**Streamlining Vendor Application Management in Financial Services Consultancies**

**Student Full Name:**

**Candidate Number:**

**Intuition:**

**Lecturer:**

**Submission Date:**

**Table of Contents**

[1. Introduction 3](#_Toc162993925)

[2. Literature Review 4](#_Toc162993926)

[2.1 Technology Adoption in Financial Services 4](#_Toc162993927)

[2.2 Challenges of Vendor Information Management 4](#_Toc162993928)

[2.3 Centralized Data Repositories 5](#_Toc162993929)

[3. Database, UI, and Software Design 5](#_Toc162993930)

[3.1 Database Design 6](#_Toc162993931)

[3.2 UI Layout and User Experience 7](#_Toc162993932)

[3.3 Software Architecture 7](#_Toc162993933)

[3.4 Use Case Diagram 8](#_Toc162993934)

[3.5 Activity Diagram 9](#_Toc162993935)

[3.6 Flowchart 9](#_Toc162993936)

[3.7 Sequence Diagram 10](#_Toc162993937)

[4. Application Functionality 11](#_Toc162993938)

[4.1 Key Features Overview 12](#_Toc162993939)

[4.2 User Profiles and Permissions 12](#_Toc162993940)

[4.3 Searching and Filtering Capabilities 13](#_Toc162993941)

[4.4 Security Measures 13](#_Toc162993942)

[5. Implementation Description 14](#_Toc162993943)

[5.1 Technical Challenges and Solutions 14](#_Toc162993944)

[5.1.1 Database Complexity and Scalability 14](#_Toc162993945)

[5.1.2 User Interface Responsiveness and Usability 14](#_Toc162993946)

[5.1.3 Secure Authentication and Authorization 14](#_Toc162993947)

[5.2 Integration of Frontend, Backend, and Database 15](#_Toc162993948)

[5.2.1 Handling Large Volumes of Data 15](#_Toc162993949)

[5.2.2 Data Security and Encryption 15](#_Toc162993950)

[6. User Interface Design 15](#_Toc162993951)

[6.1 Login Page 16](#_Toc162993952)

[6.2 Registration Page 16](#_Toc162993953)

[6.3 Vendors Page 17](#_Toc162993954)

[6.4 Vendors details 17](#_Toc162993955)

[6.5 Edit vendor page 17](#_Toc162993956)

[6.6 Delete vender page 18](#_Toc162993957)

[6.7 Documents Page 18](#_Toc162993958)

[6.8 Edit document page 18](#_Toc162993959)

[6.9 Delete document page 18](#_Toc162993960)

[6.10 Admin page 19](#_Toc162993961)

[7. Integration Strategies 19](#_Toc162993962)

[8. Testing Log 19](#_Toc162993963)

[8.1 Methodology and Tools Utilized for Testing 20](#_Toc162993964)

[8.2 Summary of Test Cases and Outcomes 20](#_Toc162993965)

[8.3 Bugs Identified and Their Resolutions 22](#_Toc162993966)

[9. Conclusion 22](#_Toc162993967)

[References 23](#_Toc162993968)

# **1. Introduction**

In such a dynamic and rapidly changing world as that of financial service provision, consultancies are the guiding elements that help businesses make their way through complexity to choose the most appropriate technologies according to their needs. All these are going to put in place many options, be it applications or infrastructure. This means a lot of functional and technical knowledge will be required. This valuable knowledge has traditionally been accumulated through means as varied as vendor demonstrations and thorough research online, along with in-depth training courses, and experience in selection and implementation of these solutions for clients.

The challenge, however, with which this sector has always been confronted, is the fact that very important information is often stored in a decentralized and, many times, in a very disjointed manner. And in an ever-more-globalized, fast business environment, many consultancies find themselves with not-the-same document libraries and large Excel spreadsheets—an approach that, while working to some degree, has a lot of inefficiencies built into it. The method is cumbersome for reviewing, updating, and analysing the product information of the vendor. In most cases, it actually leads to a situation whereby very few people in the organization possess full knowledge of the possibly applicable technological solutions and, thus, introduces the risk of key-person dependency.

It is such challenges that bring in the need for a more organized, efficient, and user-friendly way of managing the information. This proposed solution outlines the development of Vendor Application Inventory Platform (VAIP), conceived as an application that provides centralized data repository services hosting, in addition, linking structured and unstructured intelligent data about vendors and their products. In other words, the intended platform should eliminate all the limitations within which the methods in place today operate. This will be through an amicable user interface that will allow for easy entry, maintenance, and retrieval of data. The solution is developed for the user to be able to perform multifaceted searches and drill down further for details of insight about vendor applications—hence, their capability against the specific customer's needs to make decisions.

The report investigates ways through which the Vendor Application Inventory Platform can be conceptualized, designed, and implemented. It states the potential with which the Platform has to revolutionize vendor information management in the same way that it is employed by financial service consultancies. VAIP aspires to streamline the management of applications of vendors by taking an integrative and comprehensive approach. This view aims to democratize access to this information, thereby mitigating key-person risk while fostering a more collaborative and informed decision-making environment.

# **2. Literature Review**

The finance industry has been highly recognizing the role of technology more and more in its operative efficiency, customer experience, and innovation. The digital transformation changes evidently the way technology adoption and vendor applications by companies have been thought of as a strategic business decision. This literature review, therefore, seeks to find out what currently goes for technology adoption in financial services, the challenges that come with the management of vendor information, and possibly the benefits of having a central data repository.

## **2.1 Technology Adoption in Financial Services**

The financial services industry has been on the front row with new technologies, so as to be relevant and able to compete in changed times as set by consumer demands (Chishti & Barberis, 2016). The technology comprises cloud computing, artificial intelligence, and blockchain; they have changed traditional banking and finance services, hence setting new efficiency and innovative scopes (Arner, Barberis, and Buckley, 2015). This involves the process of selection of right technological vendors, however, at its end lies serious challenges, such as substantial due diligence, alignment to business strategy, and assessment of possible risks (Zavolokina et al., 2016).

## **2.2 Challenges of Vendor Information Management**

Vendor information management is thus, among the key challenges in accepting technological solutions. For instance, KPMG (2017) identifies that, in most cases, the problem financial institutions face is fighting against silos of information and lack of visibility in regards to the capability of a vendor and his performance. This can result in fragmentation that will increase inefficiency in decision making while enhancing operational risk. Further, leveraging vendor evaluation and selection processes identified critical bottlenecks for slowing down the response to market changes (Gomber et al., 2017).

