|  |  |
| --- | --- |
|  | **T.C.**  **GEBZE TEKNİK ÜNİVERSİTESİ**  **MÜHENDİSLİK FAKÜLTESİ** |

**STAJ DEFTERİ**

**Öğrencinin;**

|  |  |
| --- | --- |
| **Adı, Soyadı** | **Burak KURT** |
| **Numarası** | **210104004225** |
| **Bölümü** | **Bilgisayar Mühendisliği** |
| **Staj Yaptığı Yer** | **Ödeal Yazılım ve Bilişim A.Ş.** |
| **Staj Tarihleri** | **19/07/2024-15/08/2024** |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:19.07.2024** | | **KAPSAMI: Team Allocation and Initial Project Steps** |
|  | |  |
| On the first day of the internship, all interns were allocated to different teams within the company. Afterward, we formed a separate team of 3 interns and met our team leader. In our first meeting, we learned that we would participate in daily meetings with our team, where we would discuss the tasks of the day and any potential obstacles. Additionally, we found out that every two weeks, we would participate in sprint review meetings where we would share what we had accomplished and what we planned to do next across the entire company. On the same day, we had a meeting with the company’s CEO. During this meeting, we were informed about the company’s mission, vision, and overall structure. The CEO shared the company’s future goals and their perspective on technology. Afterward, we began the research phase for the project and selected the libraries we would use; this involved deciding on the technologies to be utilized. We chose to use the Spring framework and build the project using a microservice architecture, starting by setting up the basic project structure in a local folder. At the end of the day, we presented a general project overview to the software manager. In this presentation, we explained our project plan and the technologies we would be using, and with the feedback we received, we further refined our roadmap. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:22.07.2024** | | **KAPSAMI: Setting Up the Development Environment** |
|  | |  |
| On the second day of the internship, I focused on setting up the development environment and configuring the project. I began by installing Java JDK and IntelliJ IDEA IDE. I reviewed the Maven configuration and added the necessary dependencies for the project. Since I had previous experience with Spring Boot, I was familiar with its structure and functionalities, which helped speed up the process.  Next, I accessed the project's GitHub repository based on the information provided by our team leader. I cloned the repository to my local machine via Git and used Maven to download all required dependencies. After ensuring that the dependencies were downloaded and configured correctly, I ran the application for the first time. The application launched successfully without any issues, confirming that everything was set up properly.  Later in the day, we had a meeting with our team leader to discuss the Git branching strategy. It was decided that we would follow the "Git Flow" model for version control. This meant that for each new feature, we would create a separate feature branch, which would then be merged into the develop branch. The master branch would only contain stable versions of the project, and all members of the team were expected to adhere to this strategy to ensure smooth collaboration.  By the end of the second day, I had successfully set up the development environment, cloned the project repository, and understood the Git branching strategy, setting the foundation for the upcoming development work.  **metin, ekran görüntüsü, menü içeren bir resim  Açıklama otomatik olarak oluşturuldu** | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:23.07.2024** | | **KAPSAMI:** **Planning the Backend Structure** |
|  | |  |
| On the third day of the internship, we focused on planning the backend structure and creating the various layers of the application. We began by deciding that the project would follow the Spring Boot MVC (Model-View-Controller) architecture. I discussed with my teammates how we would organize this architecture to be modular, ensuring that each component could be developed independently and tested easily in the future. We meticulously planned how the Controller, Service, and Repository layers would work together and clarified the responsibilities of each layer. The Controller layer would handle incoming requests and direct them to the relevant Service layer, while the Service layer would contain the business logic. Although the Repository layer would not be used in this project due to the lack of a database, we still outlined its structure in case a database connection would be required in future iterations.  Next, I started designing the model classes that would be used in the file conversion processes. These model classes would act as data carriers throughout the application. For example, I created classes to represent PDF and Word file formats. These classes were designed to carry not only the data related to the files but also any necessary metadata and parameters for the conversion processes. We aimed to keep the model classes as simple as possible, ensuring that each class represented a specific function while maintaining modularity within the project.  We also discussed the database structure. Early on, we decided that there was no need for a database connection in the project, as we were primarily focused on file conversion. Our team was pleased with this decision because it allowed us to concentrate directly on the core functionality of the microservices, speeding up development significantly. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:24.07.2024** | | **KAPSAMI:** **File Upload Service and Using Postman** |
|  | |  |
