UNIVERSITY OF CALGARY | Schulich School of Engineering

Term Project I Online by-owner real estate management system (OBRM)

Prepared for:

Mahmood Moussavi

November 1, 2019

Prepared By:

Group 3: Zixin (James) Chen
Asif Bux
Blair Chau
Vaibhav Jadhav

TABLE OF CONTENTS

List of Figures

List of Tables

Glossary

Term	Definition
AWS	Amazon Web Services - A suite of services offered by Amazon including database management, analytics, and cloud computing.
AWS EC2	Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
AWS S3	Amazon Simple Storage Service (S3) is to store and retrieve any type of data.
AWS Lambda	AWS Lambda lets you run code without provisioning or managing servers.
AWS IAM	AWS Identity and Access Management (IAM) enables you to manage access to AWS services and resources securely.
HTML and CSS	HTML is a mark-up language designed to be displayed in web browser and it is often combined with Cascading Style

	Sheets (CSS) and JavaScript (JS). CSS describes how elements should be rendered on the screen.
JavaScript	JavaScript is a client-side scripting language which is used to add functionality for HTML elements and access HTML Document Object Model (DOM).
JSON	JavaScript Object Notation (JSON) is a light weighted data interchange format used to send and re
AWS RDS	Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud.
DFD	Data Flow Diagram - A diagram which displays the type and direction of data between the various components of the system.
MLS	A multiple listing service is used by real estate brokers to establish contractual offers of compensation and accumulate and mutually share information about properties.
OBRM	An Online By-Owner Real Estate Management (OBRM) system is used to facilitate buying and selling of real estate properties, where properties are listed by their owners.

1.0 Introduction

1.1 Purpose

The product whose software requirements are specified in this document is an online by-owner real estate management system (OBRM). The scope of this system is limited to the core functionality required for the sale and purchase of real estate property. Additional functionality such as access to multiple listing services (MLS), legal processes regarding real estate transactions, processing of payments between the buyer and seller and other financial considerations such as mortgage applications are all beyond the scope of this product.

1.2 Overview of the Document

1.2.1 Document Conventions

For the document convention following styles were used:

- Heading 1: Sample Text 1.0 Introduction, font size 16 and font type Calibri Light.
- Heading 2: Sample text 1.2 Overview of the Document, font size 13 and type Calibri Light.
- Heading 3: Sample Text 1.2.1 Document Convention, font size 12 and type Calibri Light.
- Normal Text throughout the document is font size 12 and font type Calibri.

1.2.2 Readers and Audience

This document is intended for the reference of developers and testers of this OBRM application and any future related applications; as well as managers and host users of this application, writers for user documentation and technical support staff for this application. The document begins with an overall description of the application, which is followed by a description of each function of the application in greater detail. Following this, a description of the external requirements is provided, after which there is a list of all non-functional requirements and issues regarding quality control. The document concludes with a summary of the tentative release plan for the application.

For managers and host users of the application, relevant sections first include the product scope of this application (Section 1.4), followed by the overall description (Section 2). Users of a specific system function should reference the section pertaining to that function directly (Section 3). Managers should also consult the external interface requirements (Section 4) and the non-functional and quality requirements (Section 5).

For all other readers of this document, start with the overall description (Section 2). Refer to system features (Section 3) and the external interface requirements (Section 4) for relevant

details. Refer to the release plan (Section 6) for overall timeline and deadline requirements. Quality requirements and non-functional requirements (Section 5) are available for system information beyond the core functionality of the application. Additional supplementary diagrams can be found in the appendices (Section 7).

1.3 References

- Barry, D. K., & Dick, D. (2013). *Web services, service-oriented architectures, and cloud computing: the savvy managers guide*. San Francisco, CA: Morgan Kaufmann.
- Buy and Sell your Home in Alberta. (n.d.). Retrieved November 1, 2019, from https://purplebricks.ca/ab.
- data flow diagram. (n.d.). Retrieved November 1, 2019, from https://www.sciencedirect.com/topics/computer-science/data-flow-diagram.
- Defining website data structure. (n.d.). Retrieved November 1, 2019, from https://docs.kentico.com/k8/developing-websites/defining-website-data-structure.
- Dick, jeremy. Hull, elizabeth. Jackson, ken. (2018). *Requirements Engineering*. S.l.: Springer International Pu.
- Forster, M., & Masters, G. N. (1998). AWS Products. Retrieved November 1, 2019, from https://aws.amazon.com/products/?nc2=h_ql_prod_fs_f.