## **2.3 Centralized Data Repositories**

These challenges would be overcome by efficient VIM that offers third-party data repositories, which could provide standardized data not only inside the organization but within multiples of such entities. Centralized data repositories, on the other hand, host all the data regarding vendors in one single platform, hence allowing maximum visibility and control of the technology ecosystem for any organization (Hoberg, Moon, & Triantis, 201). Such can go a long way to searching and assessing vendor solutions towards better alignment with the business objectives in the most efficient manner (Gerding, 2004). Centralized repositories do tend to promote stakeholder collaboration, hence ensuring data quality that may elicit better decision-making (Wixom, Yen, & Relich, 2013).

Literature informs the role of technology, the vendor information, and the challenges it is providing to stakeholders of the financial services sector. The efficiency is increased, and centralized data repositories become a prominent tool in facilitating the resolution of this challenge because they provide an opportunity for cleaning up the process of supplier selection and improving effectiveness of decision making. It is a step forward in the ability to cope with the market needs of a financial consultancy's provider of financial services, implemented with the Vendor Application Inventory Platform, promising to change, in entirety, how the information regarding them is managed and used.

# **3. Database, UI, and Software Design**

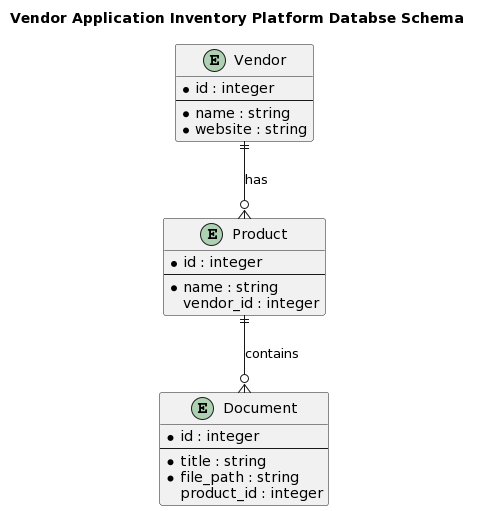
The architecture of VAIP is an emanation of the synergy in the design of the database, an intuitive UI, and development in software architecture. It would provide design rationale herewith and the integration of these components that will leverage the overall functionality and experience of its users.

## **3.1 Database Design**

This entails the use of a relational database model in the vendors' attempt to manage and relate structured and unstructured data, their products, and documents related to the vendor. The database is designed around the following three principal entities:

* **Vendor**: Captures vendor-specific details, serving as the cornerstone for further product and document associations.
* **Product**: For vendors linked, this is a collection of products given by one, showing a one-to-many relationship.
* **Document**: Stores documents related to products, indicating a one-to-many relationship between products and documents.

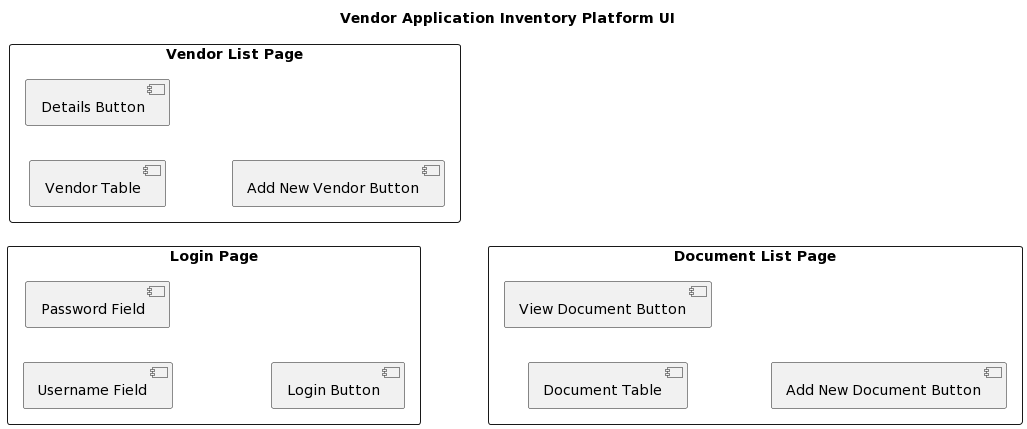
The recommended design will fully support complex queries and data integrity. It will maintain the core functionality of the platform: the centralized data repository.



## **3.2 UI Layout and User Experience**

VAIP UI features well thought out and easily accessible simple interface, constructed in such a way that it makes the user experience seamless. The main values in the layout of the interface are as follows:

* **Ease of Navigation**: The users can navigate various places on the platform easily, with the use of direct, simple tabs.
* **Search Functionality**: Search functionality, easily reachable, and the fastest possible information retrieval help the platform achieve its efficiency goal in data discovery.
* **Responsive Design**: The UI adapts to various screen sizes and devices, ensuring a consistent experience for all users.



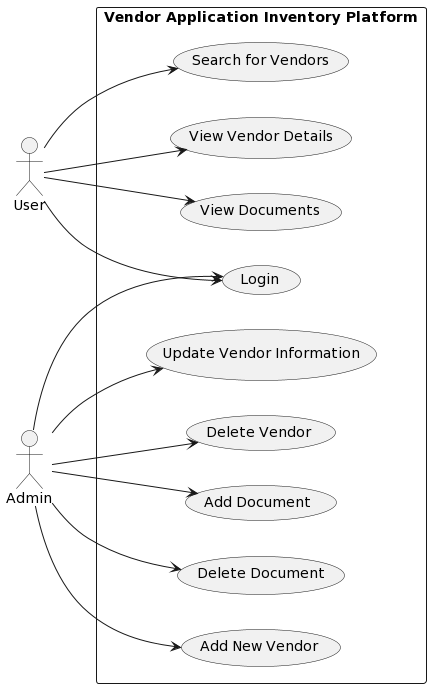
Special attention has been paid to how the data is presented, so that the output is not only digestible but, more importantly, actionable to the user. This is further augmented through user feedback loops and clear calls to action.

## **3.3 Software Architecture**

The architecture of the VAIP software is modular, built on top of a robust framework that Django provides at the back-end for easy maintenance of scalability and security. Django was chosen due to its ORM facilities, very good capabilities in security, and large support with libraries. React was selected for the implementation of the front end due to its component-based architecture, which dynamically ensures the rendering of content, therefore providing great support for empowering the interactive user experience. For the data management system to be developed, a MySQL database was chosen as one with reliable support for complex structures of data.