| Today, I focused on file upload and management. As a first step, I developed a service in our project to handle file uploads, along with a controller class to manage this service. This service allowed users to upload files via POST requests. The uploaded files were saved in the uploads directory, which we created in the main project folder.  While building the service, I used Spring Boot’s MultipartFile class to handle file uploads through POST requests. The controller class received the file and passed it to the service, which then saved the file to the specified folder. The file name and type were sent through POST requests, enabling easy handling of different file formats.  In addition, I installed and configured Postman to test the API requests and verify the functionality of the file upload process. I created collections in Postman to test the upload operations and confirmed that the service was working correctly by ensuring that the uploaded files were successfully saved in the uploads folder. This step laid the foundation for file processing in the project and became a crucial building block for the next stages. By the end of the day, I had successfully developed a file upload service and verified its functionality by testing it through Postman.  taslak, diyagram, tasarım içeren bir resim  Açıklama otomatik olarak oluşturuldu | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:25.07.2024** | | **KAPSAMI: Word Service and API Development** |
|  | |  |
| Today, I started by creating the Word services and the corresponding controller to manage the file conversion processes. My goal was to develop a functional structure using simple services to convert files between different formats. First, I created a ConversionWordPdfService class. Within this service, I handled basic tasks such as converting a TXT file to a Word document and vice versa. I also set up endpoints for converting PDF files to TXT and converting TXT files to Word documents.  After building the services, I linked them to the endpoints in the ConversionWordPdfController. For instance, I designed routes such as /convert/txt-to-word and /convert/word-to-txt, allowing users to upload files, convert them to the specified format, and save them to the uploads directory. I ensured that the uploaded files were processed correctly and were accessible afterward.  To carry out these conversions, I added the Aspose library to the project dependencies. Thanks to Aspose's flexible and robust conversion capabilities, I successfully performed various file format conversions, such as converting TXT to Word and PDF to TXT. To verify that the APIs were functioning correctly, I used Postman to upload files to the endpoints and carefully examined the outputs. I confirmed that the uploaded files were successfully converted and saved in the uploads folder. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:26.07.2024** | | **KAPSAMI:** **Postman Collections and Team Collaboration** |
|  | |  |
| Today, I started by creating collections in Postman and then set up a workspace. I added my team members to this workspace, ensuring that we could all collaborate within the same environment. After preparing collections with different scenarios for each API, my teammates also added their developed services to the workspace. We worked together on the same collections, tested the developed services, and addressed any issues that arose. Once the testing was completed, we pushed the project to the master branch on GitHub.  To keep the project up to date, we created new feature branches and began working on the latest version pulled from the master branch. This ensured that the project was ready for further development with the latest changes.  metin, ekran görüntüsü, bilgisayar, yazılım içeren bir resim  Açıklama otomatik olarak oluşturulduAdditionally, since it was Friday, I participated in the routine team meetings within the company. I provided a detailed report to our team leader and software manager regarding the project's status. During the meeting, we discussed the current state of the project and the challenges we faced. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:29.07.2024** | | **KAPSAMI: Fill Operation** |
|  | |  |
| Since it was Monday, we reached the halfway point of the sprint and held a general review meeting. During this meeting, my team and I reviewed our progress in the current sprint and divided the tasks for this week. We assessed the goals achieved in the first two weeks and set new sprint objectives.  Following the sprint planning, I began working on a new task related to filling PDF forms and working with Word document templates. I created a new controller and service class to handle this task. Specifically, I started developing a service to replace placeholders in the existing Word template files with data from JSON. In this process, I identified expressions enclosed in double curly braces in the template file and compared them with keys in the JSON data. If the keys matched, I replaced the placeholders in the template file with the corresponding values from the JSON data.  Additionally, I continued working with Adobe PDF templates downloaded from the internet. I aimed to correctly populate the fields in these templates with JSON data. This process will help in developing an effective and efficient method for filling out PDF forms and will provide a solid foundation for the next stages of the project.  metin, ekran görüntüsü, bilgisayar içeren bir resim  Açıklama otomatik olarak oluşturuldumetin, ekran görüntüsü, bilgisayar, tasarım içeren bir resim  Açıklama otomatik olarak oluşturuldu  FILL | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:30.07.2024** | | **KAPSAMI:** **Filling Word Documents and Template Manipulation** |
|  | |  |
