

PROJECT TITLE	Residents Management System		
COMPANY NAME	Southern Management Corporation	LOCATION	Prince George County, MD
PROJECT MANAGER	Sakshi Sharma	DATE SUBMITTED	7th March, 2020
AUTHOR	Pratik Pandey, Manasvi Prasad, Manas Mishra, Yash Srivastava	VERSION	1.0.0
PROJECT BEGIN DATE	10th February, 2020	END DATE	16th April, 2020

Table of Contents:

Serial No.	Name of the Topic	Page No.
I	Introduction	4
II	System Planning Phase Report	5
1	Client and Industry Background	5
1.1	Purpose of the Project	5
2	Scope of Work	5
3	Project Feasibility Study	8
3.1	Technical feasibility	9
3.2	Cost and Benefit Feasibility	10
3.3	Organizational Feasibility	12
4	Project Plan	13
4.1	Location of Project	13
4.2	Planning Phase	13
4.3	Analysis Phase	13
4.4	Design Phase	14

4.5	Implementation Phase	14
5	Conclusion	14
III	System Analysis Phase Report	15
1	Fact Finding	15
2	Proposed Data Flow Diagrams	17
2.1	Context Diagram	17
2.2	Level 0 Diagram	19
2.3	Level 1 Diagram	23
2.3.1	Level 1: Process 1 Diagram	23
2.3.2	Level 1: Process 2 Diagram	26
2.3.3	Level 1: Process 3 Diagram	27
3	Proposed ER Diagram	32
4	Process Flow Diagram	35
5	Conclusion	35

INTRODUCTION

This improvement project for Southern Management Corporation, one of the largest property management firms in the Mid-Atlantic region is aimed to benefit the firm on multiple levels. Every penny investment in the project has been made sure to reap benefits that align with the expansion strategies of the firm and contribute towards the long-term goals of the firm by upgrading the Residential management system. The hope is to develop a large reliant network of property partners and customers by improving the quality of service and the experiences of the involved parties.

The first section of the document consists of the Statement of Work which entails the planning phase of the project and consists of the purpose of the project, along with a detailed scope and a sample use case diagram. It also has a thorough feasibility analysis, which analyzes the project against parameters like cost, the benefits reaped out of the project and comparison between two technical solutions, the cloud-based and hybrid as components of the technical feasibility analysis of the project. This also consists of the Project Plan and its phases like Planning, Analysis, Design and Implementation for the complete duration of the project.

Assuming the project got approval, we proceed with the second phase and document the Analysis phase in detail. The Analysis phase has Data-flow Diagrams(DFD) to represent the flow of data of the processes we improve upon and functions we plan to add and to describe the outputs and inputs of the involved entities and processes in the Resident Management System. The first DFD is a Context diagram, to depict in a high-level manner how the improved system will interact with the external entities. This is followed by a Level 0 diagram, which is a more granular version of the Context diagram, splitting the system into major processes and depicting their interaction with the same external entities, impacted by the process improvement project. The following diagrams are the Level 1 diagrams, that are finer depictions of the Level 0 diagram into 3 separate diagrams, where each process from the Level 0 diagram is depicted separately in a very granular fashion. The following components of the document are an Entity Relationship Diagram and a Process Flow diagram. The ER diagram represents each involved entity, their attributes, the relationship and cardinality that these entities share. The aforementioned Process flow diagram depicts a sample process of one of the 5 processes that are being added or enhanced through the project. The process flow diagram shows the flow of information between the involved external entities through decision nodes and how the course might change due to the changing decisions.

SYSTEM PLANNING PHASE REPORT

STATEMENT OF WORK

1. Client and Industry Background

The project is an enhancement that we have proposed for Southern Management Corporation. Southern Management Corporation is the largest privately owned residential property-management company in the Mid-Atlantic region and owns and manages 77 apartment communities with approximately 25,000 apartment homes, several commercial office buildings, hotels and conference centers, ski resorts throughout the Baltimore/Washington area and in Pennsylvania. The goal of our project is to upgrade the existing residential services system in the College Park region. Being residents of a property of southern management, through our first-hand experience, we were able to identify several faulty functions or complete lack of them in the current system. Therefore, our aim is to upgrade this system to address these issues and come up with an efficient information system for better transparency and customer service.

1.1 Purpose of the Project

Since we truly understand the pain points of the users of the system and know for a fact that there is incredible scope for improvement as the Southern management resident services system could be improved with simple process changes. The project will benefit all parties involved and on multiple levels and it is designed to cover all bases in terms of both tangible and intangible benefits.

2. Scope of Work

There is significant scope for improvement and of the several functions, we identified five to be implemented, which we felt should have the most priority.

Provision for online in-house transfer request: The Resident management system still lacks an online in-house transfer request portal. Users who are looking to move apartments due to any issues with existing ones, start to look at other properties, because Southern management still doesn't allow online requests, while other properties allow them to go through everything online.

Search and find potential roommates: A portal for interaction between potential and existing residents about search for potential roommates and open spots is conspicuously absent. Existing members are forced to move out when they are unable to find roommates due to lack of an interaction portal, and they have to pay the full amount of rent. Designed to improve user experience and provide customers with ease of interaction. This will also save Southern Management from losing revenue from vacant listings.

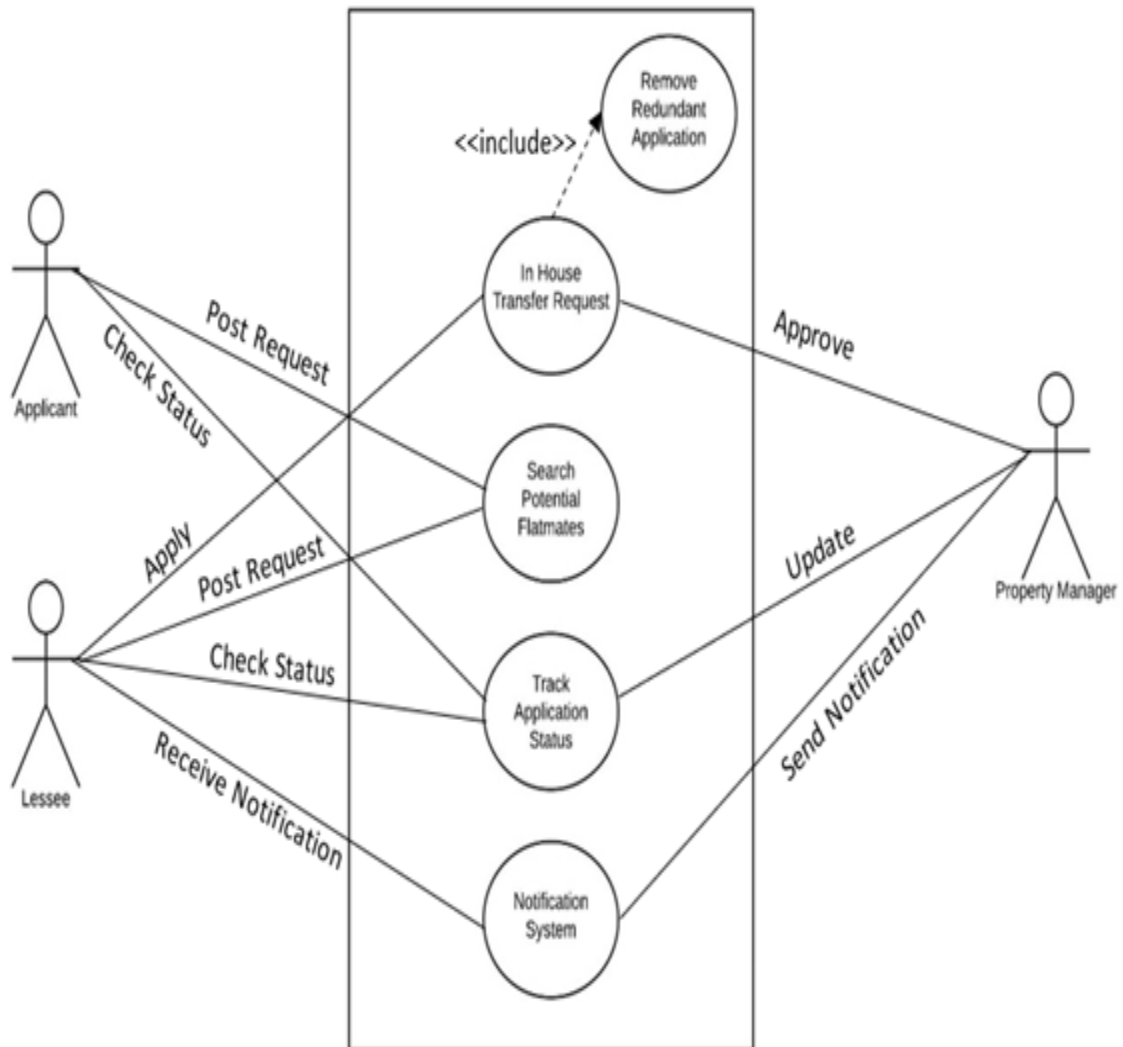
Track application status through a transparent waitlist: The waitlist is not transparent and causes mistrust between the applicants and the management. This will not affect the privacy of the applicants as we plan to show only the name of the applicants, the date they submitted the application and their standing on the waitlist, with an option to add contact details, if needed. This will help mitigate privacy issues that arise from transparency.