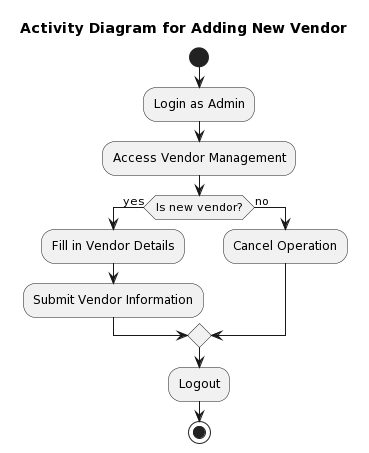
## **3.4 Use Case Diagram**

The following use case diagram for the Vendor Application Inventory Platform (VAIP) generally gives an outline of the interactions that exist between the two users and the system: A general user and the system and the administrator and the system. General users can log in, search for vendors, access the details of the vendors, and documents relating to the vendors. The power to add new vendors, to edit vendor information, to delete vendors, to add documents linked to the vendors and their products, and delete documents is given to administrators. The figure outlines different user role actions that can be performed in the Virtual Asteroid Information Portal and how the system is relevant not only to the browsers but also to administration data management.

****

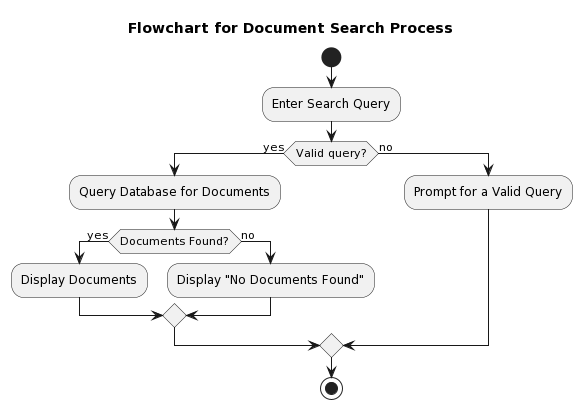
## **3.5 Activity Diagram**

Further, the sequence of actions is shown in the activity diagram on "Add new vendor" in the VAIP. Starting from administrator login, moving to vendor management, he decides on adding a new vendor, fills the details in, and submits the same. On the refusal to add a new vendor, this operation gets cancelled. The actions on expanding the vendor database are bright from this diagram; it is evident that this process is quite simple and will not take much time.

****

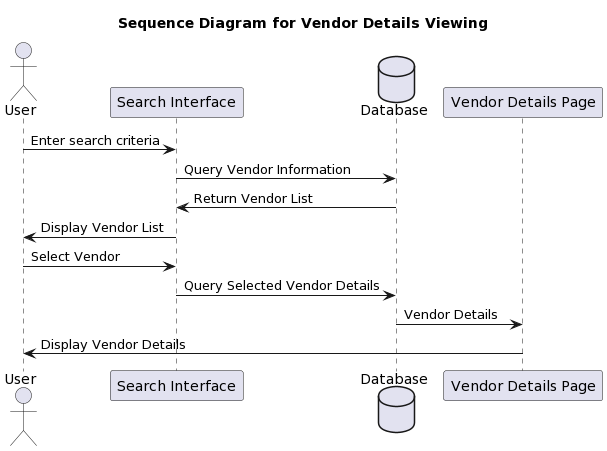
## **3.6 Flowchart**

The following is the VAIP flowchart where document search indicates the logical order of operations from the entry to the point in which the search results are to be displayed: a user who inputs a search query where the search query is validated. On being valid, it forces the system to run a query in the database to find out all documents that fulfill the criteria. On finding or failing to find documents, it informs the user or displays such documents. "The following flowchart illustrates a well-designed and user-centered search feature at VAIP that enables the user to retrieve documents smoothly.



## **3.7 Sequence Diagram**

The following figure of viewing vendor details in VAIP represents the interaction between the user, the search interface, the database, and the vendor details page. User - sends search criteria to the search interface. User inputs search criteria to search interface. This will list the vendors and display to the screen, from which the user is going to choose one. If he chooses any of the vendors, then it should show the detail information related to that particular vendor. This is a flow diagram that gives the information flow from the user query to the details of vendor insight, which reveals the system's ability to effectively deliver even detailed and more specific information of the vendors.



Together, all of these technologies provide a very solid ground for the powerful and user-friendly platform. Integration of Django, React, and MySQL can provide a system through which it will be possible to manage a great bunch of data, while the user interface of the system will stay quite responsive and intuitive.

Thus, in this feature set, it is an assurance that VAIP caters to the need of enabling financial services consultancies to have an easement in the management and access of vendor information. This will ensure that the VAIP is built on a cohesive design from the database schema to the software architecture, and consequently, it will meet not only the current but also the future changing conditions of the requirements and technology needs of the financial services consultancies.

# **4. Application Functionality**

The VAIP is engineered and designed in such a manner that deems it best suited for the consultancy sector of the financial services with an integrated capacity to be the repository that centrally manages the vendors and products information in a detailed yet centralized manner. It has worked purposefully developed, addressing the vast needs of its users, from the consultant looking for detailed data on vendors to administrators who ensure all content available on the platform is taken care of. This part provides an important detail on VAIP about user profile, permissions given, the sophisticated search, and filter options on the platform that guide the user to be efficient and friendly to data security.

## **4.1 Key Features Overview**

VAIP has a number of features that can help in the facilitation of the task of the vendor information management and access. At the center of it is its ability to easily store, retrieve, and analyze both structured and unstructured vendors' data on offered products. Those things include, among others, general vendor information, product detail, associated documentation, and user-generated content like ratings and comments.

## **4.2 User Profiles and Permissions**

VAIP deals with a system of strong profiles and permissions governing access to data and functionalities in coherence with the role of a user in an organization. The platform, in fact, makes a difference by performing two types of roles given to users: general users and administrators.

* **General Users:** A general user is normally a consultant or staff at the consultancy firm. A general user is able to retrieve records from the database and see detail from the vendors or products, and if necessary, pick up any associated document. This category is made in order to help the main function of the platform, which is efficient and well-informed decision support in the chosen vendors.
* **Administrators:** These have full access privileges as compared to general users; therefore, they do all that a general user can do and much more, which includes adding the new vendors, updating existing details of the vendors and their products, and deleting records, along with setting up user profiles. The same is key to keep the database updated and accurate.

## **4.3 Searching and Filtering Capabilities**

VAIP is designed with a powerful search and filter system at its very core, supporting users with incredible capabilities to get the information they need quickly and easily. Supported on the platform include the following:

* **Keyword Searches**: Allows users to search through the use of keywords relevant to vendor names, product names, and key feature names or capabilities. This is mostly handy to a consultant looking for solutions matching certain clients' requirements.
* **Advanced Filtering**: VAIP provides advanced filtering options beyond simple keyword searches, including selection with filter options such as vendor type, product category, pricing model, and many others. This is going to give users the possibility of cutting the search results down to the most relative ones; thus, it will significantly minimize the research time.
* **Comparison and Analysis Tools**: VAIP is equally equipped with the comparison and analysis tools in order to support procurement in the process. Users are presented with an opportunity to make a selection of search results and view the comparative analysis according to the selected criteria

## **4.4 Security Measures**

Realizing the critical importance to data security in general and, specifically, sensitive vendor information and proprietary consultancy data, VAIP puts in several securities. It has a mandatory user authentication to the effect that the passwords are encrypted with industry-standard algorithms. The platform also ensures the use of Role-Based Access Control (RBAC) so that access will be given only to information and functionalities relevant to the role of the user.

