Preface :

The Palestinian Archive Society for Documentation (PASD) project is a student-led initiative focused on preserving and documenting the rich history, culture, and heritage of Palestine and its diaspora. Grounded in a multidisciplinary collaboration between the Architecture and Computer Engineering departments at Al-Quds University, the project aims to create a dynamic and comprehensive archival resource.

This project leverages modern archival preservation techniques and digital technologies to safeguard historical records and cultural artifacts. By organizing and digitizing these materials, PASD ensures that the memories and legacies of Palestinian people remain accessible and relevant for present and future generations.

Through this documentation, PASD not only reinforces Palestinian resilience but also provides a living resource for education, research, and cultural engagement. This project serves as a cornerstone for the greater purpose of safeguarding and sharing the Palestinian legacy.

## Abstract \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Palestine has a rich history documented through numerous archives that capture its architectural and cultural heritage from various periods. However, there is a major gap in the preservation and documentation of modern Palestinian heritage, especially after 1950, which remains an essential but underrepresented part of our cultural identity. This project aims to develop a National Digital Archive for Modern Palestinian Heritage, dedicated to recording and preserving modern architectural works, cultural expressions, and other historical assets from this period.

The archive will focus on protecting these more recent elements of our heritage, which are at risk of being lost due to a lack of proper attention. By leveraging advanced technology, this project will not only preserve these assets but also raise global awareness about the significance of this important part of Palestinian identity.

## System audience:

PASD's target audience includes researchers and historians studying Palestinian architectural heritage and cultural preservation, architects and architecture students interested in historical building designs, cultural heritage organizations focused on protecting and promoting Palestinian cultural identity, municipal and government entities involved in urban planning and heritage protection, the general public, and the Palestinian diaspora passionate about Palestine.

Conclusion:  
The Palestinian Archive Society for Documentation (PASD) project is a student-led initiative dedicated to preserving Palestine's rich history, culture, and heritage. This project is a multidisciplinary collaboration between Al-Quds University's Architecture and Computer Engineering departments, with the purpose of developing a dynamic and comprehensive archive resource.

This project protects historical records and cultural objects through the use of current archival preservation techniques and digital technology. By compiling and digitizing these artifacts, PASD ensures that Palestinian experiences and legacies are accessible and relevant to current and future generations.

**Chapter 1**

* 1. Introduction

In every part of our cities and villages, we come across structures with distinctive architectural designs. However, only a few people are aware of the true significance of these architectural masterpieces, which the occupation wishes to destroy and steal from our history.  
  
This is where our initiative, PASD, comes in. We hope to document these buildings and their architects in partnership with students from the Department of Architecture. The students collect all pertinent information about the structures and their creators, as well as unique photos, which we then post on our website and archive. This ensures that these priceless items of Palestinian heritage are maintained and available to anybody interested in safeguarding and learning about Palestinian history.

The website not only exhibits these buildings in a straightforward and visually appealing manner, but it also allows visitors to examine every aspect by viewing the photographs in 360-degrees. This ensures that these priceless items of Palestinian heritage are maintained and available to anybody interested in safeguarding and learning about Palestinian history.

## 1.2 Scope:

“PASD” aims to document and preserve Palestinian architectural heritage through the creation of a comprehensive digital archive of historical and modern buildings. The scope includes collecting accurate information about buildings and their architects, as well as capturing high-quality images from different perspectives. The project also seeks to present this information through a website that offers an interactive experience for users, such as viewing images in 360-degree technology to explore architectural details. The archive will cover buildings in major Palestinian cities and villages, with a particular focus on the modern era after 1950. Additionally, the project aims to establish sustainable partnerships with academic institutions, researchers, and architects to expand and ensure the continuity of the archive.

## Chapter 2

**Functional Requirements:**

**1. Building Documentation:**

-The website will display detailed information about each building, including the building’s name, location, architect, year of construction, and a brief history.

-High-quality images of buildings will be showcased, and users will have the option to view the images in 360-degree mode for an immersive experience.