Improve customer notification system to provide rent alerts, service request updates etc.: We have also spotted the absence of a thorough notification system about due rent or status of service requests and any changes in the existing system.

Remove duplicate rental applications of the same apartment type: Residents and non-residents are allowed to have any number applications for the same type of apartment listing in the database. There is no restriction on this and even after finding apartments, years and years of applications are retained in the database.

Function	Business Need	User Experience
Provision for an online in-house transfer request	Reduces paperwork and data entry for the property manager	Residents can file in-house requests online without going to the resident office
Search and find potential roommates	Helps in gaining more residents and hence more houses on rent	Applicants can group up together and take apartments from the management
Track application status through a transparent waitlist	Reduces customer care calls and complaints	Applicants are made aware of their application status and waitlist
Improve customer notification system to provide rent alerts, service request updates etc.	Intimate residents for timely payments and keep them informed about requests.	Customers get timely notifications informing them regarding rent due, service requests etc.
Remove duplicate rental applications of same apartment type	Helps management in application management and house allocation.	Applicants have one common application for apartment request

Use Case Diagram



3. Project Feasibility Study:

This system proposes two technical solutions, namely, cloud-based and hybrid.

Cloud: The Southern Management System is already cloud-based. Hence, integrating the change management system is going to be much easier. To add to that, this system will provide various tangible benefits which have been discussed below in detail.

Hybrid: The alternate approach is developing a hybrid solution which would make the system more robust. Sensitive data could be stored on-premise and less critical information can be maintained on the cloud.

CLOUD-BASED	HYBRID
Web based solution	Web based solution.
Web based application hosted using in-house cloud service provided by Yardi	Web based application hosted in Southern Management data center
Data stored using in-house cloud service provided by Yardi Systems	Data stored using in-house cloud service provided by Yardi Systems

Comparisons between the two technical solutions have been made in the above table.

- Both the solutions are web-based. The differentiating factor between the two approaches is the server. In the cloud-based system, the server is cloud-based too. On the contrary, a hybrid model has the server situated on-premise as well as on the cloud. In this system, sensitive information is generally stored on-premise whereas the application is cloud-based. This usually propels the cost of hybrid systems.
- Another differentiating factor is the way data is stored using these two models. Currently, Southern Management uses a cloud-based system provided by Yardi systems. Yardi systems also provide in-house solutions for data storage and will be used in hybrid models.

- After comparing the two solutions, we have decided to go ahead with a cloud-based solution. This is because Yardi systems provide a bundled service for the same. To avoid migration costs, a cloud-based solution is ideal.

3.1 Technical feasibility

	Cloud	Hybrid
Familiarity with technology	Excellent	New
Project Size	Medium	Medium
Compatibility	High	Medium

Both the technical solutions have their own impacts on how the project pans out.

- The existing application is already cloud-based. Hence, implementing changes that are cloud-based will not require any stakeholder to familiarize themselves with the updated application. This might not be true in case of hybrid systems as the backend system might function differently and this would require familiarization from the perspective of the maintenance team.
- Since we are implementing a limited number of changes which are aimed at improving the efficiency of the system, the project is going to be relatively small in both cases. Implementing change management does not always mean that the project size is going to be small, but in this case, the project size is medium.
- Compatibility is going to be high for a cloud-based model as there would be no migrations that will take place. The new system will be based on the same architecture. However, in developing a hybrid system, compatibility might decrease when compared to a purely cloud-based system.

3.2 Cost and benefits analysis

Costs:

There are several costs involved in the execution of this change management system which can be classified into development and operational costs.

	Cloud	Hybrid
Development Costs		
Software development	\$45000	\$60000
Hardware and software	\$1500	\$2000
Operational Costs		
Cloud storage fees	\$1000	\$750
Staff training	\$1500	\$1500
Transactional costs	\$1500	\$1500

Development Costs:

- For both models, the majority of the costs are concentrated on software development. But the costs for hybrid development are much higher due to obvious reasons. Setting up the infrastructure on-premise is going to drive up the costs.
- Like software development, the cost for hardware and software would be higher for hybrid systems as cloud-based systems are much cheaper but investing in on-premises servers increase the costs to a certain extent.

Operational Costs:

- Cloud storage fees would be higher in the case of a cloud-based system.

- The training costs involved are pretty much the same in both models.
- Transactional costs for a cloud-based system would be more or less the same as hybrid systems. Both the methodologies would include planning and outsourcing but, in some cases, it could be slightly more in hybrid systems.

Benefits:

We have identified a few benefits for both the technical solutions and categorized them as tangible and intangible.

	Cloud	Hybrid
Tangible benefits		
Increased Sales and Revenue	Yes	Yes
Reduction in IT costs	Yes	Yes
Intangible benefits		
Increased brand recognition	Yes	Yes
Increased market share	Yes	Yes
Improved customer service	Yes	Yes

- A significant increase in the revenue from increased and consistent sales is observed for both the models and otherwise, through portals for flat mates, online in-house requests, reduced paperwork and reduced load on database.
- The mitigation of redundancy in applications will result in a lighter database.
- Better and transparent system will entice other property owners and the word of mouth will also bring more users and develop their network

- Increased transparency will consequently bring increased customer trust and a strong Customer base.
- Improving user experience, better and ease of usage functions will bring a stronger customer base for both property owners and the management.

3.3 Organizational feasibility

	Cloud	Hybrid
Strategic Alignment	Excellent	Excellent
Influence on Stakeholder	High	High

Based on the Cloud and Hybrid models, the organizational feasibility of the system can be measured in the following way:

1. **Strategic Alignment:** The new system enhances the existing system and adds new features to it. These new features are strategically aligned with Southern Management's goal to better manage apartment applications and applicants. The system will reduce paperwork and redundancy by digitalization of the process.
2. **Influence on Stakeholders:** The stakeholders of this enhancement are the residents, the intermediary platform, Rent Café, and Southern Management Corporation itself. The entire project management team operates under Southern Management Corporation.