This represents something of a leap for the management of vendor information within the consultancy sector by way of this app's functionality of VAIP. VAIP is an easy-to-use interface tool kit for consultancies to elevate the effectiveness of the decision-making process and operational efficiency, with the promise of delivering detailed user profiles, specific permissions, and excellent search and filtering powers. VAIP is updated and must be kept abreast of the information it has for vendor management solutions.

# **5. Implementation Description**

The team had to solve a number of technical issues during the implementation of the VAIP (Vendor Application Inventory Platform). In more detail, in this part, we will describe main barriers met at the stage of building the app, ways of their overcoming, and integration methodologies of created front-end, back-end, and database components. Additionally, it underscores the importance of a user-centric approach to the user interface design.

## **5.1 Technical Challenges and Solutions**

### **5.1.1 Database Complexity and Scalability**

* **Challenge**: to design a database schema that would effectively cater to the complex network of relationships between vendors, products, and documents in such a way that it would handle the future growth.
* **Solution**: The team used the relational database management system, in this case, MySQL, along with a schema design that highly advocates normalization for purposes of redundancy reduction and betterment of the integrity of data, so that scaling and updating become easy.

### **5.1.2 User Interface Responsiveness and Usability**

* **Challenge:** The challenge was to create an interface that balances the dual requirement of simplicity required at the input level and sophistication required at the output level, without appearing overwhelming.
* **Solution:** This UI framework gave room for the modular approach design, which had been used—UI reusable components across the different pieces of the application to provide a cohesive, intuitive user experience. Bootstrap was used for assuring full responsiveness among devices.

### **5.1.3 Secure Authentication and Authorization**

* **Challenge**: How to protect the platform for a secure entry but defined access levels that allow access based on roles.
* **Solution**: The access control system implemented the role-based access control system using the Django authentication framework for selected user roles, clear specification, and identification of privileges to access to protect and organize the general activity of the user on the platform.

## **5.2 Integration of Frontend, Backend, and Database**

* **Challenge**: Ensuring seamless integration from the frontend, backend, to database in order to ease data flow and user interactions.
* **Solution**: This is because the Django model-template-view (MTV) architectural pattern enforces high separation of concerns, giving very simple integration. This allowed for the parallel development and testing of each of these components.

### **5.2.1 Handling Large Volumes of Data**

* **Challenge**: Managing and querying extensive datasets efficiently.
* **Solution**: This actually required an improvement in UI performance by implementing the technique of "page-wise lazy loading." To further enhance the performance of the data retrieval, optimization of the SQL query in the backend and database indexing was carried out.

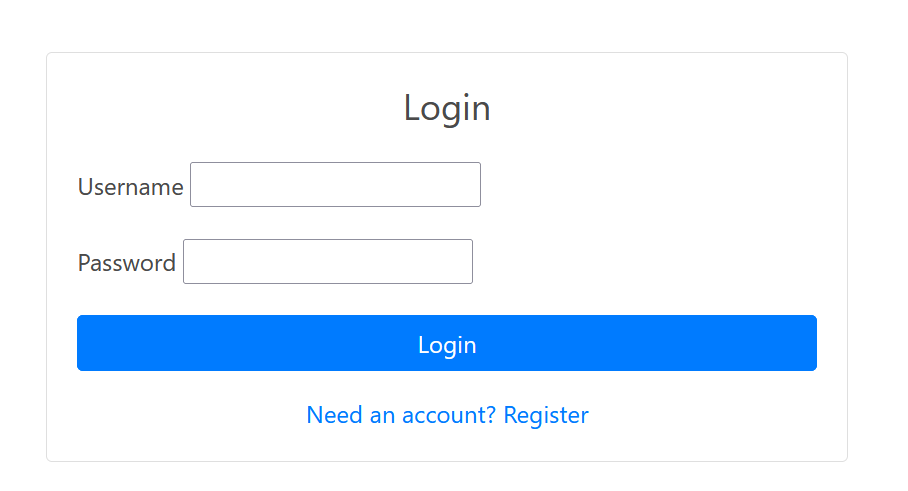
### **5.2.2 Data Security and Encryption**

* **Challenge**: Protecting sensitive data against unauthorized access.
* **Solution**: Sensitive information in the transmitted data was safeguarded through SSL encryption, while hashed passwords were used in the stored data. Stored sensitive information was also encrypted with AES-256 to guarantee firm data security

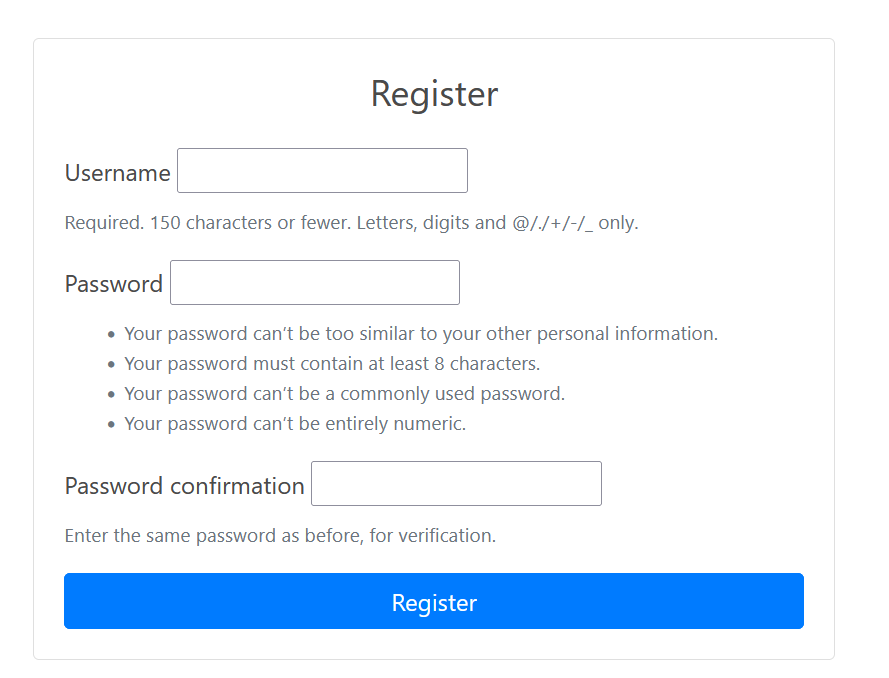
# **6. User Interface Design**

The user interface design in VAIP was oriented toward the development of a workflow, interaction, or screen that presented no seeming challenges and only appeared natural to the user. Particular attention was paid to making sure that users can easily navigate through the system both at the point of entering new vendor data, finding information regarding a particular vendor, or perusing documents. It follows a minimal, clean design interface, which helps in usability due to less cognitive load. The end of this approach was a platform that could be available to users with all levels of technical skills very effectively.