| I developed a process to fill form fields in Word documents using JSON data. I learned to manipulate Word documents using the Apache POI library. I created a service that processes JSON data and inserts it into specific fields in the Word document. Initially, I used the Map class to handle JSON data, but it proved insufficient. Instead, I created a class named KeyValuePair, which includes variables like key, value, type, and size. I generated objects of the KeyValuePair class for each key-value pair from the JSON data and stored them in an ArrayList. The process of adding these objects to the ArrayList was accomplished using a recursive method. After loading the JSON data into the DocumentService via the DocumentController class, this structure was used to dynamically replace placeholder fields in the Word document.  To perform the replacement, I needed to access the structure of the Word document accurately. For this, I utilized specialized classes and methods from the Apache POI library. The XWPFDocument class was my fundamental class, and many of my methods were derived from it.  Inside the XWPFDocument class, I used sub-classes like XWPFParagraph and XWPFTable to handle specific content within the Word document. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:31.07.2024** | | **KAPSAMI:** **Document Service and Word Document Enhancement** |
|  | |  |
| After completing the KeyValuePair class, I established an overall architecture in DocumentService. This architecture was structured to process the Word document through various stages. Each process focused on different sections of the document. Initially, I accessed the header and footer sections of the Word document using the XWPFHeader and XWPFFooter classes. I checked the paragraphs and tables within these classes, comparing each placeholder with the contents of the KeyValuePair class to make the necessary modifications. I ensured that the text in the header and footer sections was correctly placed. Next, I accessed the body section of the Word document. I applied the same methods to check the content, processing paragraphs and tables. I compared each placeholder with the KeyValuePair data to make the appropriate changes. The most challenging part in the body section was handling tables. Since JSON data came in array format, I wrote a method called rowAdder to adjust the table structure to fit this data. This method updated the number of rows in the table and performed the replace operation by placing the data correctly. I adjusted the table size by adding and removing rows as needed to ensure the table was accurately updated.  As a result of these detailed operations, all placeholders in the Word document were correctly placed and updated with JSON data. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:01.08.2024** | | **KAPSAMI: Pdf Signing** |
|  | |  |
| I conducted extensive research on digitally signing PDF files. I focused on understanding and applying the cryptographic methods required for creating digital signatures. During this process, I used Java Keytool to create the necessary certificate and private key for digital signatures. I opened a service in the terminal, executed the required commands to generate the certificate, and added a password to it. I then used this certificate to sign PDF files. When I opened and checked the signed PDFs, I was able to verify who signed them, the date of signing, and whether the document had been altered.  Additionally, I used other libraries like Bouncy Castle to assist with the signing process. I paid attention to the secure management and storage of key pairs while creating the certificate. Understanding and applying hashing and encryption techniques became crucial for ensuring the integrity of the file during the digital signing process. Hashing allows verification of whether the file content has been altered, while encryption techniques ensure the secure signing and verification of the document. I worked on understanding how hash algorithm’s function and the role of encryption methods in the signing process. These preparations helped me acquire the essential knowledge and skills needed to securely sign and validate PDF documents. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:02.08.2024** | | **KAPSAMI: PDF Digital and Visual Signature Developments** |
|  | |  |
| In addition to the process of adding digital signatures to PDF files, I also began incorporating the functionality of adding visual signatures into the project, as per our manager's request. I started by creating a separate endpoint for this task. Initially, I used the AbstractXWPFConverter class to handle existing PDF files and established two different scenarios. If the PDF had a designated signature field, the service would automatically detect it and place the visual signature within that field. If no such field existed, I designed the process to add the user's name, surname, and signature image to the bottom of the document.  To implement the visual signature functionality, I first prepared signature templates. I defined specific areas and coordinates on the PDF and calculated dynamically where the signature would be placed. Additionally, I created template files for the company logo and signature images. I then developed a service that would insert the visual signature file (e.g., JPG or PNG) provided by the user into the PDF. This service placed the visual signature at the specified coordinates, along with the user's name and surname, within the document.  During development, I encountered some technical challenges, particularly with ensuring that the visual signature appeared in the correct location and size on the PDF. I addressed these issues by making various adjustments, focusing on correct positioning and scaling. Through fine-tuning, I ensured that the visual signature would be properly added to every document. With this update, I successfully developed a comprehensive signature service that allows PDF files to be signed with both digital and visual signatures. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:05.08.2024** | | **KAPSAMI: Network 101** |