The system, when implemented, will reduce the burden on Southern Management employees and simplify various processes. This in turn will save a lot of labor hours and further the company's profits. From a stakeholder's point of view, the system will provide a high ROI (Return on Investment) as the development of the system is minimal and the benefits are high for both cloud and hybrid models. Thus, the stakeholder affiliation will increase in either of the cases.

4. Project Plan:

Location of the Project

Since the development of the Southern Management portal will be handled by their vendor, Rent Café, most of the work will be done at its headquarters in Santa Barbara, CA.

Elicitation of user requirements followed by system installation and user training will happen at Southern Management's headquarters at Vienna VA.

Project Plan First Cut

The project is going to be developed using the **waterfall development methodology**. The plan entails 4 stages which takes a total of **49 days** to complete starting on 2/10/20 and goes on till 4/16/20, which are as follows:

4.1 Planning stage:

Starting 2/10/20, taking a total of **19 days** to complete.

Tasks:

- Identification of the project requirements
- Developing a systems request
- Analysis of feasibility (technical, economic and organizational)

Milestone: Discuss and propose a plan, place a system request and run technical, economic and organizational analysis and get ballpark figures by 3/5/20.

4.2 Analysis stage:

Starting 3/6/20, taking a total of **20 days** to complete.

Tasks:

- Gather Customer requirements
- Gather stakeholder requirements
- Develop, review and approve use cases

Milestone: Study customer and stakeholder data and requirements, develop use cases which are compliant with the requirements, review and approve the use cases by 4/2/20

4.3 Design stage:

Starting 3/6/20, taking a total of **10 days** to complete.

Tasks:

- Select design strategy
- Study existing systems
- Develop use scenarios and program specifications

Milestone: The design strategy will be formulated; past and existing systems will be studied and use case scenarios and program specifications will be gathered which is needed for the new system with added functionalities by 3/19/20

4.4 Implementation stage:

Starting 3/6/20, taking a total of **25 days** to complete.

Tasks:

- Program system
- Test software
- Maintain system

Milestone: The development and testing of the system will be completed in 15 days. Maintenance and support for the new system will be provided for the next 10 days until 4/16/20.

5. Conclusion:

This statement of work recognizes the opportunities and challenges of this enhancement to Southern Management, their property owners, the residents and their brand. We appreciate the opportunity to present this and hope to add value to Southern Management through this project. We believe that the project is consistent with the goals of the firm. Once the project is acceptable to the board and gets signed for approval, the project plan that has been put into place will be implemented.

SYSTEM ANALYSIS PHASE REPORT

1. Fact Findings and Requirement Gathering

The Southern Management Corporation has long been touted as the largest residential management company in the United States. However, it has had its own issues regarding the experiences of residents and non-residents using the website to perform various functions. As part of representing Southern Management, a list of functions have been identified that can improve the efficiency of the website. Out of these, five have been shortlisted based on priority. Below is a detailed description of fact findings and requirement gathering techniques for all the functions that have been identified for execution.

Function 1:

Provision for online in-house transfer request

Fact findings: In the age of online transactions, which Southern Management has incorporated pretty effectively, in-house transfer requests were still being processed offline. This was a problem for residents who didn't stay close to the residential services offices of Southern Management.

Requirement Gathering Techniques:

Interviews: Short interviews were conducted with students staying in Southern Management communities. A top-down interview structure was followed which was open-ended as the focus was on opinions, not specifics. Interviews were documented and later sent to the interviewees for clarifications, corrections and confirmations.

Function 2:

Search and find potential roommate

Fact findings: Portal to find new roommates is absent. This often leads to people surrendering their apartments unwillingly and bearing the full rent. An interactive portal to improve residents' experience and provide them with a platform to expedite the process of finding roommates has led to the creation of this requirement.

Requirement Gathering Techniques:

Interviews: Since this requirement is aimed at improving the problems of the residents, we believe that interviews were the best way to go about it. Residents were interviewed in a similar fashion to the above-mentioned function. Users (residents) were able to provide us with clear explanations.

Questionnaires: Questionnaires were floated to the residential services teams at various Southern Management offices to understand what is needed from the service end. The added functionality will require more maintenance and gathering facts and opinions from all stakeholders is going to be helpful.

Function 3:

Track application status through a transparent waitlist

Fact findings: There is a waitlist system in place but the residents don't have access to view it. This causes mistrust and there have been a few occasions on which the waitlist hasn't been used efficiently to allot homes to applicants.

Requirement Gathering Techniques:

Document Analysis: The existing system was studied in depth. The system and its design documentation created by analysts and consultants were used to collect facts. Information was classified into current and outdated to determine its accuracy. This method helped us understand the existing system well.

Function 4:

Improve customer notification system to provide rent alerts, service request updates etc.

Fact findings: The existing application is devoid of an alert system. Although it is the responsibility of the resident to set reminders to pay bills in time, a notification system would help residents and help Southern Management manage service requests better.

Requirement Gathering Techniques:

Interviews: Residents were interviewed to collect feedback regarding the existing service request system. Their opinion on how the system should function was considered. All responses were documented and sent for confirmation.

Observations: Participated in activities performed by residents to get a hang of the existing system. This helped gather data accurately and understand what the system was doing. This technique was relatively quick to implement to gather requirements in conjunction with the interviews of residents.

Function 5:

Remove duplicate rental applications of the same apartment type

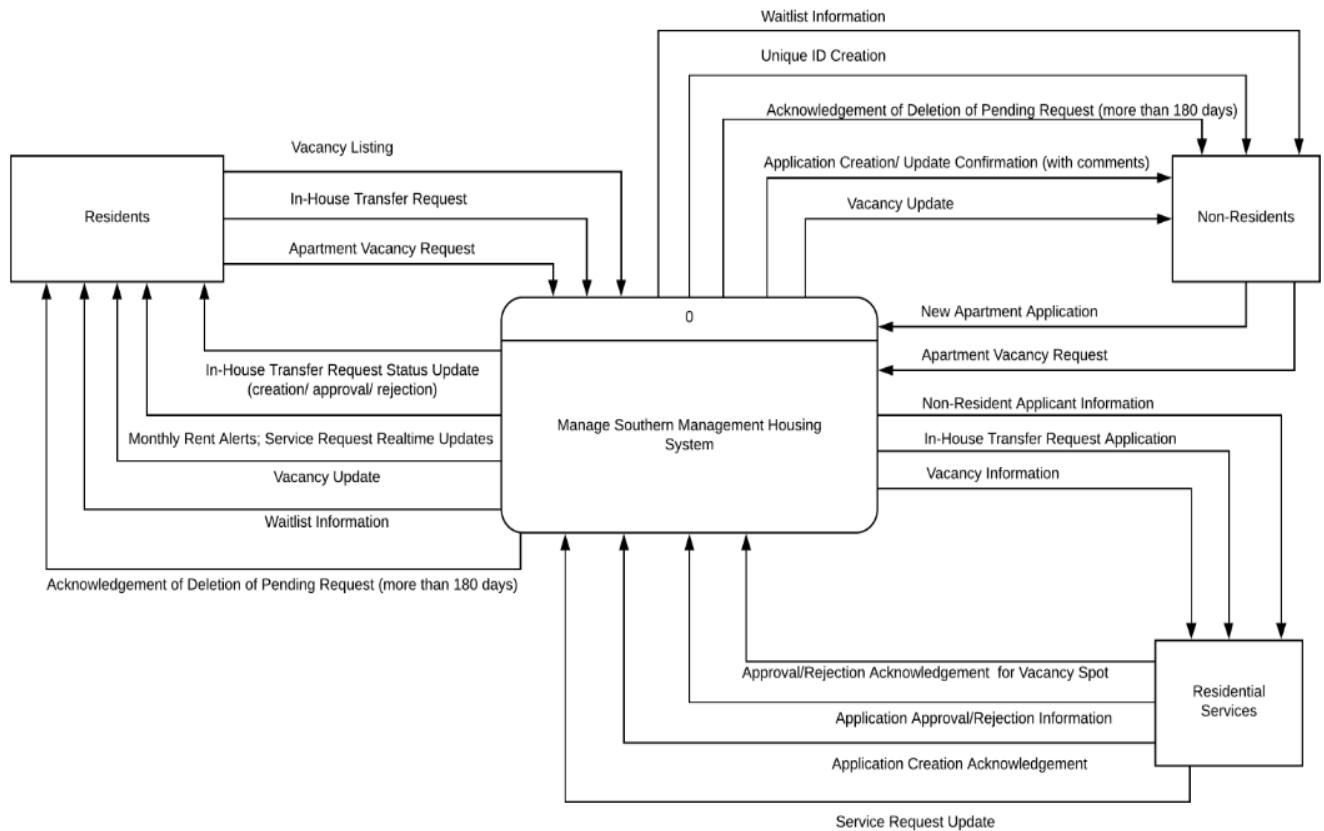
Fact findings: Unlimited applications for similar apartments were causing data storage and maintenance issues. There was no time period as to when the applications would be removed from the system.