**2. Search Functionality:**

-Users will be able to search for buildings by name.

-The users can look for buildings by city, enabling them to explore buildings within specific geographical locations.

-Users can also search for buildings associated with a particular architect, making it easier to find works by specific professionals in Palestinian architecture.

**3. Interactive Map:**  
  
-The website will include an interactive map that shows the geographical locations of documented buildings in Palestine.  
  
- Users can view extensive information about each building by clicking on the map markers.

**4. Content Management System (CMS):**

-Admins will have access to a CMS where they can add, update, or delete building records, images, and associated data.

-The CMS will allow admins to manage user accounts and monitor user activity on the website.

**6. Multilingual Support:**

-The website will support both Arabic and English to cater to a wider audience, ensuring accessibility for both local and international users.

**7. User Contributions:**

-Users will have the ability to **share an archive** with PASD through the **"Support Us"** page.

-They can upload documents, images, or historical information to contribute to the preservation of Palestinian heritage.

-Users can send an **email to the admin** directly from the Support Us page to ask questions, submit information, or offer support.

**8. Archival Preservation:**

The website will ensure that all **building information and images** are stored in a secure and organized manner, preserving the data for future generations.

| **HTTP Method** | **Node.js Example** | **React Example** |
| --- | --- | --- |
| **GET** | app.get('/api/items', (req, res) => {...}) | fetch('/api/items') or axios.get('/api/items') |
| **POST** | app.post('/api/items', (req, res) => {...}) | fetch('/api/items', { method: 'POST', body: JSON.stringify(data) }) or axios.post('/api/items', data) |
| **PUT** | app.put('/api/items/:id', (req, res) => {...}) | fetch('/api/items/1', { method: 'PUT', body: JSON.stringify(data) }) or axios.put('/api/items/1', data) |
| **DELETE** | app.delete('/api/items/:id', (req, res) => {...}) | fetch('/api/items/1', { method: 'DELETE' }) or axios.delete('/api/items/1') |

**Development Requirements:**

To guarantee a top-notch, scalable, and secure digital archive platform, PASD development calls for a blend of technology, tools, and human resources.

1. Technical specifications  
A. Development of Frontends:  
  
React.js is the framework; Tailwind CSS is the UI library.  
-Interactive Features: GeoJson (for maps) and Three.js (for 360-degree photos)  
-API Requests: Retrieve  
  
B. Development of the Backend:  
  
-Server Framework: Express.js and Node.js  
-MongoDB, a NoSQL database for scalability and flexibility  
-Authentication: JWT (if user access control is applied) for user authentication  
  
C. Collaboration & Version Control:  
  
-GitLab or GitHub for teamwork and code versioning

**Study of Feasibility**  
By creating a digital archive, the Palestinian Archive Society for Documentation (PASD) seeks to record and save Palestinian architectural history. To ascertain the project's viability, the feasibility study assesses its technical, financial, operational, legal, and timetable viability.  
  
**1. Technical viability**  
The PASD website was created with the help of contemporary web technologies, guaranteeing scalability, security, and performance (MongoDB for the database, and React.js for the front-end). The assistance provided by the website:  
  
A digital archive that can be searched, featuring thorough descriptions of buildings and excellent photos.  
seeing images in 360 degrees to provide them a fully immersive experience.  
An interactive map that shows the locations of buildings.  
Arabic and English are supported in multiple languages to reach a larger audience.  
cloud-based infrastructure for security, scalability, and backups.

**Technical Risks:**-Image archives require a lot of storage space.  
-Possible difficulties in combining 360-degree photos and the interactive map.  
-risks to cybersecurity and data security.  
  
**solutions :**  
The answer is to effectively handle big datasets by utilizing cloud storage services.  
putting strong security measures in place, such as frequent backups and SSL encryption.  
carrying out thorough testing and debugging before to launch.