1.4 Product Scope

The purpose of the OBRM application is to provide a platform through which a seller can create an accessible property listing for selling real estate property, and through which a buyer can access, browse and search for real estate property listings in order to begin the process of purchasing property from a seller.

The system operation for a seller is shown in Figure 1.

Figure 1: Product Scope for Seller

As shown in the figure, the process for a seller is as follows:

- A seller creates and posts a listing for a property for sale under a listing contract with a set duration.
- A property manager verifies the property listing to ensure details provided are accurate.
- If verification is successful, the listing is published on the application. The listing will be available for any prospective buyers to browse/search for.
- A buyer contacts the seller externally and the deal is eventually closed by the seller if an agreement is reached, or the listing is either removed after its contract duration expires, or the seller renews the contact listing to continue to have it visible for buyers.

The system operation for a buyer is shown in Figure 2.

Figure 2: Product Scope for Buyer

As shown in the figure, the process for a buyer is as follows:

- A buyer searches for properties from available published listings on the application.
- Listings matching the buyer's search are retrieved and listed for the buyer to browse through for details.
- A buyer interested in purchasing a property then contacts the seller of that property externally.
- The buyer meets with the seller and views the property in person. If the buyer and seller do not reach a deal, the buyer can return to continue to search for other properties.

It is important to note that all details regarding contact between a buyer and seller or the transaction for a property purchase is not included in the scope of this application. The application scope also does not include any interaction or data flow with any MLS.

2.0 Overall Description

2.1 Product Perspective

This system will be a new, self-contained, standalone product and not part of any series of product offerings. It will offer an alternative to people looking to buy or sell a house without hiring Real Estate Agents, with the core functionalities being browsing and listing properties. The system will be hosted using AWS and their suite of computational and database services, which will power the backend for the website application.

2.2 Product Features

The context diagram for this system is shown in Figure 3 below.

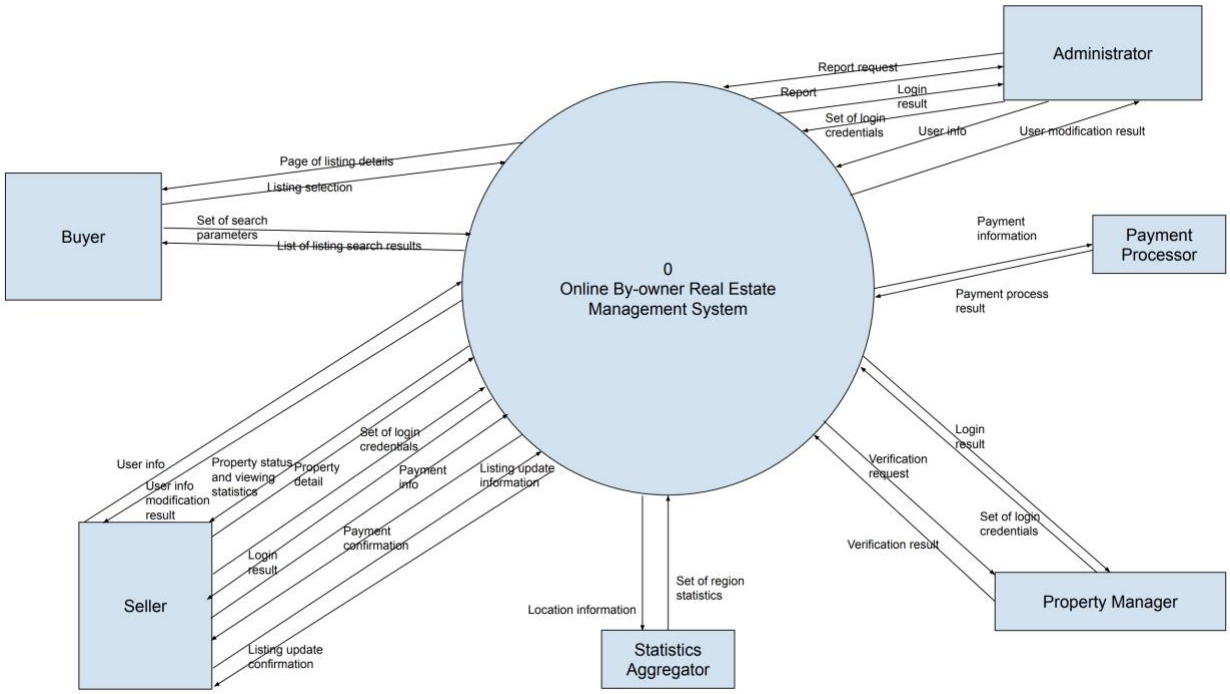


Figure 3: OBRM Context Diagram

The primary functions of the system are the following:

- Registers a user as a seller to keep track of their property listings.
- Creates a listing based on property information provided by a registered seller.
- Provides sellers with access to view their listings' status and statistics.