Requirement Gathering Techniques:

Questionnaires: Out of all five functions, this function requires the most enhancement and requires very low user involvement. Hence, the approach of questionnaires was chosen. Opinions of stakeholders like system analysts were considered.

2. Proposed Data Flow Diagrams

2.1 Context Diagram



Explanation:

External entities:

- **Residents:** This external entity represents a resident that can apply for in-house transfer requests, look for space in other apartments in the community and post ads related to vacancy in their apartment on the Southern Management portal.
- **Non-Residents:** This external entity represents applicants who are not part of the Southern Management community. They fill in new applications and apply for vacancies.
- **Residential Services:** This external entity represents the residential services or in simpler terms, offices that control day-to-day activities of Southern Management. From screening applications to serving plumbing requests, everything goes through these offices and they are located in every Southern Management housing community.

Manage Southern Management Housing System is the central processing system for all data used and stored in the application creation and transfer request submission process. System inputs and outputs are listed below:

System Inputs

From Residents (External Entity):

- In-House Transfer Request.
- Vacancy Listing.
- Apartment Vacancy Request.

From Non-Residents (External Entity):

- Applicant Information.
- New Apartment Application.
- Apartment Vacancy Request.

From Residential Services (External Entity):

- Approval/Rejection Acknowledgement for Vacancy Spot.
- Application Approval/Rejection Information.
- Application Creation Acknowledgement.
- Service Request Update.

System Outputs:

To Residents (External Entity):

- In-House Transfer Request Status Update (creation/ approval/ rejection).
- Vacancy Update.
- Monthly Rent Alerts; Service Request Real-time Update.
- Waitlist Information.
- Acknowledgement of Deletion of Pending Request (more than 180 days).

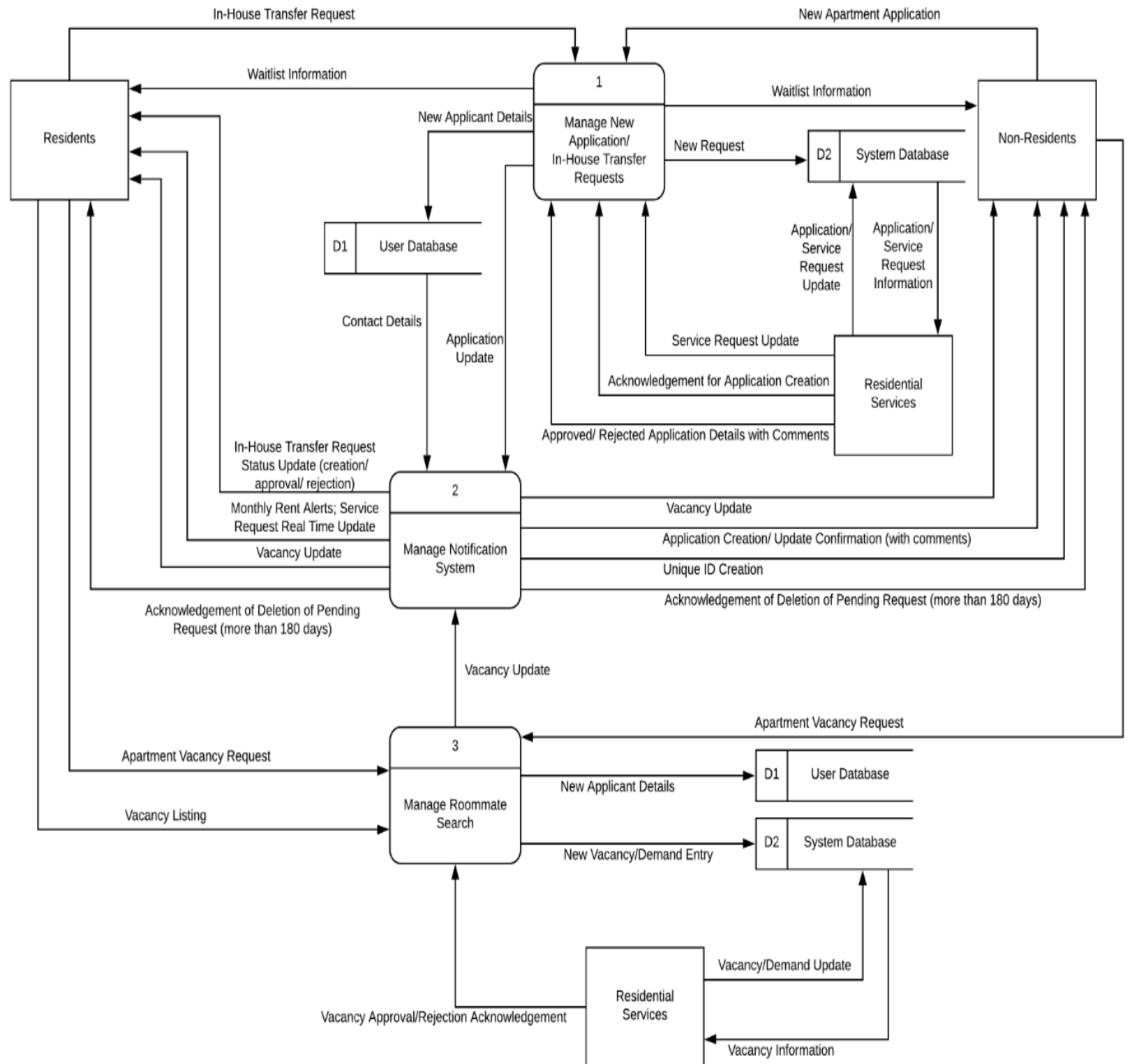
To Non-Residents (External Entity):

- Acknowledgement of Deletion of Pending Request (more than 180 days).
- Application creation/ Update confirmation (with comments).
- Vacancy Update.
- Waitlist Information.
- Unique ID Creation.

To Residential Services (External Entity):

- Non-Resident Applicant Information.
- In-House Transfer Request Application.
- Vacancy Information .

2.2 Level 0 Diagram

**Explanation:**

The three entities described in the context diagram are retained in the Level 0 diagram. However, the system has now been expanded and has three processes and two data stores.

Data Stores:

User Database (D1): This data store stores the information of users. Users can be either residents, non-residents or employees of Southern Management. The information stored is generally name and contact information. This data is inputted to the 'notification system' process. This data store is being used and updated in the scope of this project.

System Database (D2): This data store stores information that is relevant to the system. This includes applications, property details, and service requests. This data store is also being used and updated in the scope of the project.