2. Economic Feasibility  
The economic viability of the project weighs the advantages and disadvantages.  
  
  
- Estimated Costs:   
  
Website development, database setup, hosting, UI/UX design, and security are all included in the $21,000–$41,000 one-time development costs.   
Expenses for hosting, upkeep, content updates, and security monitoring range from $5,000 to $12,000.   
  
- Expected Benefits:   
  
Cultural and Historical Preservation: Prevents the erasure of Palestinian history.   
Impact on Education: Offers a resource for architects, students, and scholars.   
Global Awareness: Makes Palestinian heritage more accessible everywhere.   
Possible Sponsorship & Grants: Funding may be available through collaborations with educational institutions, non-governmental organizations, and cultural heritage groups.   
  
Benefit-Cost Analysis:   
  
Even though the project requires a large upfront investment, the long-term advantages in protecting Palestinian history outweigh the expenses. To pay for upkeep costs, the initiative might raise money through donations and sponsorships.

**Operational Feasibility**

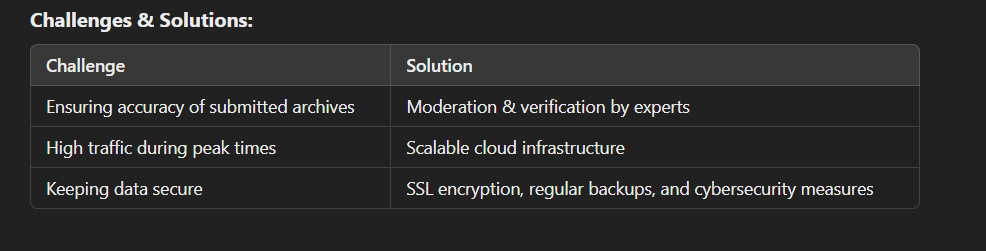
A group of developers, architects, and academics will oversee the project, making sure that data gathering, archiving, and website upkeep are done effectively.

Key Considerations:

Data Collection & Verification: Working with historians, architecture students, and historical organizations to ensure correct documentation.

Administration of a Website: Using a content management system (CMS) will make updating and archiving your website simple.

Contributions from Users: Building archives submitted by users on the website will be examined prior to publication.



**Feasibility Study**

The **Palestinian Archive Society for Documentation (PASD)** aims to document and preserve Palestinian architectural heritage through a digital archive. The feasibility study evaluates the project's **technical, economic, operational, legal, and schedule feasibility** to determine its viability.

**1. Technical Feasibility**

The PASD website will be developed using modern **web technologies (React.js for the front-end and MongoDB for the database)**, ensuring scalability, security, and performance. The website will support:

* A **searchable digital archive** with high-quality images and detailed descriptions of buildings.
* **360-degree image viewing** to provide an immersive user experience.
* An **interactive map** to visualize building locations.
* **Multilingual support (Arabic & English)** to cater to a wider audience.
* **Cloud-based infrastructure** for scalability, backups, and security.

**Technical Risks:**

* High storage demand for image archives.
* Potential challenges in integrating the interactive map and 360-degree images.
* Cybersecurity threats and data protection.

**Solution:**

* Using **cloud storage services** to manage large datasets efficiently.
* Implementing **robust security protocols**, including SSL encryption and regular backups.
* Conducting extensive **testing and debugging** before launch.

**2. Economic Feasibility**

The project's economic feasibility evaluates its costs and benefits.

**Estimated Costs:**

* **One-Time Development Costs:** $21,000 - $41,000 (Website development, database setup, hosting, UI/UX design, security).
* **Annual Operational Costs:** $5,000 - $12,000 (Hosting, maintenance, content updates, security monitoring).

**Expected Benefits:**

* **Cultural & Historical Preservation:** Protects Palestinian heritage and prevents erasure.
* **Educational Impact:** Provides a resource for researchers, students, and architects.
* **Global Awareness:** Increases accessibility to Palestinian heritage worldwide.
* **Potential Sponsorship & Grants:** Partnerships with academic institutions, NGOs, and heritage organizations could provide funding.