- Obtains payment information from registered sellers for listings to send to a payment processor/financial institution.
- Provides buyers with access to search for listings based on various search parameters.
- Provides buyers with access to select and view listing information from search results.
- Provides property managers with requests to assess properties listed and receives the results of the property assessments.
- Creates reports based on operation data as requested by administrators.

2.3 User Classes and Characteristics

User classes are defined into four categories of sellers, buyers, administrators and property managers. Sellers are the property owners who are responsible for inputting property details on the application to create property listings. Buyers comprise the target market who will browse properties through the application and make purchases through external contact with sellers. Administrators are responsible for maintaining the whole application and the different accounts involved with the application. Property managers are responsible for the verification of property listings when created or modified.

1. ***Owners/Sellers***
    - Sellers create property listings through the application by purchasing a listing contract with a set duration.
    - Sellers are responsible for managing their property listing on the application.
    - Sellers can upload images, videos and a brief description of their property on the listing page and have access to request changes.
    - Sellers can change the status of their listings by marking them as sold, deleting them or extending the duration of their contract.
    - Sellers have access to statistics such as number of views or searches their property listing appeared in.

2. ***Buyers***
    - Buyers are users of the application who will browse property listings by using appropriate search parameters on a listing search page.
    - External email alerts and SMS service can be used to send out notifications for interested properties from application, but the buyer does not have a registered account with the system.
    - Contact with the property owner involving offers, negotiation, legal considerations and transactions are all external to the application.

3. ***Administrators***
    - Administrators have access to the entire application, and they can configure or modify the application as per business or maintenance needs.
    - Administrators have access to a report generation feature allowing them to view aggregations of various system statistics.

4. ***Property Managers***
    - Property managers monitor and verify owner's property listing before allowing it to be published on the application and verify any changes owners make to their listings before the listing is re-published.
    - Property managers monitor sellers' accounts to ensure that only genuine sellers will have access to the application.

2.4 Operating Environment

The application will run on both web portal platforms and mobile platforms. The application's software can be divided into front-end and back-end components, where the front-end will be written in HTML, CSS and JavaScript and the back-end will be using Amazon Web Services (AWS).

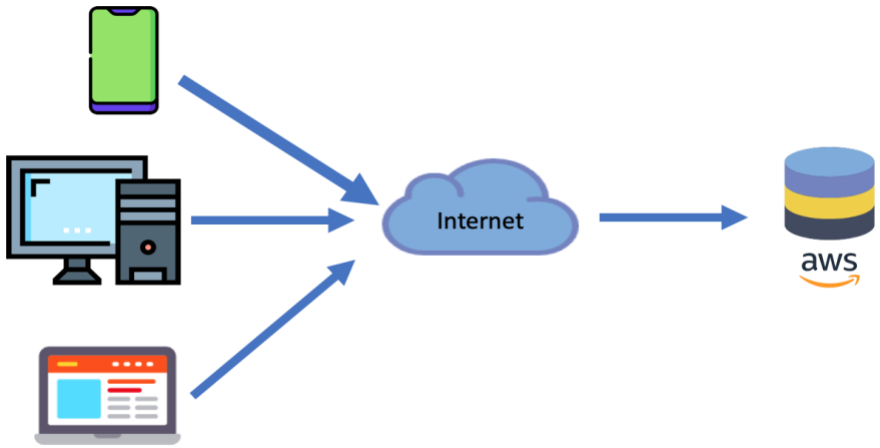


Figure 4: Operating Environment

Since the application will make use of this cloud platform, the system will not require much hardware as AWS will remove the need for in-house servers and data centres. AWS will use a Linux-based operating system to host the application. Amazon's Elastic Compute Cloud (EC2) will also be a part of the backend, which will have resizable computing capacity that allows for better web-scale computing in the cloud. The application will also use Amazon's Simple Storage Service (S3) to store data such as images, videos and other similar files.

The operating environment for users to run the web application is any computer with standard web browsers such as Google Chrome, Internet Explorer or Mozilla Firefox. The supported operating systems will be Windows and MacOS for the web application and Android version Lollipop – 5+ and iOS version 11+ for the mobile application.