Processes:

1. Manage New Application/In-House Transfer Request:

This process manages the applications submitted by residents/non-residents. It includes capturing user information and storing it in appropriate data stores. Once recorded, this information is used by other processes and entities to perform other functions. This process directly interacts with users of the system (residents and non-residents)

Process Inputs:

From Residents (External Entity):

- In-House Transfer Request.

From Non-Residents (External Entity):

- New Apartment Application.

From Residential Services (External Entity):

- Approved/ Rejected Application Details with Comments.
- Acknowledgement for Application Creation.
- Service Request Update.

Process Outputs:

To System Database (Data Store):

- New Request.

To User Database (Data Store):

- New Applicant Details.

To Manage Notification System (Process):

- Application Update.

2. Manage Notification System:

This process manages all alerts and notifications of the new system. This includes receiving updates from the 'Residential Services' and the 'User Database' data store and sending relevant notifications to external entities mainly residents and non-residents. This process interacts with the residents and non-residents.

Process Inputs:

From User Database (Data Store):

- Contact Details.

From Search Roommates (Process):

- Vacancy Update.

From Manage New Application/In-House Transfer Request (Process):

- Application Update.

Process Outputs:

To Residents (External Entity):

- In-house Transfer Request Status Update (creation/ approval/ rejection).
- Monthly Rent Alerts; Service Request Real Time Update.
- Vacancy Update.
- Acknowledgement of Deletion of Pending Request (more than 180 days).

To Non-Residents (External Entity):

- Vacancy Update.
- Application Creation/ Update Confirmation (with comments).
- Unique ID Creation.
- Acknowledgement of Deletion of Pending Request (more than 180 days).

3. Manage Roommate Search:

This process manages vacancy listings for residents as well as non-residents. It fetches applications from both these entities and updates the 'System Database' data store accordingly with new requests and listings. This information is used by the 'Residential Services' which sends acknowledgement of approval or rejection back to the process for dissemination to residents and non-residents through the notification system.

Process Inputs:

From Residents (External Entity):

- Vacancy Listing.
- Apartment Vacancy Request.

From Non-Residents (External Entity):

- Apartment Vacancy Request.

From Residential Services (External Entity):

- Vacancy Approval/Rejection Acknowledgement.

Process Outputs:

To Manage Notification System (Process):

- Vacancy Update.

To User Database (Data Store):

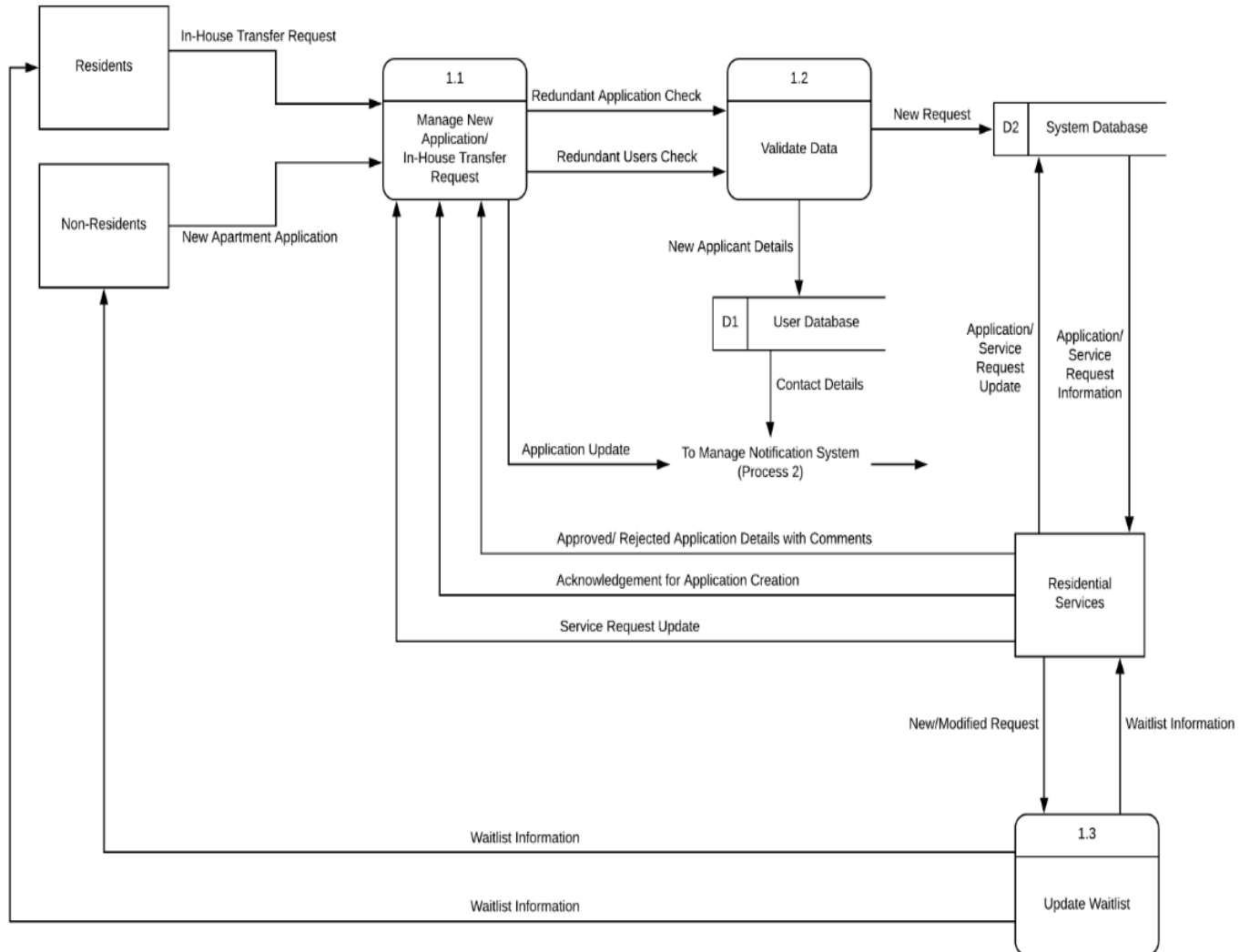
- New Applicant Details.

To System Database (Data Store):

- New Vacancy/Demand Entry/Update.

2.3 Level 1 Diagram

2.3.1 Level 1 : Process 1



Explanation:

The Level 0 diagram has now been split into three Level 1 diagrams. Each process at Level 0 has been explained separately. The 'Level 1: Process 1' explains the process 'Manage New Application/In-house Transfer Request'. This diagram shows management of all the new applications and in-house transfer requests through three new sub-processes and the existing data stores.

Processes:**1.1 Manage New Application/In-House Transfer Request:**

This process gets application requests from residents and non-residents. It passes on these applications to the validation process which performs redundancy checks on these applications. It receives acknowledgements based on approvals/ rejections and service request updates from 'Residential Services' and later on, sends application updates to the notification system.

Process Inputs:

From Residents (External Entity):

- In-house Transfer Requests.

From Non-Residents (External Entity):

- New Apartment Application.

From Residential Services (External Entity):

- Approved/Rejected Application Details with Comments.
- Acknowledgement for Application Creation.
- Service Request Update.

Process Outputs:

To Validate Data (Process 1.2):

- Redundant Applications Check.
- Redundant Users Check.

To Manage Notification System (Process 2):

- Application Update.

1.2 Validate Data

This process takes inputs from the process 'Manage New Application/In-House Transfer Request' and performs validation checks on users and applications to check for redundancy. This helps manage the issue of duplicate applications in the system. Using these inputs, this process sends new applicant details to the 'User Database' data store (D1) and the new request to the 'System Database' data store (D2).