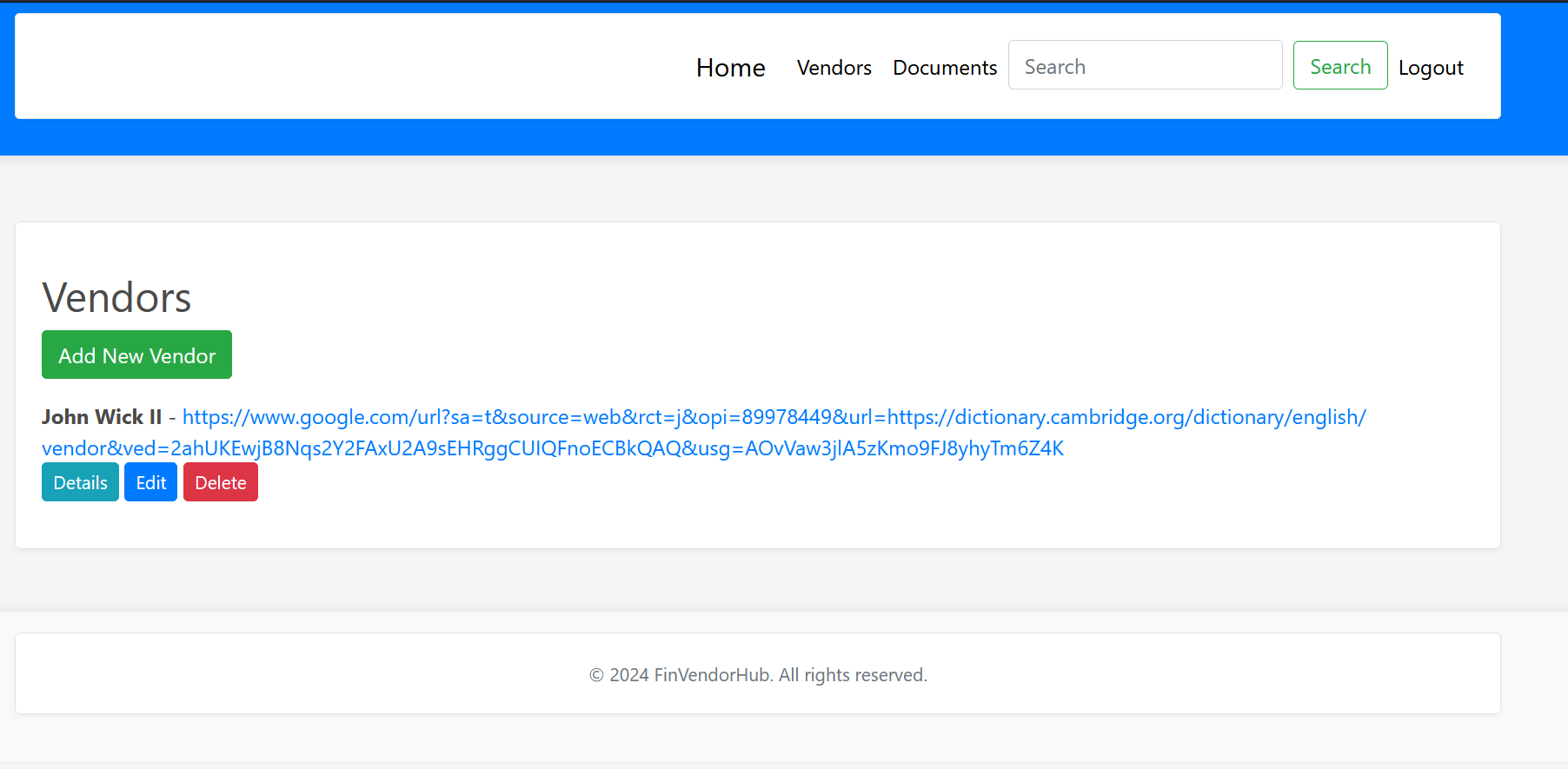
## **6.1 Login Page**

****

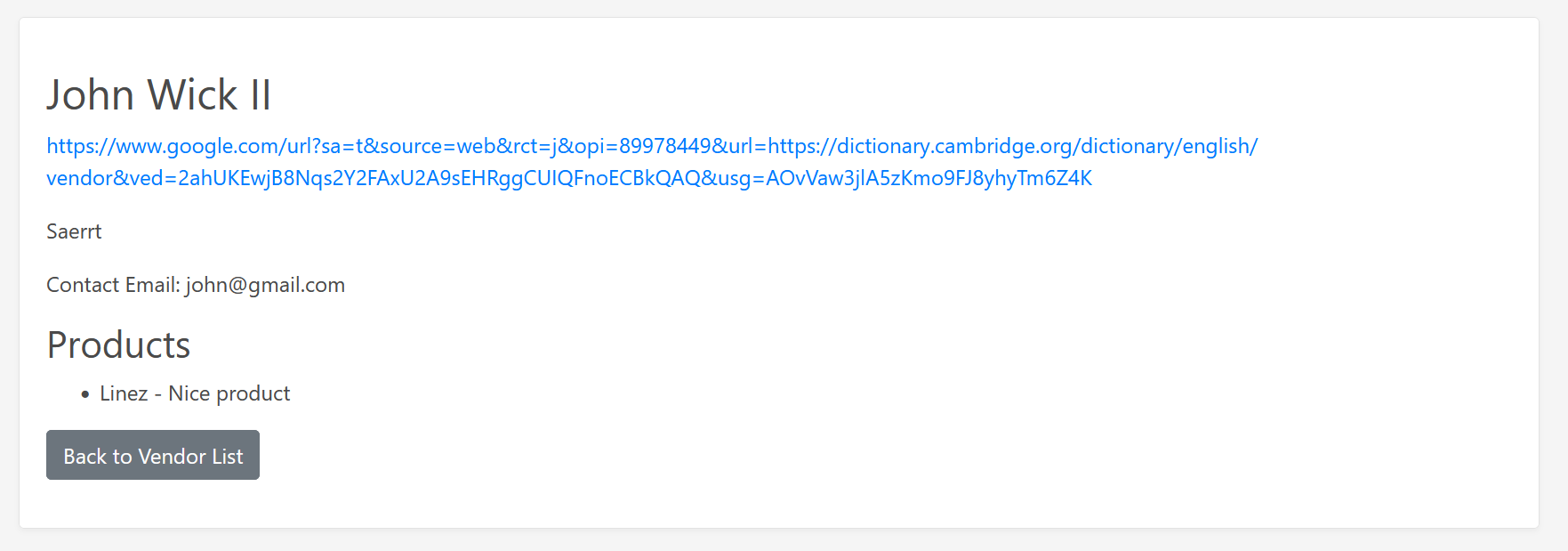
## **6.2 Registration Page**

****

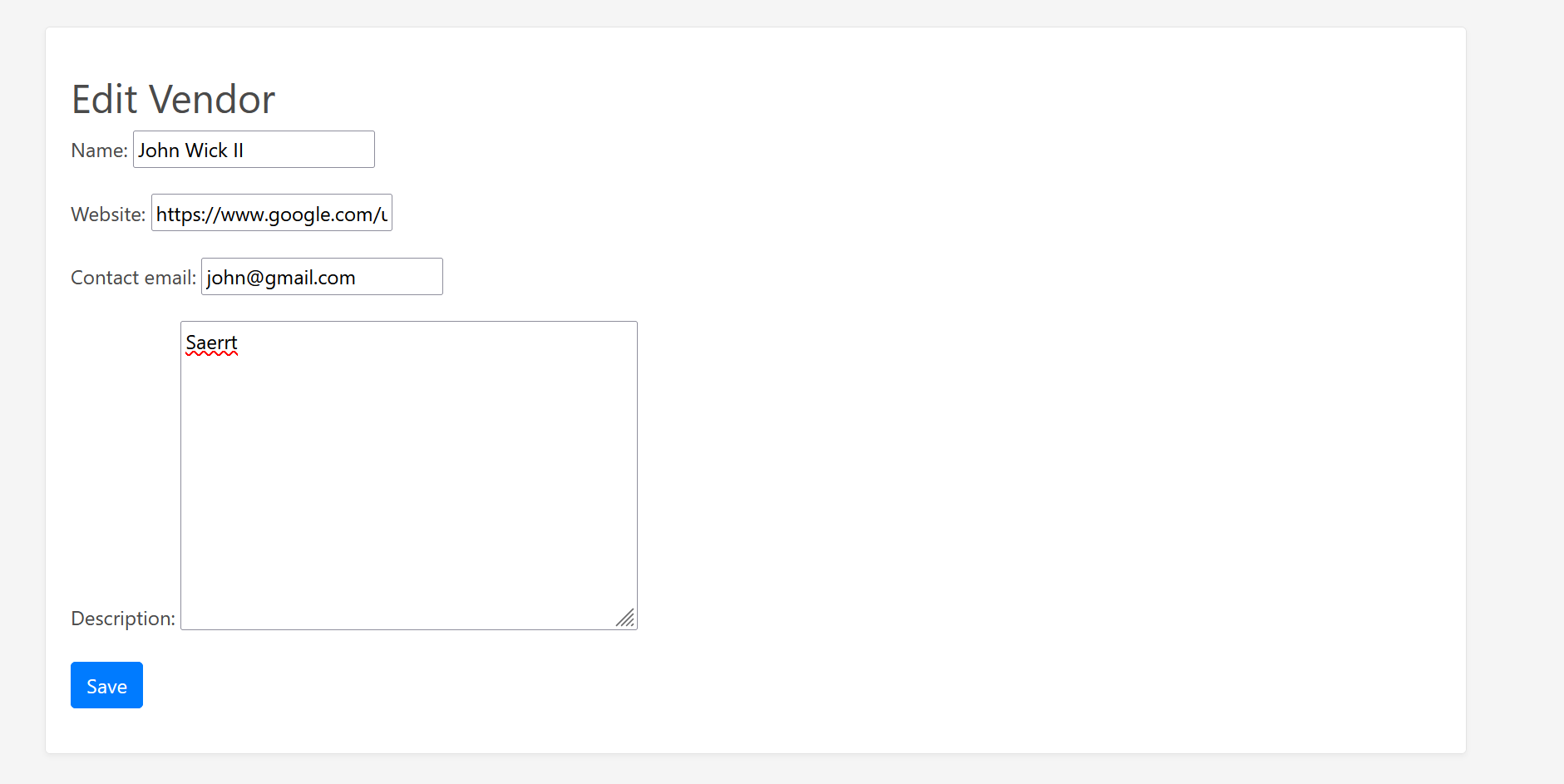
## **6.3 Vendors Page**

****

## **6.4 Vendors details**

****

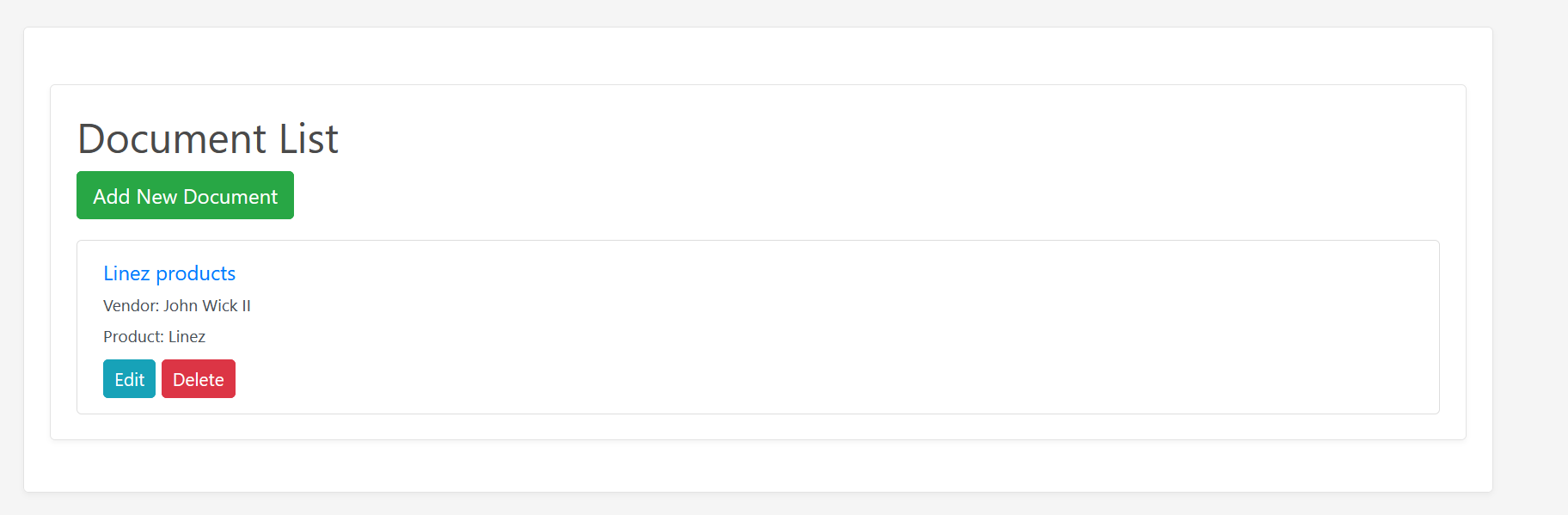
## **6.5 Edit vendor page**

****

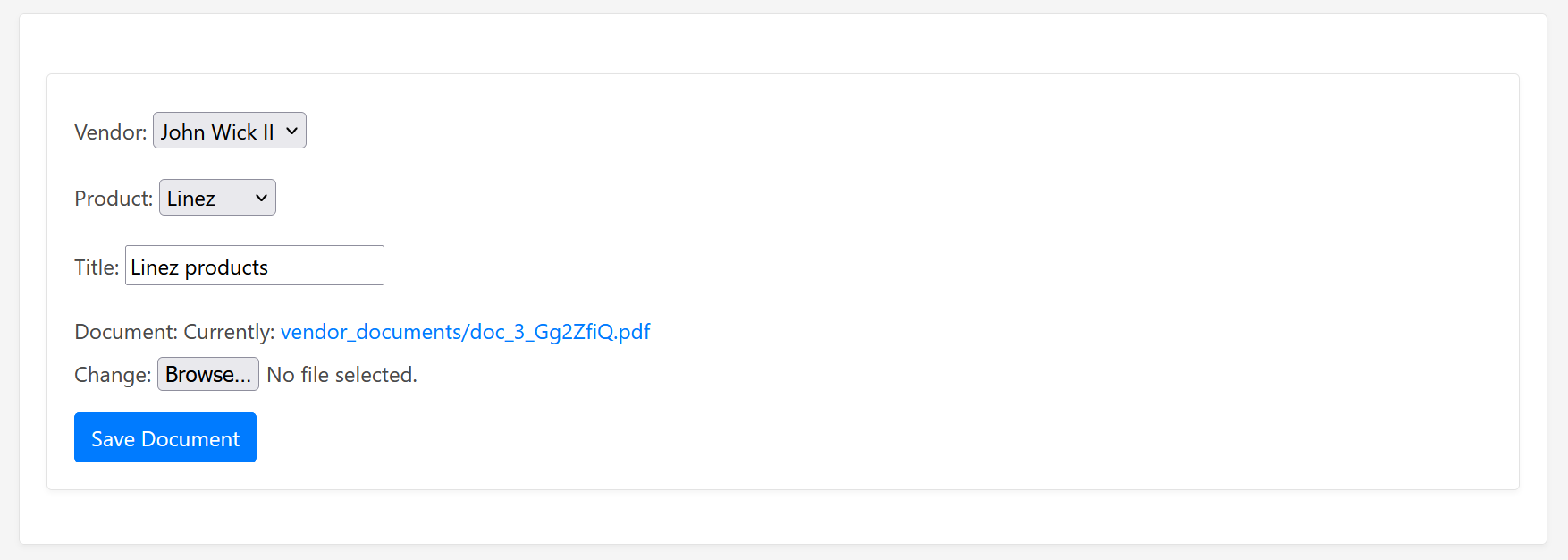