2.5 System Constraints

Since the site is hosted on AWS, the biggest constraint associated with this is the developers' requirement to be familiar with AWS, the services it provides, and cloud computing in general. This can mean developers will have to take courses online or read extensive documentation to learn exactly how the services work and which services would be best to use for specific tasks. For example, developers would need to distinguish between which database service would fit their needs best, such as choosing between Amazon RDS or Amazon Aurora, and deciding on the database size and memory retrieval speeds. Developers will need to learn about the services AWS provides for API construction, site hosting and domains, database and site file storage, as well as services that can be used to enhance optimization and increase reliability. They will also need to become familiar with how to secure site data and access along with learning how to properly allocate security features such as VPCs and security groups, as well as limiting access with IAM roles. AWS charges users for each service, so there could also be budget constraints depending on which services are desired, and so budget must be balanced out with desired performance. This ties into hardware constraints, as AWS has many options for powerful hardware services such as higher RAM and CPU in EC2 instances; however, these come at a higher cost. The developers would also require an individual or small team of individuals to be devoted to monitoring AWS and serving as a point of communication with Amazon in case any questions or problems arise. There are also language requirements, as AWS does not support every programming language. Developers would need to be familiar with languages like Python, Java, Node.js, PHP, and JSON, to name a few, in order to successfully use AWS and its services.

A sample user search interaction with AWS is shown in the diagram below. The user would search for a home and filter criteria, and that search criteria will be sent as a keyword to the Amazon API Gateway API. The API would then trigger a Lambda function that would contain code that converts the keywords to a SQL query, and the function will query the database. It will find necessary information from the database and return that through the Lambda, which will send relevant information to the API and that returns the result back to the user to be displayed on-screen.

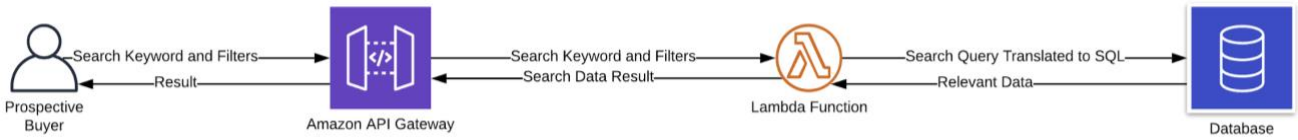


Figure 5: Query Search Flow on AWS

2.6 User Documentation

Documentation for home-sellers and homebuyers to use the website will include a short tutorial video linked on the home landing page, as well as a link to a text FAQ to answer common questions. Users will also be able to ask further questions that are not answered by the video tutorial or FAQ in a dialog box located on the FAQ page and an email will be sent to a site admin who's suited to answer their questions. For the site administrators and managers, the documentation to use the site will include text-based documents detailing the steps and requirements to approving home-buyer posts and verifying the authenticity, eligibility and accuracy of their listings.

2.7 Assumptions and Dependencies

First and foremost, this system will be hosted on AWS and will not use standalone databases. All data will be stored on AWS databases and all processing will be performed by AWS cloud computing. Payment processing for hosting property listings will be handled by Stripe integrated with AWS. Data for housing area statistics will be taken from Statistics Canada using its APIs.

3.0 System Features

3.1 System Function 1 - Property Search

3.1.1 Description of Function

This process allows buyers to search for a list of properties based on various search parameters. The Level 1 and 2 data flow diagrams for this feature are shown in Figure 5.