Process Inputs:

From Manage New Application/ In- House Transfer Request (Process 1.1):

- Redundant Applications Check.
- Redundant User Check.

Process Outputs:

To User Database (Data Store D1):

- New Applicant Details.

To System Database (Data Store D2):

- New Request.

1.3 Update Waitlist

This process takes in new or modified requests from 'Residential Services' and updates the waitlist accordingly. Later, it sends this information to all entities involved, namely, 'Residents', 'Non-Residents', 'Residential Services'. This waitlist information is continuously updated in the 'System Database' data store(D2).

Process Inputs:

From Residential Services (External Entity):

- New/ Modified Request.

Process Outputs:

To Residents (External Entity):

- Waitlist Information.

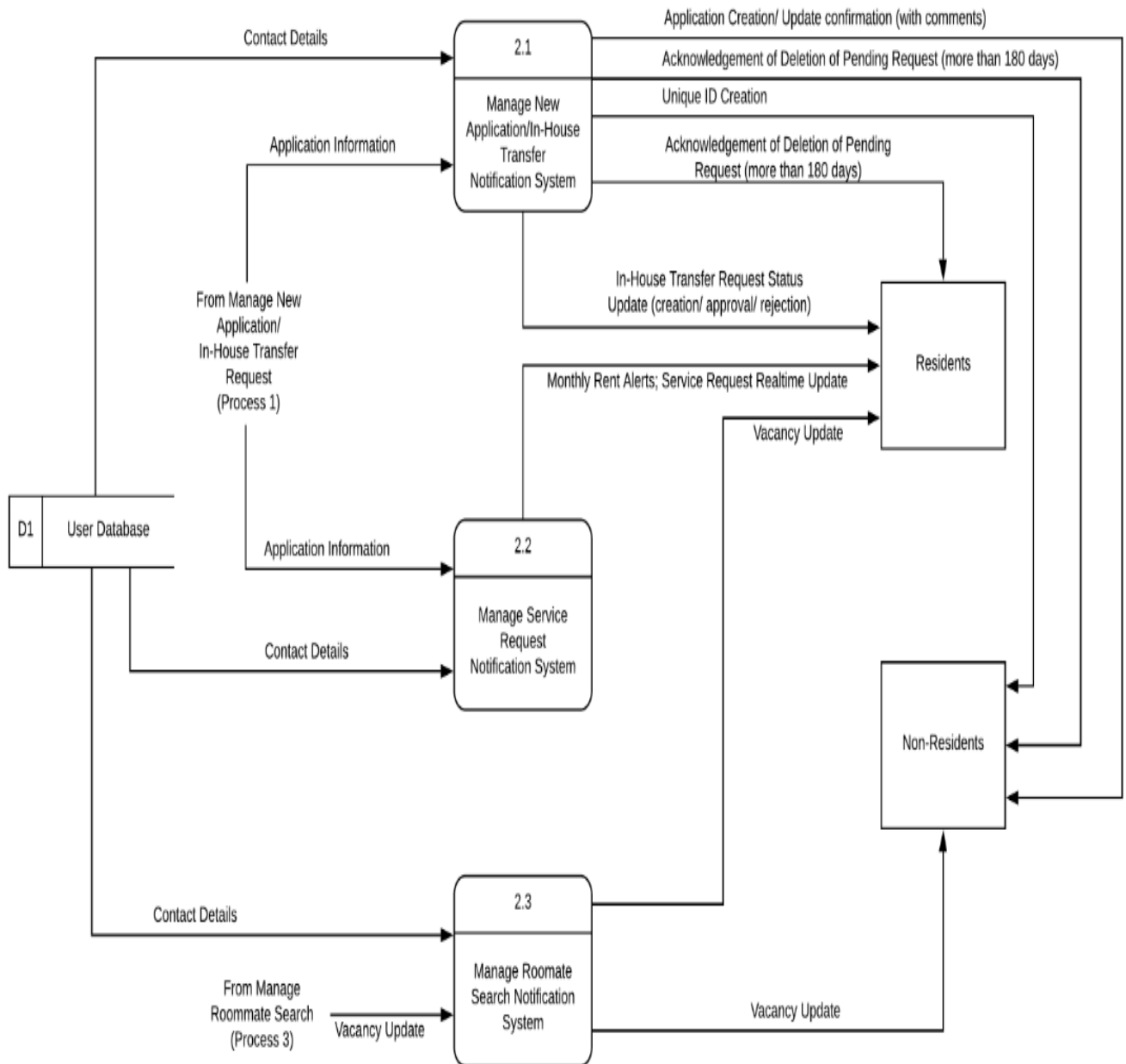
To Non-Residents (External Entity):

- Waitlist Information.

To Residential Services (External Entity):

- Waitlist Information.

2.3.2 Level 1 : Process 2

**Explanation:**

The 'Level 1: Process 2' diagram explains the notification system flow. The 'Manage Notification System' process has been further divided into three sub-processes in this diagram. These processes receive information from Processes 1 and 3 during the scope of the project at different points. This process decides which notifications will go to which entity (user) and how it will flow through the system.

Processes:

2.1 Manage New Application/In-House Transfer Notification System

This process gathers new applications and in-house transfer requests from 'Process 1' along with contact details of the users from data store D1 (User Database). The primary purpose of this process is to send out acknowledgments to residents and non-residents regarding their application status.

Process Inputs:

From Manage New Application/ In-House Transfer Request (Process 1):

- Application Information.

From User Database (Data Store D1):

- Contact Details.

Process Outputs:

To Residents (External Entity):

- In-House Transfer Request Status Update (creation/approval/rejection).
- Acknowledgement of Deletion of Pending Request (more than 180 days).

To Non-Residents (External Entity):

- Application creation/ Update confirmation (with comments).
- Acknowledgement of Deletion of Pending Request (more than 180 days).
- Unique ID Creation.

2.2 Manage Service Request Notification System

This process has been created for the purpose of sending rent alerts and service request updates to residents of Southern Management. It takes its inputs from 'Process 1' and contact details from the data store 'User Database' (D1).

Process Inputs:

From User Database (Data Store D1):

- Contact Details.

From Manage New Application/ In-House Transfer Request (Process 1):

- Application Information

Process Outputs:

To Residents (External Entity):

- Monthly Rent Alerts; Service Requests Real-time Update.

2.3 Manage Roommate Search Notification System

This process solely focuses on sending out updates for vacancy requests that are created by residents of Southern Management and non-residents. It takes in the contact information from the 'User Database' data store (D1) and 'Process 3' and sends out vacancy updates to residents and non-residents.

Process Inputs:

From Manage Roommate Search (Process 3):

- Vacancy Update.

From User Database (Data Store D1):

- Contact Details.