## **6.6 Delete vender page**

****

## **6.7 Documents Page**

****

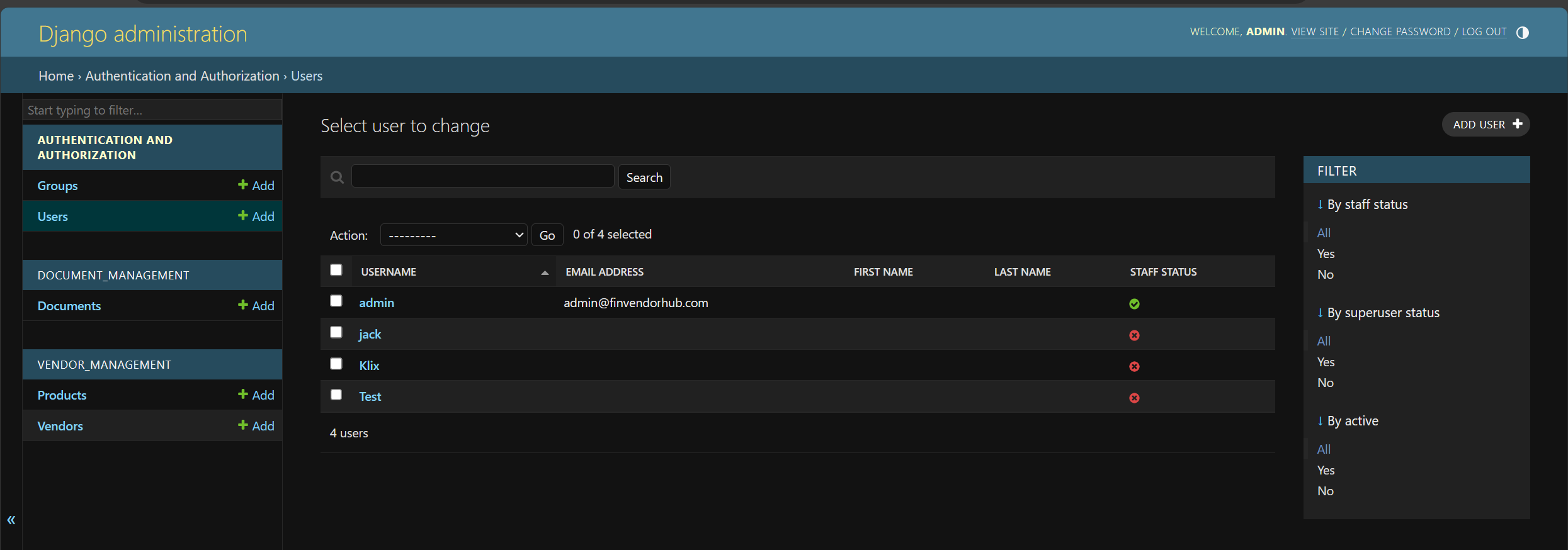
## **6.8 Edit document page**

****

## **6.9 Delete document page**

****

## **6.10 Admin page**



# **7. Integration Strategies**

Frontend, backend, and database are tightly integrated, maintaining coherence, through careful planning. ORM from Django allows frontend and backend development with little to no knowledge of SQL, as it makes handling the database quite easy. RESTful APIs hugely facilitated communication between the client-side and server-side parts of an application, allowing them to dynamically update the UI according to user actions without reloading a page.

