Allegato K - Requisiti informatici delle piattaforme di fornitura e vendita da utilizzare.

The technical specifications will be evaluated on the compliance to the following requirements.

Table of Contents

1. General Requirements

1.1 Operating System Compatibility

The application must be fully operational on Windows (version 10 and above), macOS (version 10.14 and above), and Linux (Ubuntu 18.04 and above).

1.2 Web Browser Compatibility

Full functionality must be retained across leading web browsers: Google Chrome (version 90 and above), Mozilla Firefox (version 85 and above), Apple Safari (version 14 and above), and Microsoft Edge (version 90 and above).

1.3 User Interface

The user interface should be intuitive and easy to navigate, with a focus on user experience.

1.4 Mobile Responsiveness

The application must be mobile-responsive, allowing users to easily navigate and perform key tasks on various mobile devices including tablets and smartphones.

1.5 Performance and Speed

The application should have fast load times (less than 2 seconds for primary functionality) and should be able to support multiple users simultaneously without performance degradation.

1.6 Scalability

The architecture of the application should be scalable to accommodate future expansion in terms of users, product listings, and additional features.

1.7 Search and Filter Capabilities

Users should be able to easily search for and filter products based on various parameters such as category, price range, and availability.

1.8 Documentation

Comprehensive documentation, including a user guide and FAQs, should be readily available to aid in the usage and troubleshooting of the application.

2. Security requirements

The following is a non-exhaustive list of security requirements that need to be adopted when building, deploying, running, and maintaining a website, or Internet facing application.

2.1. Secure domain ecosystems

a) Deploy the website (application or service) in a trusted infrastructure, operating with maximum flexibility and business agility without compromising security but ensuring a resilient and flexible infrastructure, supporting a Zero Trust approach.
b) The whole technology stack used to build a website (as well as an application or service) needs to be secured;
c) The Operating Systems used must be from an official distribution or vendor that are able to timely deliver patches, bug fixing, remediation for critical vulnerabilities and updates;
d) Enable automatic updates whenever possible;
e) Software must be provided from a commercial or trusted vendor releasing updates and fixes;
f) The website (application and service) must be deployed, updated and maintained during the whole lifetime;
g) While selecting the Domain name, Check that it was not blacklisted (when in use by a previous owner) to avoid reachability issues;

2.2. Secure user accounts

a) Enforce Single Sign-on with EUI Identity repository when the access to website, application and service should be granted to EUI users;
b) Implement the principle of least privilege and disable unnecessary accounts and privileges in website, application and service;
c) Enforce MFA on internet-accessible accounts;
d) Enforce the security of identities and password for local accounts on websites, applications and services and never share (or transfer) credentials with other systems;
e) The privileged accounts (i.e., administrator) of websites, applications and services require (i) enabling MFA; (ii) enforcing secure passwords (minimum length >8 digits, complexity, age and history...); (iii) performing administrators' account review at least once a year;
f) For privileged accounts, initial (or default) passwords must be changed on the first access, especially those provided in the SaaS cloud environment.

2.3. Review and remediate critical and high vulnerabilities

a) Patch critical and high vulnerabilities within 15 and 30 days, respectively, on internet-accessible systems;
b) Request (or perform) scans for detecting configuration and software vulnerabilities before going live and few times a year.

2.4. Secure data in transit

a) In order to ensure that encrypted communication takes place between browser and website, disable Hypertext Transfer Protocol (HTTP) and enforce Hypertext Transfer Protocol Secure (HTTPS);
b) Apply valid digital certificate issued by a Trusted Authority
c) Accurately manage the digital certificate lifecycle in order it should not expire inadvertently and its data are stored in a safe and accessible place for administrators;
d) Disable weak and deprecated cyphers.

2.5. Secure data at rest

a) Depending on the risk, evaluate to encrypt data repository (databases, files, …);
b) For enhancing database security, check OWASP website.

2.6. Backup data

a) Devise a 3-2-1 backup strategy, having 3 copies of your data (your production data and 2 backup copies) on two different media (disk and cloud) with one copy off-site for disaster recovery;
b) Use backup solution that automatically performs daily backup critical data and system configurations of website, application and service;
c) Store backups in a safe and physically distinct (remote) environment;
d) Periodically, at least once a year, test disaster recovery scenarios.

