

Project name	Agro nursery online store
--------------	---------------------------

Project author		
№	Names of Participants	FN
1	Ivan Petrov	3MI0700035

1. Short project description (Business needs and system features)
<p>Project Description: Agro Nursery Online Store</p> <p>The Agro Nursery Online Store is designed to allow customers to purchase nursery products. This platform will enable users to browse a wide range of plants, seeds, and gardening supplies while providing a seamless shopping experience.</p> <p>Key Features:</p> <ul style="list-style-type: none"> ● User-Friendly Interface: Using <i>html/template</i>, <i>tailwindcss</i>, <i>Alpine.js</i> for the UI and interactivity. Customers can easily navigate through the product catalog, view detailed descriptions, and make purchases. ● Real-Time Customer Support: Integrated chat support using <i>gorilla/websockets</i> allows users to connect with customer service representatives for immediate assistance. ● Secure User Registration: Users can register, manage profile data, and track their orders. Using <i>gorilla/sessions</i> as an auth service on the server and DB ● Product Management: Administrators can add, update, and manage products within the store efficiently. As well as see real-time statistics with <i>e-charts</i> ● Dynamic Routing: The application will utilize (TO BE DECIDED) <i>gin // gorilla/mux // chi</i> for server-side routing, ensuring smooth navigation between product pages and main views

- **Robust Database Management:** Leveraging *sqlc* and *pgx* for PostgreSQL database interactions, the platform will ensure efficient data handling and security.

1

2. Main Use Cases / Scenarios		
Use case name	Brief Descriptions	Actors Involved
2.1. Browse Information	The <i>User</i> can browse the information views (Home, Products, About) in <i>the system</i> and can choose to order products	All users
2.2. Register	<p><i>Anonymous User</i> can register in the system by providing a valid e-mail address, first and last name, and choosing password. By default, all new registered users have <i>Customer</i> role.</p> <p><i>Administrator</i> can register new users by entering <i>User Data</i> and choosing a Role (<i>Customer</i>, <i>Support</i>, or <i>Administrator</i>).</p>	<i>Anonymous User</i> , <i>Administrator</i>
2.3. Change User Data	<p><i>Registered User</i> can view and edit their personal <i>User Data</i>.</p> <p><i>Support</i> can view <i>User Data</i> of all users</p> <p><i>Administrator</i> can view and edit <i>User Data</i> of all <i>Users</i> and assign them <i>Roles</i>: <i>Customer</i>, <i>Support</i>, or <i>Administrator</i>.</p>	<i>Registered User</i> , <i>Support</i> , <i>Administrator</i>
2.4. Manage Users	<p><i>Administrator</i> can browse and filter users based on different criteria: first and last name, email, Role.</p> <p><i>Administrator</i> can choose a <i>User</i> to manage, and can manage the chosen <i>User</i> - edit (using <i>Change User Data UC</i>) or delete.</p> <p><i>Administrator</i> can create a new user using <i>Register UC</i>.</p>	<i>Administrator</i>
2.5. Manage Products	<i>Administrator</i> can browse and filter <i>Products</i> based on different criteria: name, category, creation date, etc.	<i>Administrator</i>

¹ Can add like Manage subscriptions use case for automatic orders of materials

	<p><i>Administrator</i> can view statistics about products</p> <p><i>Administrator</i> can create a new Product.</p> <p><i>Administrator</i> can choose a <i>Product</i> to edit and delete if necessary</p>	
2.6. Manage Orders	<p><i>User</i> can browse their orders, add new <i>Order</i> using <i>Cart View</i></p> <p><i>Administrator, Support</i> can browse <i>Orders</i> of all <i>Users</i>, update and delete them.</p>	<i>User, Support, Administrator</i>
2.7. Order a product	<i>User</i> can add/remove items from their <i>Cart</i> and order them by processing their cart order.	<i>All users</i>
2.8. Contact Customer Service	<p><i>Registered users</i> may contact the support of the site for issues they came across</p> <p><i>Support and Administrators</i> can answer users</p>	Registered users

3. Main Views (Frontend)		
	Brief Descriptions	URI
3.1. Home	Show special offers and products catalog	/
3.2. Products category	Shows all subcategories of the product category	/products/{category}/
3.3. Products subcategory	Presents all store products from the subcategory	/products/{category}/{subcategory}
3.4. Product	View with full information for a product and option to order it	/product/{productId}
3.5. Customer service	Dialog chat window / Page view to contact customer support	/chat/{chatId}

3.6. User Registration	Presents a view allowing the <i>Anonymous Users</i> to register in the <i>System</i>	<i>/register</i>
3.7. Login	Presents a view allowing the users to login.	<i>/login</i>
3.8. User Data	Provides ability to view and edit personal <i>User Data</i>	<i>/profile</i>
3.9. Users	Presents ability to manage (CRUD) Users and their User Data (available for <i>Administrators</i> only, as described in UCs).	<i>/users</i>
3.10. About	Presents information about the the project and his owner.	<i>/about</i>

4. API Resources (Backend) Not sure it's needed as it's server-side only app		
View name	Brief Descriptions	URI
4.1. Users	GET <i>User Data</i> for all users, and POST new <i>User Data</i> (Id is auto-filled by <i>the server</i> and modified entity is returned as result from POST request). Available only for <i>Administrators</i> .	<i>/api/users</i>
4.2. User	GET, PUT, DELETE <i>User Data</i> for <i>User</i> with specified <i>userId</i> , according to restrictions described in UCs.	<i>/api/users/{userId}</i>
4.3. Login	POST <i>User Credentials</i> (e-mail address and password) and receive a valid <i>Session Id</i>	<i>/api/login</i>
4.4. Logout	POST a logout request for ending the active session, and invalidating the issued <i>Session</i> .	<i>/api/logout</i>
4.5. Products	GET <i>Product Data</i> for all products, and POST new <i>Product Data</i> (Id is auto-filled by <i>the server</i> and modified entity is returned as result from POST request). Available only for <i>Administrators</i> .	<i>/api/products</i>
4.6. Product	GET, PUT, DELETE <i>Product Data</i> for <i>Product</i> with specified <i>productId</i> , according to restrictions described in UCs.	<i>/api/products/{productId}</i>
4.7. Chats	GET <i>Chat Data</i> for all chats, and POST new <i>Chat Data</i> (Id is auto-filled by <i>the server</i> and modified entity is returned as result from POST request). Available only for <i>Administrators</i> .	<i>/api/chats</i>
4.8. Chat	GET, PUT, DELETE Chat Data <i>Chat</i> with specified <i>chatId</i> .	<i>/api/chats/{chatId}</i>

4.9. Messages	GET <i>Message data</i> for all messages for specified <i>chatId</i> Chat, and POST new <i>Message data</i> (Id is auto-filled by <i>the server</i> and modified entity is returned as result from POST request). Available only for <i>Administrators</i> .	<i>/api/chats/{chatId}/messages</i>
4.10. Test Result	GET, PUT, DELETE <i>Message data</i> for <i>Message</i> with specified <i>messageId</i> in specified <i>chatId</i> Chat	<i>/api/chats/{chatId}/messages/{messageId}</i>
4.11. Orders	GET <i>Order data</i> for all orders, and POST new <i>Order data</i> (Id is auto-filled by <i>the server</i> and modified entity is returned as result from POST request). Available only for <i>Administrators</i> .	<i>/api/orders</i>
4.12. Order	GET, PUT, DELETE <i>Order Data</i> for <i>Order</i> with specified <i>orderId</i> , according to restrictions described in UCs.	<i>/api/orders/{orderId}</i>