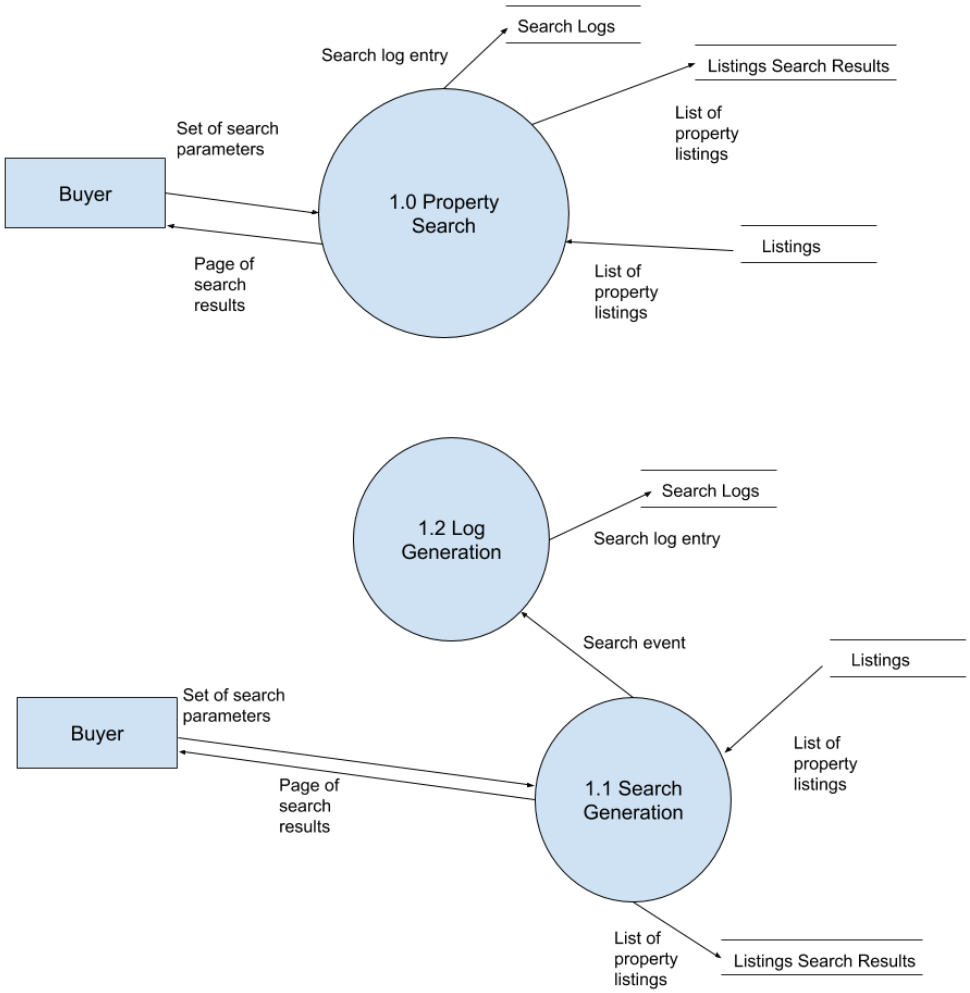


Figure 6: Property Search Level 1 & 2 DFD

3.1.2 Preconditions and Promises

Preconditions:

- Needs a valid set of search parameters.

Promises:

- Obtains a page of listings matching the search parameters.
- Generates a search log entry.

3.1.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 6.

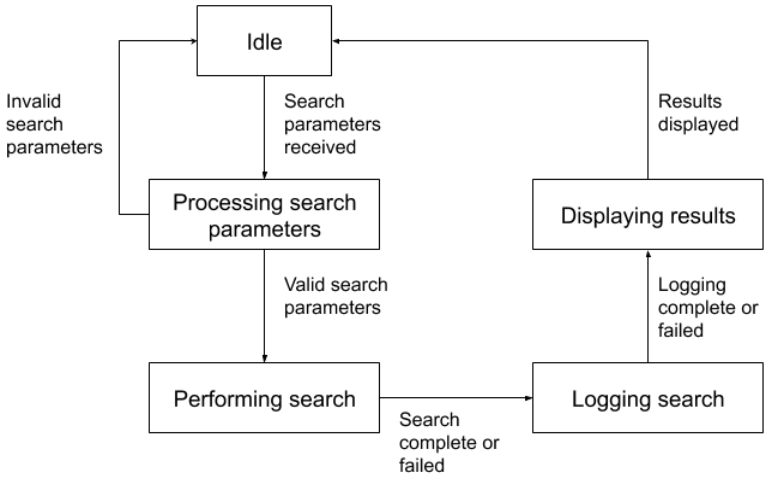


Figure 7: Property Search STD

3.2 System Function 2 - Property Showcase

3.2.1 Description of Function

This process allows buyers and sellers to view the details of a property listing. The Level 1 and 2 data flow diagrams for this feature are shown in Figure 8.