Process Outputs:

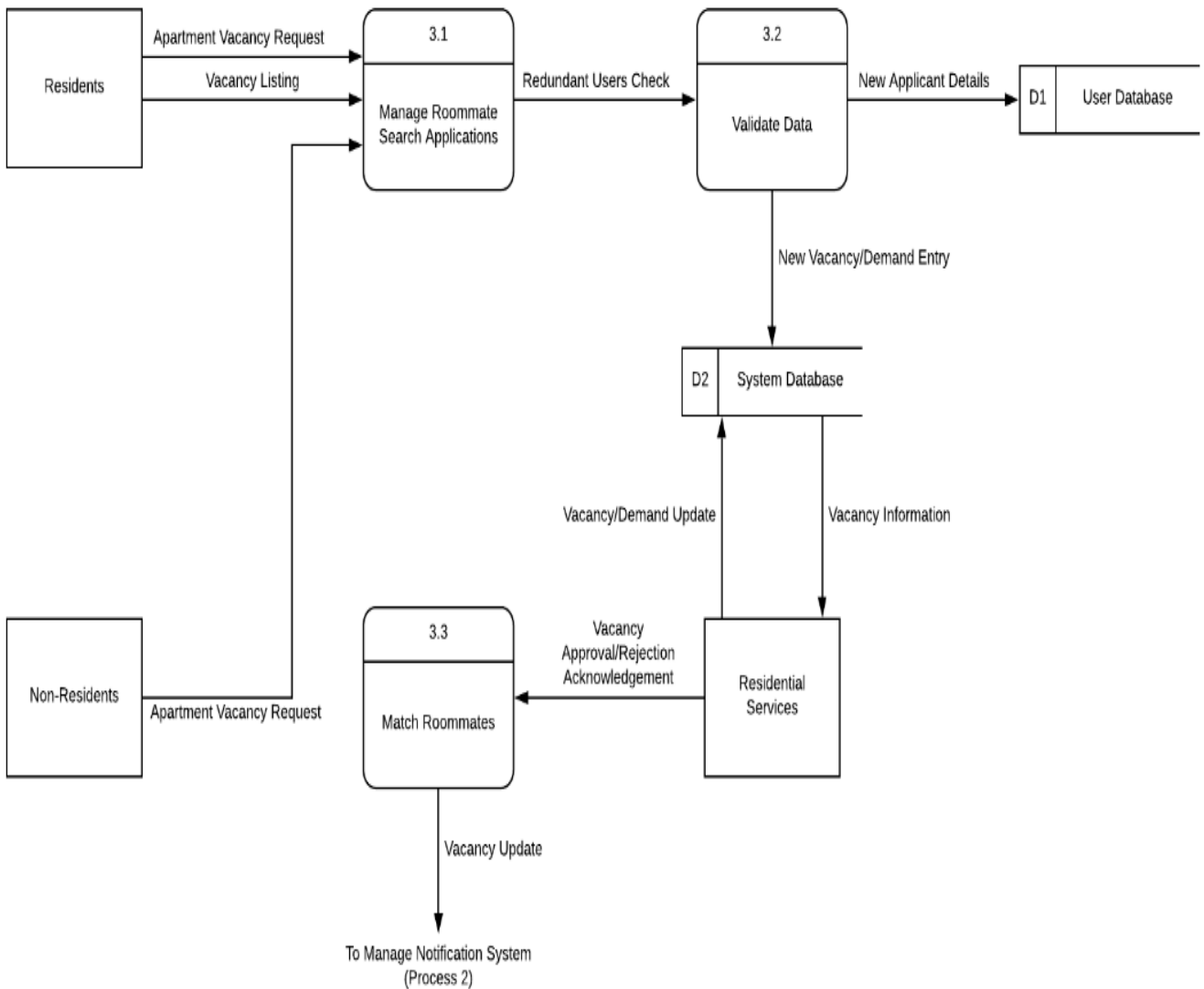
To Residents (External Entity):

- Vacancy Update.

To Non-Residents (External Entity):

- Vacancy Update.

2.3.3 Level 1 : Process 3



Explanation:

The 'Level 1: Process 3' diagram focuses on the management of the roommate search functionality that is being used by both residents and non-residents. 'Process 3' has been divided into three sub-processes to explain the overall flow. This process updates and uses the data stores through the scope of the project.

Processes:

3.1 Manage Roommate Search Applications

This process takes as inputs all the vacancy requests of residents and non-residents. In addition to that, it also fetches vacancy listings from residents of Southern Management. This information is sent to the validation process for redundancy checks.

Process Inputs:

From Residents (External Entity):

- Vacancy Listings.
- Apartment Vacancy Request.

From Non-Residents (External Entity):

- Apartment Vacancy Request.

Process Outputs:

To Validate Data (Process 3.2):

- Redundant Users Check.

3.2 Validate Data

This process validates the information coming in from 'Process 3.1' and updates the data stores accordingly. It stores the new applicant details in the 'User Database' data store (D1) and the new vacancy or demand entry into 'System Database' data store (D2). This information is later passed on to the 'Residential Services'.

Process Inputs:

From Manage Roommate Search Applications (Process 3.1):

- Redundant Users Check.

Process Outputs:

To User Database (Data Store D1):

- New Applicant Details.

To System Database (Data Store D2):

- New Vacancy/ Demand Entry.

3.3 Match Roommates

This process has been created to send vacancy updates to the notification system. These notifications are sent by the 'Residential Services' after gathering this information from the System Database (data store D2).

Process Inputs:

From Residential Services (External Entity):

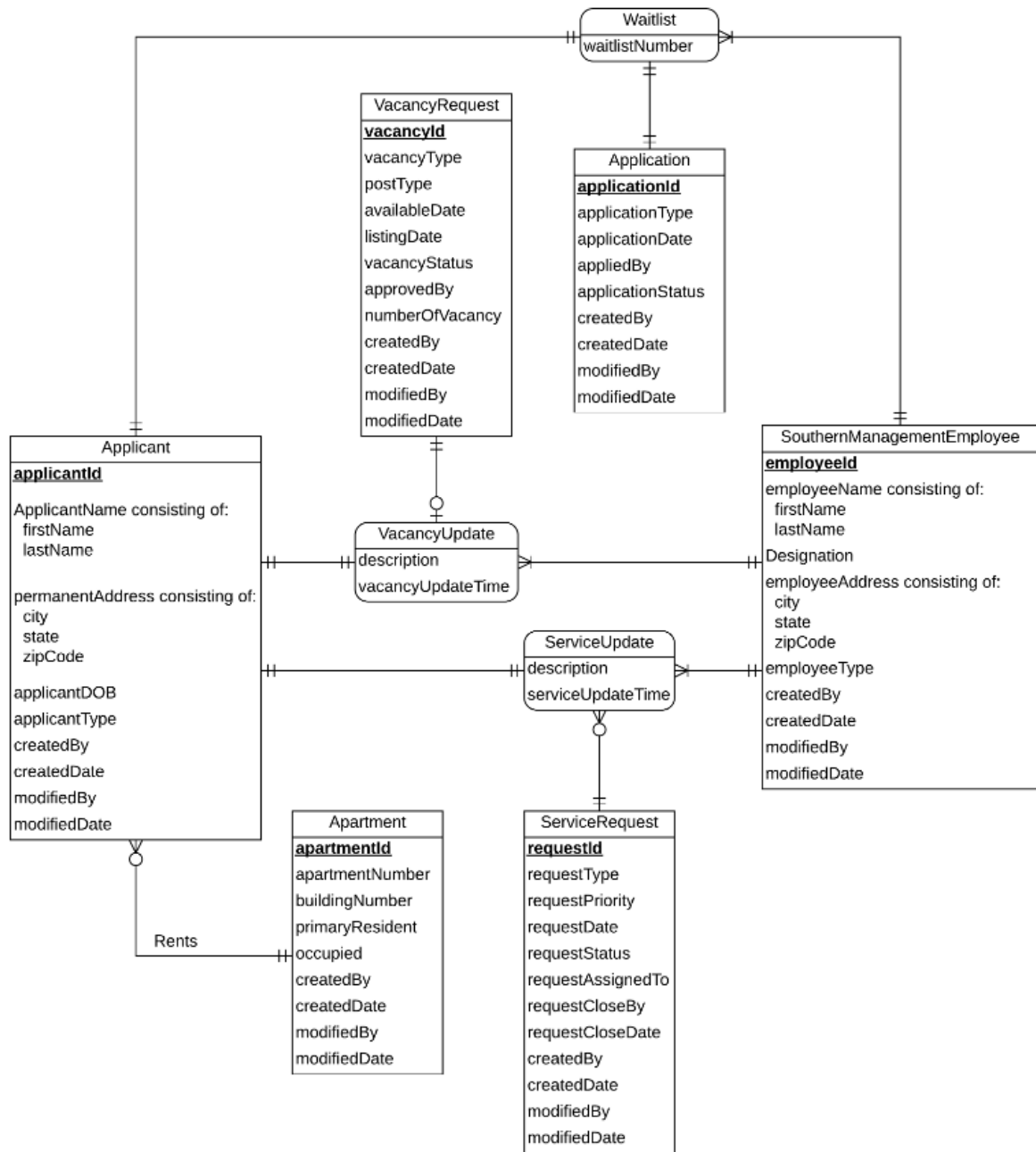
- Vacancy Approval/ Rejection Acknowledgement.