**Cost-Benefit Analysis:**

While the project involves **significant initial investment**, its **long-term benefits** in preserving Palestinian history outweigh the costs. The project can secure **sponsorships and donations** to cover maintenance expenses.

**3. Operational Feasibility**

The project will be managed by a **team of developers, architects, and researchers**, ensuring that data collection, archiving, and website maintenance are efficiently handled.

**Key Considerations:**

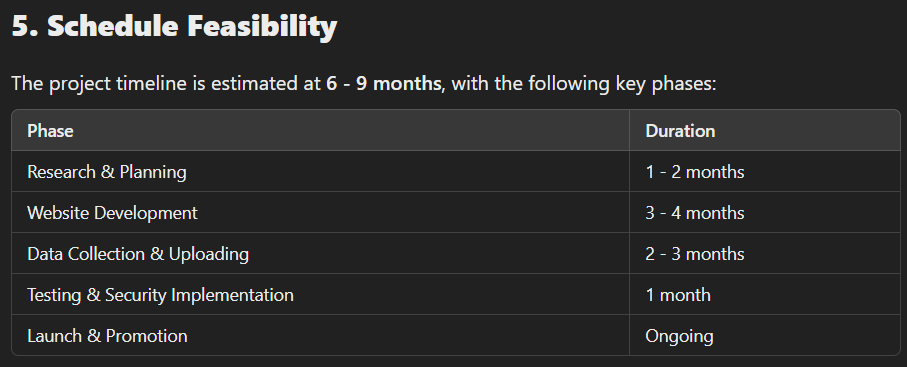
* **Data Collection & Verification:** Ensuring accurate documentation through collaboration with **architecture students, historians, and heritage organizations**.
* **Website Administration:** A **content management system (CMS)** will allow easy updates and archiving.
* **User Contributions:** The website will allow users to **submit building archives**, which will be reviewed before publication.

**Challenges & Solutions:**

| **Challenge** | **Solution** |
| --- | --- |
| Ensuring accuracy of submitted archives | Moderation & verification by experts |
| High traffic during peak times | Scalable cloud infrastructure |
| Keeping data secure | SSL encryption, regular backups, and cybersecurity measures |

**4. Legal Feasibility**

The project complies with:  
  
Intellectual Property Laws: Making sure that photos and architectural information are utilized with the appropriate authorization.  
  
Data Privacy Regulations: protecting archives and data submitted by users.  
  
Using hosting services that permit digital preservation and content exchange is subject to hosting regulations.  
  
The answer is to ask pertinent sources for permission to use archival documents.  
putting privacy and terms of use policies into effect to safeguard user information.



**In conclusion**  
The PASD project is technically feasible, fiscally justified, operationally controllable, complying with the law, and deliverable within a reasonable timeframe. The project is a worthwhile and essential endeavor since its long-term effects on conserving Palestinian history outweigh its expenses.

**Values Added by the Project:**  
  
Creating a digital archive that records and conserves Palestinian architectural heritage is the goal of the Palestinian Archive Society for Documentation (PASD) project. For developers and users alike, this effort offers substantial technical, cultural, and educational advantages.  
  
**-Benefits for Education:**  
-Increases understanding of digital documentation and archival preservation strategies.  
  
-enhances knowledge of front-end technologies (React.js), database administration (MongoDB), and web programming.  
  
-enhances proficiency with interactive media and image processing, including GIS mapping and 360-degree picture viewing.  
  
-gives practical expertise with content management systems (CMS) and data organizing.

-enhances research and project documentation abilities, which are critical for both academic and professional development.

**Impact on Culture and Society:**  
-Preserves Palestinian architectural history by guarding against cultural appropriation and erasure.  
-Creates a consolidated and accessible archive for academics, students, architects, and historians.  
  
-Uses a user-friendly digital platform and linguistic support to increase awareness of Palestinian heritage worldwide.  
-Promotes community involvement by permitting user input and past entries.