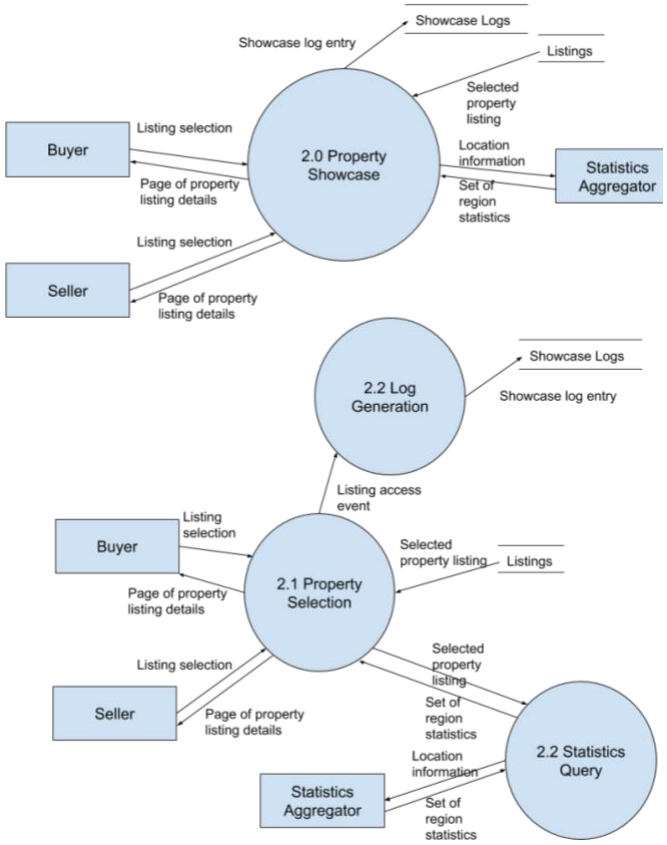


Figure 8: Property Showcase Level 1 & 2 DFD

3.2.2 Preconditions and Promises

Preconditions:

- Needs an existing listing selection.

Promises:

- Obtains a page of details of the selected listing.
- Generates a showcase log entry.

3.2.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 9.

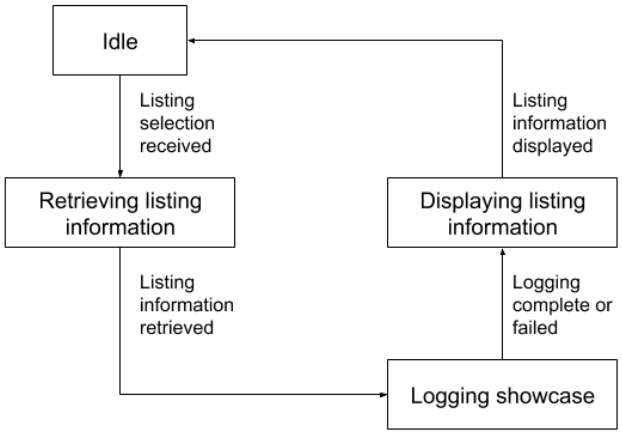


Figure 9: Property Showcase STD

3.3 System Function 3 - Property Listing Creation

3.3.1 Description of Function

This process allows buyers and sellers to view the details of a property listing. The Level 1 and 2 data flow diagrams for this feature are shown in Figure 10.