2.7. Security controls and capabilities

The following security controls are required to protect websites from most common threats and to allow performing incident analysis:

a) Enable Anti-Malware and Advanced Threat Protection to automate response to the most common cyber-threats (i.e. malware, crypto-lockers);
b) Use Web Application firewalls to reduce the likelihood of severe impact from cyber-attacks due to the exploit of unused ports, misconfigurations, known software vulnerabilities;
c) Providing load balancing and resilience against peaks of traffic and leveraging content delivery networks (CDNs).
d) Enable logging collecting data traffic, successful login/logout activities, failed login attempts, security events and changes in configuration files; in the case of a data breach, these data are required to perform incident analysis;
e) Retain data logs for at least 2 to 3 months to provide evidence in post-incident analysis, or judicial authority inquiry.
f) Perform configuration audit requesting vulnerability assessments at every stage during developing, testing and pre-production phases.
g) Request to perform security scans to discover (see also 8.b above) vulnerabilities before attackers. Vulnerability assessments are also possible for websites in the cloud following Cloud Service Providers' rules. You can request and schedule scans contacting DSO and review results for improvements;
h) Request to perform **penetration testing** on the most critical websites at least once a year to discover misconfigurations and vulnerabilities.

2.8. The compliance to security standards

a) The compliance with the latest **ISO 27001,** an international standard of information security.
b) The compliance with PCI DSS, Payment Card Industry Data Security Standard is an information security standard used to handle credit cards from major card brands. The standard is administered by the Payment Card Industry Security Standards Council, and its use is mandated by the card brands.

2.9. Regulatory compliance

The application must comply with applicable data protection and privacy laws, including GDPR.

3. Application Functionality

3.1 Product Catalog

- **Searchable Database**: The product catalogue should be organized in a searchable database, allowing users to find items based on various parameters like name, category, or SKU.

- **Detailed Descriptions**: Each product should have a detailed description including technical specifications, dimensions, available colours, and other relevant information.
- **Cost Information**: Each product listing should include cost information (without VAT), in a clear and easy-to-understand format.
- **Product Availability**: The system should indicate the availability or stock levels of each product, updated in real-time if possible.

3.2 Order Management

- **Quotes**: Users should be able to request quotes for products, which should be generated automatically by the system with all relevant details, including estimated delivery times and costs.
- **Purchase Orders**: Once a quote is confirmed, after entering information relating to the cost centers for invoicing (e.g. financial commitment number), users should be able to transition it into a purchase order, directly from within the application.
- **Order History**: The application should maintain a history of all orders placed, available for review by authorized users.

3.3 Tracking and Delivery

- **Order Status**: Users should be able to check the status of their orders, updated in real-time, including stages like processing, dispatched, and delivered.
- **Delivery Estimates**: Upon placing an order, an estimated delivery time should be provided, based on product availability and shipping methods.
- **Delivery Notifications**: Automated notifications, via email or within the application, should be sent at key stages of the delivery process.
- **Special Delivery Options**: The system should offer different delivery options, like shipping, if applicable, and clearly outline the additional costs for these services.

4. Interoperability Requirements

4.1 API Integrations

The application must offer robust API (Application Programming Interface) capabilities to allow for seamless integration with other institutional systems like financial management or inventory management systems.

4.2 Data Formats

The application should support common data interchange formats like JSON to ensure compatibility with other systems.

4.3 Export Capabilities

Users should have the capability to export reports, orders, and other relevant data in commonly used formats like PDF, CSV, or Excel for analysis or archival purposes.

4.4 Batch Processing

The application should offer batch processing capabilities to handle large sets of orders, updates, or data imports and exports, without affecting system performance.

4.5 Real-Time Syncing

The application should provide real-time data synchronization capabilities to ensure that changes in one system are immediately reflected in others.

4.6 Error Handling and Recovery

The application must have robust error handling and recovery features to ensure that any issues during data transmission between integrated systems can be easily identified and corrected.

4.7 Scalability for Interoperability

The architecture must be scalable not just in terms of user load and feature set but also in its ability to integrate with an increasing number of external systems as needed.

5. Maintenance and Support Requirements

5.1 Service Availability

The application must guarantee at least 99.9% uptime, ensuring that it is accessible for users during crucial operational hours.

5.2 Help Desk Support

Customer support should be available to assist users with any issues or queries they may have. Support channels should include email, chat, and phone support.

5.3 Software Updates

Regular software updates must be carried out to enhance features, correct bugs, and improve overall security. The scheduling of these updates should be communicated in advance and should occur during non-peak hours to minimize disruption.

5.4 User Training

Initial training sessions should be provided for key personnel, along with refresher courses for new updates or functionalities. These could be in the form of webinars, tutorial videos, or in-person sessions.

5.5 Security Patches

Security patches must be applied promptly in response to the discovery of any vulnerabilities, with minimal downtime and prior notification to users.