|  | |  |
| Today, apart from our project work, we attended an internal company training session. A member of the DevOps team provided a comprehensive "Network 101" training, covering the fundamentals of network technologies. The session focused particularly on LAN and WLAN. We learned how LAN is used for wired connections within the company and how WLAN is configured for wireless devices to access the network. The training explained how these networks are managed, how devices connect to them, and how a balance is maintained between performance and security.  The company's use of a VPN was also discussed. We were shown how VPNs are utilized for remote access, allowing employees to securely connect to the company's network. Additionally, encryption techniques and security protocols used during data transmission over VPN were covered.  In terms of security, the training explained how the company blocks unnecessary network ports, keeping only the essential ones open. This minimizes potential external threats and ensures network protection. We also learned about the use of firewalls and security tools like IDS/IPS to prevent and detect network attacks. Overall, the training provided valuable insights into how networks are secured and emphasized the critical role of network management within the company. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:06.08.2024** | | **KAPSAMI:** **Virtual Machine and Docker Training** |
|  | |  |
| Today, the DevOps team provided an extensive training session related to our project. The training began with detailed instructions on virtual machines and the process of setting up servers on these machines. We covered the installation of operating systems on servers, how to configure them, and how to set up databases. We learned how to configure virtual machines, install databases, and ensure database performance and security. Emphasis was placed on the performance and security considerations for database configuration on virtual machines.  The training continued with a comprehensive overview of Docker usage. It was explained how Docker, with its containerization technology, makes software more portable and manageable. We were introduced to Docker's core concepts, how to create and manage containers, prepare Docker images, and run containers. Detailed information was provided on Docker Compose, including how to coordinate multiple containers, how to pull and push images from Docker Hub, and best practices for Docker usage. Additionally, the training highlighted how Docker can be used in projects to improve efficiency, manage different environments (development, testing, production), and isolate applications, emphasizing Docker's advantages in streamlining project development and infrastructure management.  metin, ekran görüntüsü, diyagram içeren bir resim  Açıklama otomatik olarak oluşturuldu | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:07.08.2024** | | **KAPSAMI: Endpoint Development and Testing** |
|  | |  |
| I developed new endpoints for PDF signing, OCR processing, and file conversion, expanding the functionality of the API. I thoroughly tested these endpoints using Postman, particularly verifying that the PDF signing and OCR processing endpoints were working correctly. During the tests, I checked for any errors in file uploading and processing and resolved potential issues.  I also integrated the signature and OCR services that my teammates developed into the Postman collections by setting up the necessary connections. This allowed the entire team to test the API more efficiently. Additionally, I restructured the system that my teammates previously built, where they manually entered file paths. Now, the endpoint only requires the file name and extension, and with Java’s root.resolve() method, the system automatically locates the file in the 'uploads' folder for processing. This improvement eliminated the need for manual file path entries, ensuring that the system correctly finds and processes the necessary files, making file management more efficient.  metin, ekran görüntüsü, yazılım, bilgisayar içeren bir resim  Açıklama otomatik olarak oluşturuldu | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:08.08.2024** | | **KAPSAMI:** **Bug Fixing and Testing Process** |
|  | |  |
| Today, we tested our project’s conversion, OCR processing, and PDF filling functionalities using different templates. During the tests, we noted the issues we encountered, and I focused on resolving the problems related to PDF filling.  The main issue arose when the JSON data contained nested arrays and the footer section had a table. In this case, the code did not work correctly. First, I fixed the methods handling the table in the footer section and connected the necessary functions to this part.  Next, to handle nested arrays, I modified the internal structure of the KeyValuePair class. If nested arrays were present, I marked the parent array type as UP\_ARRAY. I also made adjustments to the recursive method that reads the JSON data, ensuring it processes arrays correctly.  After implementing these changes, I reran the tests using the same templates and confirmed that the issues were resolved. The PDF filling process now works smoothly with nested arrays and tables in the footer. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:09.08.2024** | | **KAPSAMI:** **PDF-Word Conversion and Optimization Efforts** |
|  | |  |