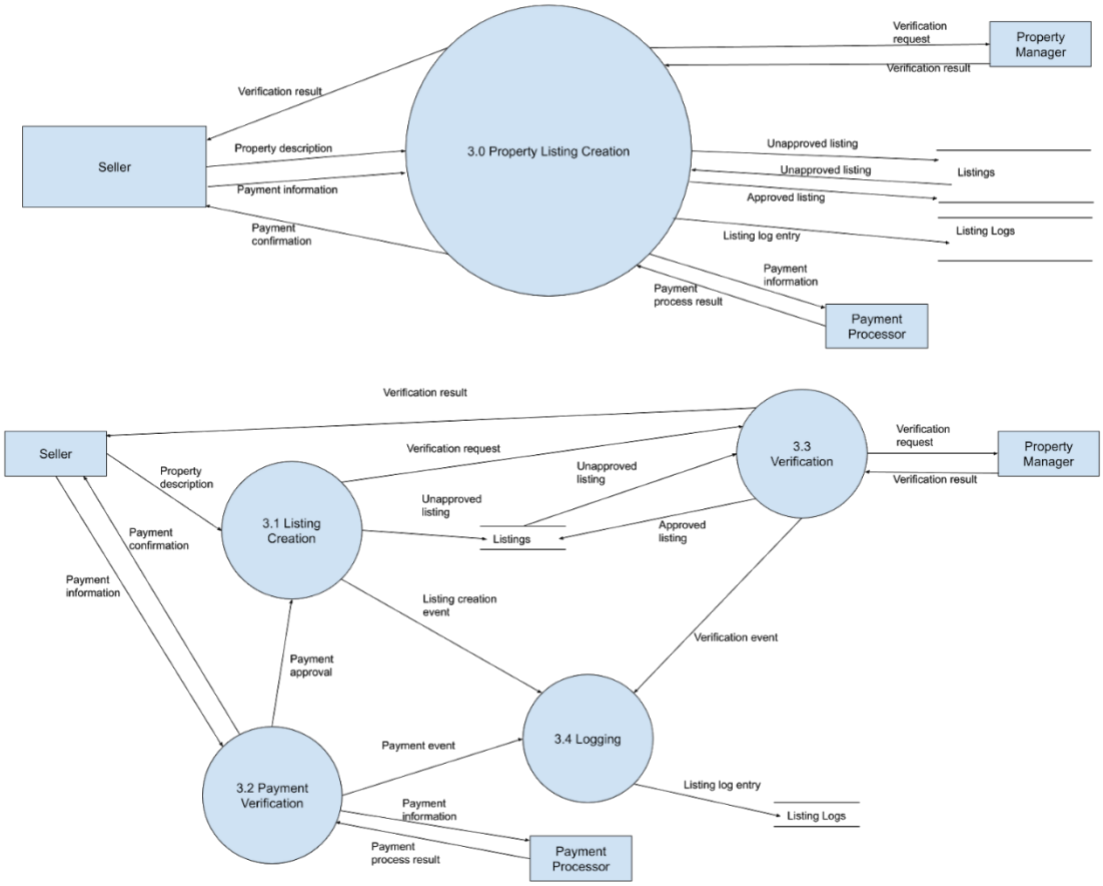


Figure 10: Property Listing Creation Level 1 & 2 DFD

3.3.2 Preconditions and Promises

Preconditions

- Needs a description of a property from a registered, logged in seller (see Section 3.5 - System Feature 5 for user registration; Section 3.7 - System Feature 7 for login).

Promises

- Generates a listing entry for the listing data store.
- Generates a listing log entry at each stage of the listing creation process (creation, payment, verification).

3.3.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 11.

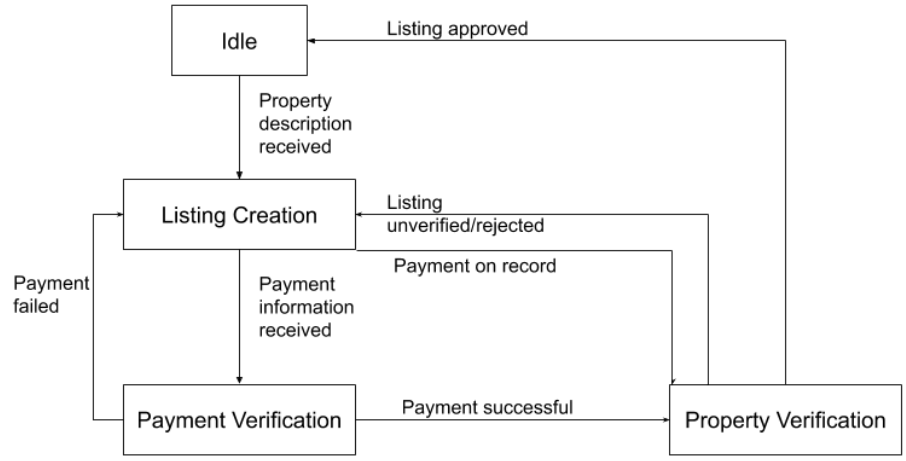


Figure 11: Property Listing Creation STD

3.4.1 Description of Function

This process allows administrators to generate reports based on log information collected from the other processes in the system. The Level 1 data flow diagram for this feature is shown in Figure 12.

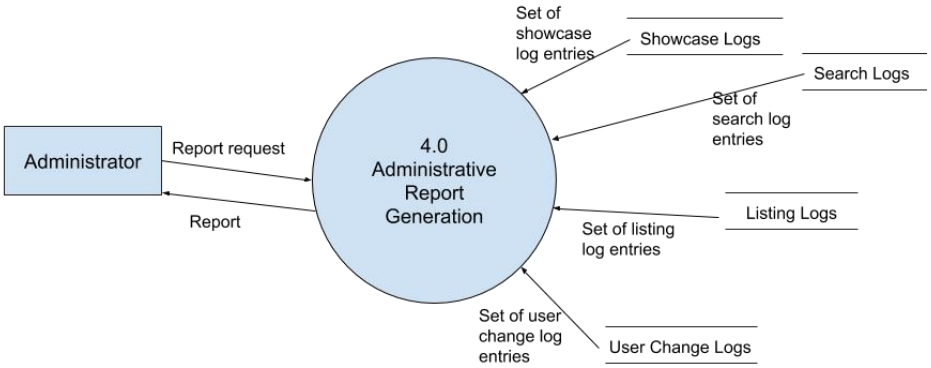


Figure 12: Administrative Report Generation Level 1 DFD

3.4.2 Preconditions and Promises

Preconditions

- Needs a valid report request from a logged in administrator.