**Project Management:**  
The creation and effective deployment of the PASD digital archive depend on a methodical approach to project management. Technical research, software development, data collection, and stakeholder participation are all part of the development process.  
  
**Phases of Project Development:**  
1. Research & Planning: Examining web technologies, database architecture, and architectural documentation techniques.  
2. System Design & Development: developing the architecture of the website, putting the CMS into place, and incorporating the 360-degree image viewer and interactive map.  
3. Data Collection & Archiving: Working together with historians, scholars, and architecture students to collect authenticated historical information and photographs.  
4. Testing and Security Implementation: Guaranteeing system security, performance optimization, and data integrity.  
5. Deployment & Continuous Improvement: Starting the website and making sure it is updated and maintained over time.

**Communication plan:**   
Efficient communication among team members, managers, and stakeholders guarantees project effectiveness and goal alignment. The PASD project team communicates on a regular basis by:  
  
  
-Convening to think and solve problems at Al-Quds University.   
-For official talks and document exchange, use email correspondence.   
-Reviewing the design in meetings with the college dean and architecture students.  
---------------------------------------------------------------------------------------------------------------------------

**Conclusion and Recommendations:**

The PASD project is a trailblazing effort that uses contemporary technologies to record, conserve, and disseminate Palestinian architectural history. Future generations will be able to access Palestinian history thanks to the project's creation of an interactive, user-friendly digital archive.  
  
**Key Accomplishments:**  
 -Effectively incorporates state-of-the-art web technologies (MongoDB, React.js) to create a scalable and effective solution.  
-offers an interactive map and 360-degree photos for a more engaging experience.  
-creates a digital repository for Palestinian architectural documentation that is organized and safe.  
-permits user involvement, enabling the general public to make contributions to the archive.  
  
**Recommendations for Future Enhancements:**-Extend the scope of the archive to encompass historical records, architectural designs, and oral histories.  
-Improve AI's capacity to analyze previous data and tag images automatically.

-Create a mobile app to make the archive more accessible.  
-For future expansion and finance, fortify alliances with foreign research institutes and cultural institutions.  
  
By putting these suggestions into practice, PASD will keep developing as a useful tool for conserving Palestinian history and culture while encouraging participation and awareness around the world.

# Chapter 3:

**3.1 Introduction**

The success of the Palestinian Archive Society for Documentation (PASD) project depends heavily on the implementation of a functional and efficient system. This chapter outlines the system’s functional and non-functional requirements, which are essential for the development of the website. The functional requirements specify the capabilities the system must have to serve its target audience, while the non-functional requirements describe the system's performance attributes, such as usability, reliability, and security. This section provides an overview of the key requirements and sets the foundation for the detailed design and development of the PASD website.

**3.2 Context Diagram**

The PASD website's context diagram provides a high-level overview of the system and shows how it communicates with other parties. The PASD system is depicted as the main element in the figure, along with its connections to databases, users, administrators, and other systems.

================

**3.3 Use-case Diagram**

The different user-system interactions are depicted in the PASD website's use-case diagram. Every use case illustrates a particular task or action that the administrator or user is capable of carrying out.

**3.4 Functional Requirements Description**

The fundamental functions that the PASD system must support in order to satisfy the demands of its administrators and users are outlined in the functional requirements. The essential functional criteria are as follows:

Building Documentation:  
-Details about each building, including its name, location, architect, year of construction, and historical value, must be shown on the website.  
-Users should have access to high-quality photos of every building. For an immersive experience, 360-degree photos must be supported by the system.  
  
Search Functionality:  
Buildings must be searchable by a number of parameters, such as architect, city, and building name.  
  
Interactive Map:  
-An interactive map showing the locations of the buildings that have been documented must be included on the website.  
  
-To display comprehensive information, users should be able to click on the location of a building on the map.

Content Management System (CMS):

-Building records, photos, and metadata must be manageable and updateable by administrators using a content management system (CMS).  
-The CMS must give administrators the ability to accept or reject user-submitted archives.  
  