| Today, I thoroughly tested the PDF to Word and Word to PDF conversion processes in our project and made optimizations to address performance issues. Since these conversions are critical for us, I focused particularly on the Word to PDF service. The conversion was being executed through the LibreOffice application by running its source file via the command panel. However, LibreOffice was failing to convert the footer section of the Word document properly, which was corrupting the file's structure. I manually intervened to fix the issues in the footer conversion and ensured that the structure is now processed correctly.  For the PDF to Word conversion, I used the Aspose library, which provided a smoother and more accurate transformation of the document, especially when handling complex tables and graphics. To tackle errors encountered during the conversions, I enhanced the exception handling mechanisms. This improved the system's resilience to incorrect data inputs, allowing it to provide more descriptive and guiding error messages in case of any failures.  Finally, I refactored the code according to clean code principles. By eliminating repetitive code blocks, I made the code more modular, readable, and easier to maintain. These improvements not only enhanced the overall performance of the system but also ensured a more robust structure capable of handling potential future errors more effectively. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:12.08.2024** | | **KAPSAMI:** **Adding Try-Catch Blocks** |
|  | |  |
| Today, I performed comprehensive testing of the APIs and endpoints, creating detailed test scenarios using Postman. These tests verified that all functions operate as expected. While finalizing the service classes, I added additional try-catch blocks, especially in the DocumentService class, which handles PDF fill operations. Given that the PDF fill process is lengthy, detailed step-by-step monitoring and error detection were necessary. In this context, I verified the JSON data reading process and ensured that the KeyValuePair class was correctly interpreting the data by printing and reviewing its entire content in the terminal.  Moreover, I incorporated try-catch blocks during file opening and fill operations to handle potential errors more effectively. Specifically, I added try-catch blocks to prevent the ArrayList used to store KeyValuePair objects from exceeding its capacity, thus enhancing the performance and reliability of this structure.  I also improved error message content to be more descriptive and user-friendly. Various optimizations were made to the code to enhance performance and provide a better user experience. These additions and adjustments have made the error management processes of the services more robust and reliable. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ:13.08.2024** | | **KAPSAMI:** **Controller Configurations** |
|  | |  |
| Today, as we approached the final stages of our project, I made some adjustments to the controller classes. I added try-catch blocks to improve error handling and completed the error checks. Notably, I realized that we were only using POST requests to operate the services. Therefore, I added DELETE requests for the Word service, PDF service, Document service, and ConversionWordPdf service to align with our architecture.  Additionally, I noticed that the GET requests we had added for the Word and PDF services were returning all files. To address this, I customized the GET requests to return only the files with the requested extensions. These GET requests were configured within the model package, connected to the FileInfo class, ensuring that the response body is now returned from a single source.  Finally, I reviewed all controller classes and added ResponseEntity to the response outputs. This change ensured that each response includes the URL and status code, making the API interactions and error management clearer and more structured. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ: 14.08.2024** | | **KAPSAMI:** **GitHub and Branch Management** |
|  | |  |
| Today, my team and I reviewed and tested the project in its final state. We pulled the latest changes into the master branch via GitHub and corrected errors in the dependencies section, ensuring we used the most up-to-date versions. We resolved conflicts found in our branches and fixed a few minor issues. Additionally, we conducted comprehensive tests on all the project's endpoints on the master branch, verifying that all API functions work correctly and the system's reliability.  We noticed that different file size limits were set in the properties section for file uploads across the project. We updated these limits to a uniform 3 MB for consistency and applied this change to the master branch.  Finally, we prepared the project for presentation to the project manager by finalizing it and preparing the necessary template files for the demo. These preparations will ensure that we present the project effectively in tomorrow's presentation. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |

|  |  |  |
| --- | --- | --- |
| **YAPILAN İŞİN;** | | |
| **TARİHİ: 15.08.2024** | | **KAPSAMI: Project Closure** |
|  | |  |
| On the final day, all developments were completed, and the project was successfully closed. Final tests were conducted, and last-minute adjustments before deployment were reviewed. We presented our project, confirming that all services functioned seamlessly. Our project manager indicated that the project could be used within the company and provided insights on potential future enhancements if it were to be adopted. Additionally, the project manager offered feedback on areas where improvements could be made and pointed out some minor deficiencies. The project was successfully completed. | | |
| **Stajımı bu firmada yaptım.**  **Staj Yapanın İmzası** | **Staj Yeri Yetkilisinin**  **Adı, Soyadı, İmzası, Firma Kaşesi** | |
|  |  | |