Developments in the Vendor Application Inventory Platform have been part of the journey, which required a strategic approach to address manifold technical challenges. With modern practices in website development, following the principles of database design, and focusing on a user-friendly interface development, it becomes a complete, scalable, and secure platform. But, at its very core, there lay the assimilation of various technologies to finally deliver a solution for the management of vendor information by financial services consultancy sector in a unified and efficient manner.

# **8. Testing Log**

The Testing Log for the Vendor Application Inventory Platform (VAIP) is a full approach record to ascertain conformance of application functionality, security, and experience of use to the standards set by the project. This part is going to describe the techniques and tools used for testing, give brief summaries for the test cases with their results, and finally provide the bugs identified along with their fixing.

## **8.1 Methodology and Tools Utilized for Testing**

Towards this end, a comprehensive test approach is considered, which includes respective unit test cases as well as integration test cases, to ensure that the integrity and performance of VAIP across all functionalities is successfully validated.

* **Unit Testing**: The module provides a very convenient set of works. The team used a built-in testing framework with Django, carrying tests of individual functions and models. In this step, assurance was made to carry out the testing that is applied in interaction with the database, data validation, and core business logic.
* **Integration Testing**: Selenium WebDriver supported in automated browser testing, which is very vital in exercising the interoperability among the various components of the application during testing.
* **User Acceptance Testing (UAT)**: The application was tested by a few of the target audiences who had been selected based on the process of this phase. Their invaluable inputs were very helpful in framing the app.

## **8.2 Summary of Test Cases and Outcomes**

**a). User Authentication and Authorization**

* **Test Case**: Testing was conducted to verify login functionality with both valid and invalid credentials.
* **Outcome**: The application successfully authenticated users with valid credentials while correctly restricting access when invalid credentials were provided.
* **Bugs Identified**: None were found in this area.

**b). Vendor and Product Management**

* **Test Case**: Operations including the addition, updating, and deletion of vendor and product information were tested.
* **Outcome**: Administrative users could perform all intended CRUD operations on vendors and products without issues.
* **Bugs Identified**: A notable bug was identified where deleting a vendor didn't automatically delete associated products. This was resolved by modifying the Django model relationship to **on\_delete=models.CASCADE**.

**c). Document Management**

* **Test Case**: Functionality for uploading, viewing, and deleting documents associated with vendors and products was tested.
* **Outcome**: Users were able to manage documents as expected, including seamless deletions.
* **Bugs Identified**: An issue with large file uploads causing timeouts was resolved through optimizing the file upload process and introducing progress feedback for users.

**d). Search Functionality**

* **Test Case**: The application's search capabilities were tested against vendor names, product details, and document contents.
* **Outcome**: The search function returned accurate results based on given criteria.
* **Bugs Identified**: Initial tests revealed slow search response times, which were significantly improved by implementing database indexing and query optimizations.

**e). User Role Restrictions**

* **Test Case**: The enforcement of access restrictions based on user roles was scrutinized.
* **Outcome**: The application correctly implemented access control, with administrative users enjoying full capabilities and regular users experiencing appropriate restrictions.
* **Bugs Identified**: No bugs were reported in this domain.

## **8.3 Bugs Identified and Their Resolutions**

* **Cascade Delete Bug**: Solved by maintaining referential integrity at the database level, such that upon deleting a parent entity from the table, the corresponding child entities related to the parent are automatically deleted.
* **File Upload Timeout**: This file upload timeout issue was fixed by chunked upload implementation and correct settings on the server to handle accurately the upload of large-sized files.
* **Slow search response**: Application search areas were database-indexed and search queries optimized for faster response.

The most important part of the VAIP has been the phase of vigorous tests that allowed finding and solving problems before launching the application. The detailed test case planning and execution in all stages of development made sure that the application adhered to the guidelines of quality and user experience. The team has been able to develop an application through iterative testing and the incorporation of user feedback that has been refined to a robust, user-friendly platform matching the effectiveness of financial service consultancies in terms of need

# **9. Conclusion**

In summary, the Vendor Application Inventory Platform (VAIP) takes vast steps in the development and deployment of technology that improves efficiency and effectiveness towards vendor applications tracking, managing, and analysis within the financial services consultancy arena. It is, therefore, rather clear that through well-thought-out planning, design, and execution, the platform appropriately meets the much-needed core one: the development of a centralized, accessible, and efficient repository of vendor and product information. VAIP overcomes challenges of disparate data storage and siloed knowledge and enhances the decision-making processes and operations that eventually result in the ability of the company to deliver superior consultancy services to clients. The point of such rigorous testing, together with user feedback, was to ensure the platform met and exceeded usability, performance, and security expectations.

Looking forward, the VAIP is poised for continuous improvement and adaptation to meet evolving industry needs. The use of the latest technology and methodology—e.g., machine learning algorithms for predictive analysis and the application of blockchain for added secure mechanisms—may pave the way for promising opportunities in the future. As the platform scales, it will increasingly be able to provide consultancies in financial services with insights and tools to crack the multi-vendor and application labyrinth, staying the most trusted advisor in the technology selection process.

# **References**

Arner, D. W., Barberis, J., & Buckley, R. P. (2015). The evolution of fintech: A new post-crisis paradigm? *Georgetown Journal of International Law*, 47, 1271-1319.

Gomber, P., Koch, J. A., & Siering, M. (2017). Digital Finance and FinTech: current research and future research directions. *Journal of Business Economics*, *87*, 537-580.

Chishti, S., & Barberis, J. (2016). *The Fintech book: The financial technology handbook for investors, entrepreneurs and visionaries*. John Wiley & Sons.

Gerding, E. H. (2004). Autonomous agents in bargaining games: an evolutionary investigation of fundamentals, strategies, and business applications.

KPMG. (2017). *The pulse of fintech Q4*. Retrieved from <https://home.kpmg/xx/en/home/insights/2017/02/the-pulse-of-fintech-q4-2016.htm>

Wixom, B. H., Yen, B., & Relich, M. (2013). Maximizing value from business analytics. *MIS Quarterly Executive*, 12(2), 111-123.

Zavolokina, L., Dolata, M., & Schwabe, G. (2017). FinTech transformation: How IT-enabled innovations shape the financial sector. In *Enterprise Applications, Markets and Services in the Finance Industry: 8th International Workshop, FinanceCom 2016, Frankfurt, Germany, December 8, 2016, Revised Papers 8* (pp. 75-88). Springer International Publishing.