Multilingual Support:  
To enable users from various geographical locations to engage with the system, the website must support both Arabic and English.  
  
Contributions from people:  
-The "Support Us" page must enable people to upload their own historical data, records, or photos.  
-Before submissions are made public, administrators must review and approve them.  
  
Archival Preservation:  
The system must guarantee that all information, including text, photos, and metadata, is safely kept in an orderly and future-proof manner.

**3.5 Non-Functional Requirements Description**

**1. Responsive Design:**

-The website will be fully responsive, ensuring that it functions properly across different devices (desktop, tablet, and mobile).

**2. Security & Data Protection**

-Secure "SSL" encryption is used by the system for all communications.   
  
-Archives donated by users must first undergo moderation before being made public.   
  
-Secure authentication is necessary for administrative access.

**3. Dependability & Performance:**  
-PASD can load pages within 3 seconds under normal conditions.  
  
-At least 500 users can utilize the system at once without experiencing any performance issues.  
  
-Large amounts of text and graphic data are readily supported by the database.

**4.** **Scalability and Maintainability:**  
-To enable future growth, the website was constructed with scalable technologies (MongoDB and React.js).  
  
-In upcoming updates, the system facilitates seamless incorporation of new features.

**5. Accessibility & Usability:**   
  
-The website is responsive and functions flawlessly on mobile, tablet, and desktop computers.  
  
-The UI is easy to use and requires little training.  
  
Arabic and English are among the languages that the system supports.

**3.6 Conclusion and Recommendations**

The PASD system's functional and non-functional needs have been covered in length in this chapter. These specifications guarantee that the website will serve as a scalable, dynamic, and user-friendly platform for conserving Palestinian architectural history. While the non-functional criteria make sure the system operates effectively, safely, and is still available to a large audience, the functional requirements concentrate on essential features like creating documentation, search capabilities, and user contributions.  
  
The following are suggestions for upcoming improvements:  
  
1. Including more sophisticated filtering and search features.  
2. Extending the archive's reach to encompass additional types of cultural heritage.  
3. Updating the system frequently to add new functionality and enhance efficiency.

## Chapter four System Design and Development

**4.1 Introduction**

The design and development of the Palestinian Archive Society for Documentation (PASD) system are the main topics of this chapter. It goes over the architectural framework and intricate diagrams that are necessary to comprehend how the system will be constructed, how its parts work together, and how user requests are handled. The system architecture guarantees that PASD will meet all functional and non-functional needs while operating effectively and efficiently. To direct the development and give a clear understanding of the structure and functionality of the system, it contains a number of UML diagrams, an entity-relationship diagram, and an input/output design.

**4.2 Class Diagram**

………..

………..

……………

………….

**4.3 Sequence Diagram**

The sequence diagram shows how system components work together over time to accomplish a particular task. A sequence diagram for "User Viewing a Building's Information," for instance, would display the following order:  
  
1. User: Requests to view the details of a building.  
2. System: Looks up building information in the database.  
3. Database: Provides information on the building, such as pictures and metadata.  
4-System: Delivers the information to the user interface for presentation.  
5. User: Examines the data, which includes pictures and descriptions.  
  
By illustrating the data and control flow throughout the system, the sequence diagram makes it evident how several objects work together to satisfy a user's request.

**4.4 Entity Relationship Diagram (ER-D)**

entity relationship diagram (ER-D) is a data model for describing a pasd database in an abstract way.

**4.5 Activity Diagram**

The PASD system's user activity workflow is shown in the activity diagram. It illustrates the actions a user or administrator takes to finish particular tasks.

The activity diagram helps visualize the flow of activities and decisions within the system, ensuring that all necessary steps are followed to complete tasks.

**4.6 System Interface (Input/Output Design)**

==

==

==

**4.7 Conclusion and Recommendations**

The system design and development process for the PASD website has been explained in this chapter, with particular attention paid to important elements such the input/output designs, ER-D, class diagram, sequence diagram, and activity diagram. These diagrams give the development team a clear framework by describing how the system should function and communicate with users, as well as how data should be kept and retrieved.

