programming terms?

In terms of development terms, this represents building a full stack piece of software that would have a front end, a backend, a visual system to manage the backend. There will need to be services involving servers and hosting as live data needs to be streamed and potentially access to external APIs to manage payment processing.

2. What further information (if any) do you need before being able to design and estimate

the project.

While we know the basic information of what the client is expecting, there is still more information that is required. A good way to potentially start thinking about more information required is breaking it down in functional and non-functional requirements or user experience flow.

Starting off with potentially understanding similar websites or products that the client wants to replicate or reproduce, on a functional level.

Secondly understanding if there are any requirements of technical frameworks or non-functional requirements, as with many technical architectures and frameworks there are tradeoffs that need to be considered.

Any expected turnaround times would also be useful to understand, as this would require to potentially use premade templates/frameworks for certain aspects rather than building customized solutions.

3. Based entirely on gut feel, roughly how many hours do you estimate this project will

take?

There are certain assumptions that may need to be made to estimate a timeline. While the project seems straightforward, the first assumption would be that there would not be any unexpected delays in the process. Based on a standard team of 3-5 people, it may be 500 – 1000 hours for turnaround time, but this may vary extremely.

4. Propose the key work stages or blocks in this project

1. Requirements gathering and survey – this step would involve interviewing the client to understanding user stories and potential flows. This is a very important part as this will allow us to understand all the potential UX flows that need to be generated and the auxiliary features that may need to be developed. It will also provide an understanding of any frameworks or non-functional requirements that are required.
2. UX/UI/Frontend– Once the flows are understood, the next step would be to showcase all the UX/UI diagrams that will be featured in the final product. This will be useful to understand if all functional requirements are met.
3. Backend Development – Once the frontend and flows are approved, then building the backend to get functionality would be the next step. This involves building a server (or serverless) backend and databases and potentially integrate and build any middleware.
4. Auxiliary/API integration – for example logging in in with socials, or payment processing to provide full features of the final product.
5. Testing/ debugging – there will be bugs and testing that needs to be done to ensure the final product is fully functional.
6. Handoff/maintenance – once the project is handed off, there may be a transitionary period where the client may need to be trained in how to operate the backend/software, there will likely be future requests for features or maintenance of the website.

There probably are steps that may be missing, however generally I would break down the project in these key stages.