Process Outputs:

To Manage Notification System (Process 2):

- Vacancy Update.

3. Proposed ER Diagram



Explanation:

The proposed ER Diagram helps us identify the different system elements and their relationships with each other. We have a total of 9 entities including 3 associative entities. They are listed and explained as follows:

- 1. Applicant:** This entity holds the information of residents and non-resident applicants. It stores general information about the applicants including contact information.
- 2. Apartment:** This entity hold the information about the apartments in the Southern Management communities. It also holds information about primary occupants.
- 3. SouthernManagementEmployee:** This entity stores information about the Southern Management employees. It holds details such as their basic information and type of employment.
- 4. ServiceRequest:** This entity stores information about the service requests that are initiated by residents of the community. It includes basic information like what type of service request it is and what its priority is.
- 5. Application:** This entity stores information about applications. This includes new applications and in-house transfer requests. It also holds the creation date and type of the application.
- 6. VacancyRequest:** This entity stores information about vacancies in general. This includes applications from residents and non-residents.
- 7. Waitlist (Associative Entity):** This entity is a relational entity between Applicant, Application and SouthernManagementEmployee. It holds the waitlist information.
- 8. VacancyUpdate (Associative Entity):** It is a relational entity between VacancyRequest, Applicant and SouthernManagementEmployee. This entity provides updates about vacancies.
- 9. ServiceUpdate (Associative Entity):** This is a relational entity between Applicant, ServiceRequest and SouthernManagementEmployee. This entity provides updates on service requests.

Relations:

We have a total of three ternary relationships and one binary relationship. They are listed as follows:

1. Rents (Binary):

One Applicant to exactly one Apartment

One Apartment to zero or many Applicant.

2. Waitlist (Ternary):

One Applicant, one Application to one or many SouthernManagementEmployee

One Application, one SouthernManagementEmployee to exactly one Applicant.

One Applicant, one SouthernManagementEmployee to exactly one Application

3. VacancyUpdate (Ternary):

One Applicant, one VacancyRequest to one or many SouthernManagementEmployee

One VacancyRequest, one SouthernManagementEmployee to exactly one Applicant.

One Applicant, one SouthernManagementEmployee to zero or one VacancyRequest

3. ServiceUpdate (Ternary):

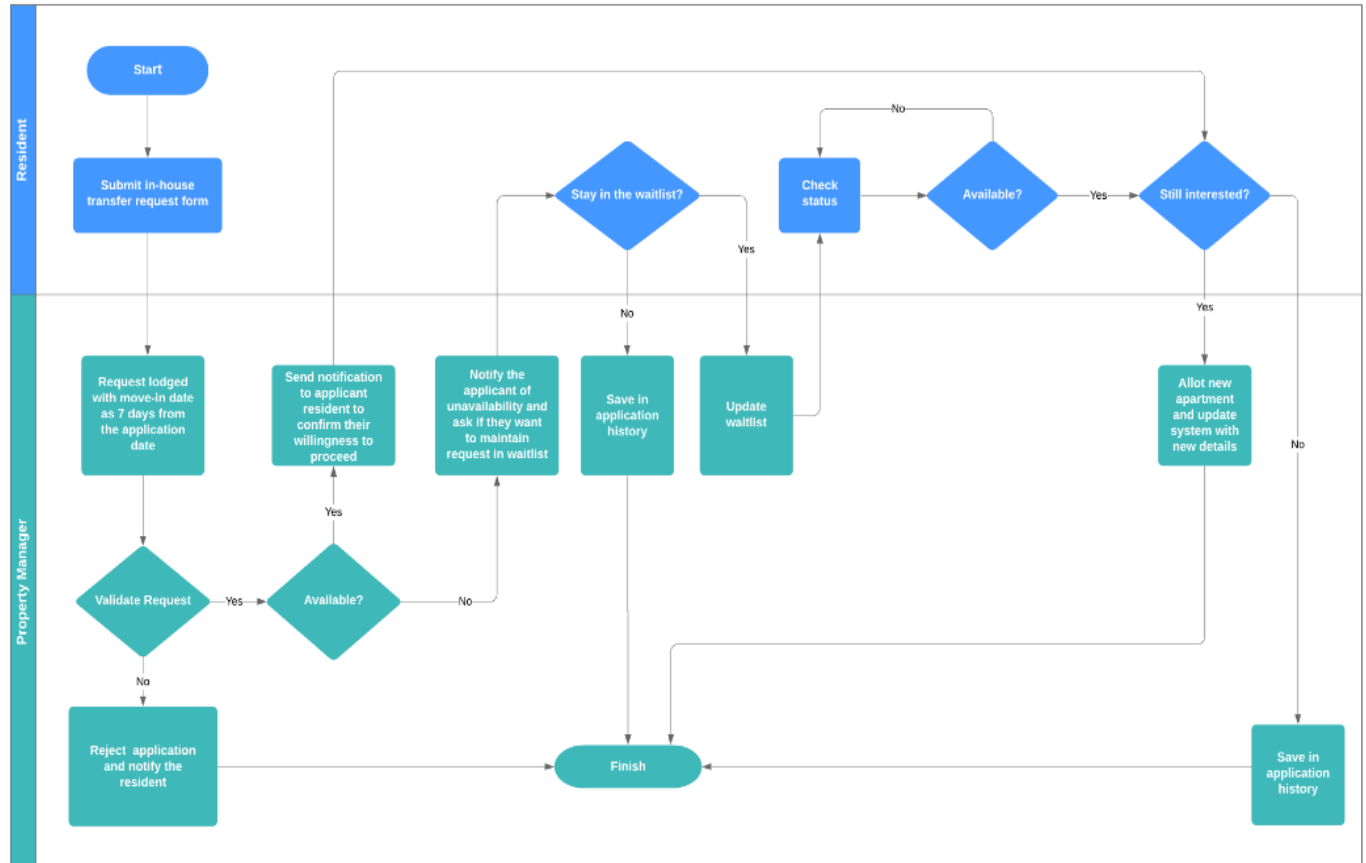
One Applicant, one ServiceRequest to one or many SouthernManagementEmployee

One ServiceRequest, one SouthernManagementEmployee to exactly one Applicant.

One Applicant, one SouthernManagementEmployee to zero or many ServiceRequest.

4. Process Flow Diagram

Function: Provision for online in-house transfer request



5. Conclusion

This concludes the first 2 phases – the Planning and the Analysis phase of the project. During the planning phase, the scope and various feasibilities of the project were discussed. A consolidated project plan was put into place. Based on this, the analysis phase was initiated, and a comprehensive set of analyses were laid out.

The data flow diagrams have been worked well upon and are granular to a great level and intended to help in the following phases, like developing an accurate design plan, for starters. The overview created using the context diagrams helped us draft the Level 0 and Level 1 diagrams. Collectively with the help of the ER diagram and Process Flow diagram, we can confidently infer that our recommendations to enhance the Southern Management system are up to the mark. This will improve the efficiency of their system and improve user experience and benefit them significantly in the long run and the investment will significantly be outweighed by the rewards it will bring.