**Recommendations for further development include:**

-adding more features, like sophisticated search and filtering.  
-creating a mobile-friendly website to increase accessibility.  
-enlarging the user interface to improve design and usability.

## Chapter five Coding and Implementation

**5.1 Introduction**

Using the right programming languages and tools, the project's coding and implementation phase aims to turn the concept into a functional system. The technical components of creating the website are covered in this chapter, including database integration, programming language selection, development environment setup, and implementation of essential features. In order to facilitate the seamless documentation of Palestinian architectural art, the system must be responsive, effective, and easy to use.

**5.2 Coding Programming Languages**

We have selected cutting-edge web development technologies that guarantee responsiveness and usability for the creation of the website for the Palestinian Archive Society for Documentation (PASD). For front-end development, CSS, and JavaScript are the main programming languages used. Because of its capacity to effectively generate dynamic user interfaces and offer an improved user experience, React.js has been chosen. The system handles requests and API routing on the back-end using Node.js and Express.js. The project incorporates MongoDB, a NoSQL database, to store architectural data, including building specifications and photos, in order to guarantee flexibility and scalability.

**5.3 Database System**

MongoDB, a NoSQL database, was selected for this project because of its capacity to effectively manage substantial amounts of unstructured and semi-structured data. The flexible schema design of MongoDB makes it ideal for documenting architectural information since it makes it simple to store many kinds of content, including metadata, photos, and descriptions. Each building is represented as a document with attributes like name, location, image URLs, and a thorough description. The system uses MongoDB's collections and documents to arrange the data. The scalability and performance of the database facilitate future expansion and allow for smooth front-end interface integration.

**5.4 Establishment of Development Environment**

The development environment for the PASD website is established using industry-standard tools to ensure efficiency and consistency. The Node.js runtime is used for the back-end server, while React.js is utilized for the front-end development. For code editing and version control, Visual Studio Code and Git are employed. The project uses npm (Node Package Manager) to manage dependencies, while MongoDB Atlas provides a cloud-based solution for hosting the database. The development environment is configured to enable smooth collaboration among the team members and facilitate the implementation of features such as the search functionality, document display, and image handling.

**5.5 Stored Procedures**

Although NoSQL databases, such as MongoDB, do not employ relational databases' conventional stored procedures, the system uses a number of backend processes to efficiently manage data. These include retrieval, update, delete, and data insertion activities, all of which are managed by JavaScript-written proprietary backend methods. These operations are carried out through Express.js-created API routes, which allow the front-end to communicate with the database. For instance, the system has features to update building metadata, add new document entries, and get architectural data based on search queries. All data operations are carried out safely and effectively thanks to the backend logic.

**5.6 Conclusion and Recommendations**

The PASD website has a strong basis thanks to the coding and implementation phase, which successfully satisfies the requirements of recording Palestinian architectural history. A scalable, responsive, and user-friendly system is guaranteed by the selection of React.js for the front end and MongoDB for the back end. In order to maintain the website's speed and accessibility going ahead, it is advised to regularly evaluate and improve the system for performance, especially when managing big data sets. The functionality and security of the system could also be improved by include features like user authentication and sophisticated search filters.

## Chapter 6, System Testing and Implementation

**6.1 Introduction**

A critical step in the development process is system testing, which makes sure the program works as intended and satisfies the requirements. The testing approach used for the PASD website is covered in this chapter, along with the several kinds of testing that were done to assess the functionality, performance, and usability of the system. We make sure the website is dependable, effective, and able to manage various use cases by conducting extensive testing. The testing phase's findings and conclusions offer important information about the system's general quality and any areas that require improvement.

**6.2 System Testing Plan**

The methodology and strategy for verifying the PASD website are described in the system testing plan. To make sure that the front-end and back-end components operate as intended, the testing phase consists of a variety of tests. Among these tests are:  
  
-Functional Testing: To confirm that the system satisfies all functional requirements, including database interface, data retrieval, and architectural information display.  
  
