**Spiced Academy - Final Project Proposal - Abrar Faisal**

*Responsive Website for “GOLPO-Berlin” Catering Services*

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**Possible features:**

1. General (Guest customer ):

* Gallery of food items with description (read-only)
* Food cart for orders with payment (online transfer or PayPal)
* Report generation of order and payment (pdf)
* Email notification to owner on order placement and payment
* Email notification to customer about order, payment confirmation and pick-up/delivery information
* Contact form to communicate with owner as guest

1. Registered Customer (Frequent Foodie, FF):

* All general features from above
* Registration as frequent customer with profile - personal details with profile pic, all orders, discounts available, etc.
* Comment/feedback on food items in gallery (add/delete own)
* Chat with owner when online - in case owner offline - email notification to owner
* Request for sharing selected recipes
* Discount on orders e.g. once after 5/10 consecutive orders

1. Owner:

* All features from general and registered user
* Modify gallery of food items , description, recipes, etc.
* Delete comments/feedback of any customer (security)
* Organization of all orders, orders by individual customer/food item, monthly, etc.

**Layout:**

* Homepage/Welcome
* News/What’s new?
* About/Vision/Stories
* Menu + Order Food
* Food Cart/Basket
* Register as FF
* Recipes for FFs
* Login/Logout

**Owner wishes:**

1. Shopping cart and payment options
2. Multilingual - English and German
3. Color scheme, pictures and necessary information from owner

**Language and frameworks:**

* HTML, CSS, JavaScript (frontend/client)
* React with Redux (frontend/client) - single page layout
* Node.js, Express (backend/server)
* PostgreSQL for relational database
* AWS S3 and SES for online storage and email services
* Heroku for first online deployment
* socket.io for chat feature
* Others:
  + CSRF for malicious cookie requests
  + bcrypt for password protection