System Context Diagram Any user utilizing the Niibish Aki website **Favorites** Niibish Aki writes to [JSON] Description Menu [Container: MongoDB] ongoDB Collection that stor teas and food. [Container: MongoDB] MongoDB Collection that stores order information su as user ID, beverage (tea) food, and bill. **Container Diagram** User Accounts [Container: MongoDB] MongoDB Collection that stores user information, password, and address. **Next JS Frontend** Frontend System —utilizes backend functionality—utilizes backend functionality— Menu API Ordering API User API [located under /api/inventory] [located under /api/order] [located under /api/user] Provides all of the functionality ovides all of the functionali ovides all of the functiona for user management Makes PUT API calls Makes GET API to [JSON/HTTPS] calls to [JSON/HTTPS] Makes GET/PUT API calls to [JSON/HTTPS] Makes GET API calls to [JSON/HTTPS] **Favorites** [Container: MongoDB] [Container: MongoDB] [Container: MongoDB] [Container: MongoDB] MongoDB Collection that MongoDB Collection that Component Diagram teas and food. [Component: React Component] Layout page used for every page. Cart Page Menu Page Register Page Login Page User Page Home Page Component Static Pages [Component: React Page] [Component: React Component] [Component: React Page] Static pages that do not provide Includes the HTML to display a pecific functionality related to featured items section. --uses-----**Next JS Frontend** [primarily located under /pages productCard Component address Component [Component: React Component] Frontend System Includes the HTML to display a [Component: React Component] **Component Diagram** checkoutInfo Component product card. cludes the HTML and validation [Component: React Component] for address updating. Includes the HTML to display a form and validates user's contact information and tip. —uses cartitem Component bill Component Component [Component: React Component] userFavorites updateUser Component Includes the HTML to display one Includes the HTML to display the Includes the HTML to display as a Component Component item in the cart and cart's bill. modal and verifies item info. [Component: React Component] [Component: React Component] [Component: React Component] GET /order/{orderId} GET /order/user/{userId} Includes the HTML to display the PUT /order/{orderId} DELETE /order/{orderId} Includes the HTML, validation, POST /order Includes the HTML, validation, Includes the HTML, validation, Includes HTML for logout and submittal for account user's favorite items based on and submittal for user login and submittal for user update [Component: React API Route] [Component: React API Route] [Component: React API Route] [Component: React API Route] clears **▼** user context GET call that fetches the user's POST call that creates an POST call that updates an order. DELETE call that removes a in-progress order. in-progress order for a user. order history. userContext Component orderHistory Component [Component: React Component] Includes the HTML, validation, user context [Component: React Component] and submittal for user login/info Includes the HTML to display the Reads from and user's order history. writes to [JSON] writes to [JSON] Order API Reads from [JSON] **Next JS Frontend** [primarily located under /pages] Component Diagram Frontend System Orders [Container: MongoDB] MongoDB Collection that stores order information such as user ID, beverage (tea), food, and bill. Ordering API Menu API User API **Next JS Frontend** [located under /api/order] [located under /api/user] [located under /api/inventory] ovides all of the functionality rovides all of the functionality vides all of the functiona GET /menu/{itemId} /menu/featuredItems GET /menu/addIns for placing orders Frontend System for using menu items. for user management [Component: React API Route] [Component: React API Route] [Component: React API Route] [Component: React API Route] **Component Diagram** GET call that fetches all menu GET call that fetches menu item GET call that fetches all featured GET call that fetches all add-ins ----------Menu API reads from [JSON] GET POST /user/{userId} user/Favorites/{userId} user/Favorites/{userId} user/Favorites/{userId} [Component: React API Route] POST call that validates user GET call that fetches favorite POST call that updates favorite DELETE call that removes favorite GET call that fetches registered POST call that validates user information and saves to items for a user items for a user item for a user exists and updates it. exists and returns information Menu collection. [Container: MongoDB] Reads from and reads from and writes to [JSON] ongoDB Collection that store writes to [JSON] Reads from [JSON] teas and food. Reads from and writes to [JSON] Reads from [JSON] Reads from [JSON] User Accounts [Container: MongoDB] MongoDB Collection that stores user information, password, and address.