-Usability Testing: To evaluate the website's user interface (UI) and make sure people can easily navigate and understand it.  
  
-Performance testing: To assess how well the website functions under various load scenarios and make sure it can support numerous users without experiencing any problems.

-Security testing: To find possible weaknesses in the system and make sure that information is shielded from unwanted access.  
  
-Compatibility testing: To make sure the website works correctly on various devices and browsers.

**6.3 System Integration Test**

The main goal of the system integration test is to assess how well the various PASD website components function as a whole. This test makes sure that the database and back-end server communicate with the front-end interface without any problems. Important domains for integration testing consist of:

-**API Integration**: Testing the communication between the front-end (React.js) and the back-end (Node.js/Express.js), ensuring that data is retrieved, processed, and displayed correctly.

-**Database Integration**: Ensuring that the database (MongoDB) correctly stores and retrieves architectural data, such as building descriptions, images, and metadata, without errors.

-**Third-Party Services Integration**: Verifying that any external services (such as cloud hosting for the database or image storage) are properly integrated into the system and function as expected.

**6.4 Testing Plan Results**

An assessment of how effectively the PASD website achieved its goals may be seen in the testing phase data. The results of the system tests can be summed up as follows:

1. **Functional Testing Results**: All functional requirements were met, including the successful retrieval and display of architectural data, smooth navigation between pages, and proper handling of user requests.

2. **Usability Testing Results**: The website received positive feedback regarding its intuitive interface and ease of use. Users were able to easily navigate the archive and find the information they were looking for.

3. **Performance Testing Results**: The website performed well under normal load conditions, with fast page load times and smooth transitions between pages. However, some performance improvements are recommended for handling higher traffic volumes.

4. **Security Testing Results**: No major security vulnerabilities were found, although additional security measures, such as data encryption for sensitive information, could further enhance the system's protection.

5.**Compatibility Testing Results**: The website is compatible with modern browsers (Chrome, Firefox, Safari, etc.) and mobile devices, with consistent appearance and functionality across platforms.

* 1. **Conclusion and Recommendations**

The PASD website is operating efficiently, according to the testing phase, and no serious problems have been found. Nonetheless, a number of suggestions for enhancement have surfaced:

**1.** **Performance Optimization**: Although the website functions well in typical circumstances, load times during periods of high traffic might be shortened by streamlining the back-end code and putting caching techniques in place.

**2.** Increase the number of languages and translations so that users from different backgrounds and cultures may readily access the website.   
**3.** To improve the user experience, provide more features like bright and dark modes.

**In conclusion**   
The PASD website satisfies its goal of recording Palestinian architectural heritage and is a well-executed application. It can develop into an even more reliable and scalable platform with a few tweaks and further testing.

## **Chapter Seven Conclusions and future works**

**7.1 Conclusions**

In order to combat the threat of cultural asset theft, the Palestinian Archive Society for Documentation (PASD) works to conserve and record Palestine's architectural legacy, especially those structures built after the mid-1990s. For the benefit of future generations, the website offers a platform for gathering, organizing, and showcasing photos and details about these structures. Constructed using React and MongoDB, the system guarantees a seamless user experience, convenient data access, and an easy-to-use interface. The website guarantees that anybody can freely browse the collection and learn about Palestinian architectural history, even in the absence of a registration system.

**7.2 Future Works**

By adding more languages and translations in the future, PASD hopes to improve the user experience and make the archive available to a larger audience. Including tools like interactive maps and sophisticated search capabilities will increase user engagement even more. Additionally, adding a content management system (CMS) would make it easier for contributors and architecture students to upload and maintain data.

**7.3 References**

* Learning React: Modern Patterns for Developing React Apps(Alex Banks ,Eve Porcello)
* React Up & Running: Building Web Applications(Stoyan Stefanov)
* <https://react.dev/reference/react>
* <https://learn.mongodb.com/>
* <https://www.w3schools.com/mongodb/>