Promises

- Obtains a report based on the provided request.

3.4.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 13.

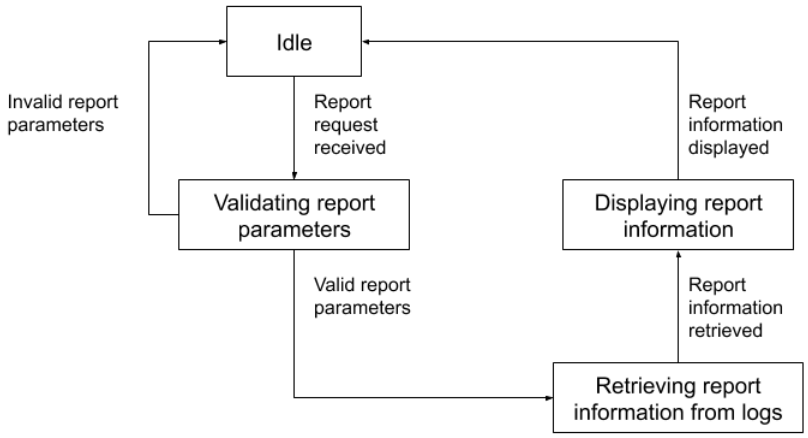


Figure 13: Administrative Report Generation STD

3.5 System Function 5 - User Management

3.5.1 Description of Function

This process allows sellers to manage their own account, or for administrators to manage the account of a manager or seller. This includes the functionality of registering an account, deleting an account and changing account information. The Level 1 and 2 data flow diagrams for this feature are shown in Figure 14.

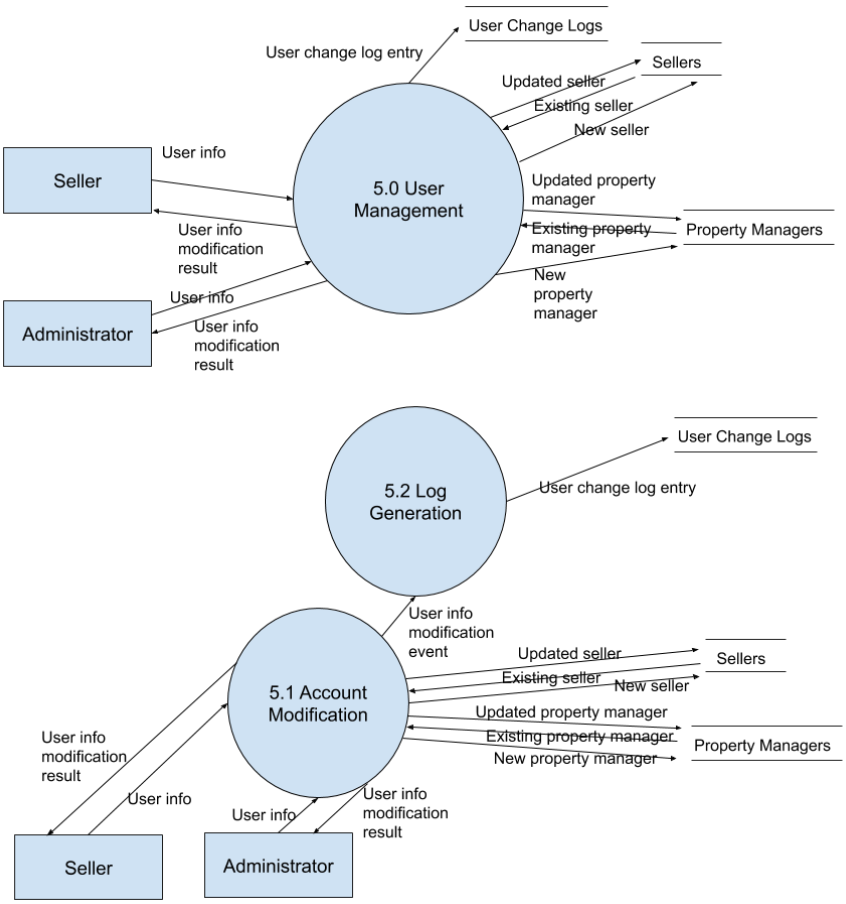


Figure 14: User Management Level 1 & 2 DFD

3.5.2 Preconditions and Promises

Preconditions

- Needs valid account creation information from a seller who does not yet have an account (not logged in) OR valid account creation/modification information from a logged in administrator.

Promises

- Creates a new seller account based on provided seller information OR creates/modifies any account based on information provided by the administrator.

3.5.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 15.

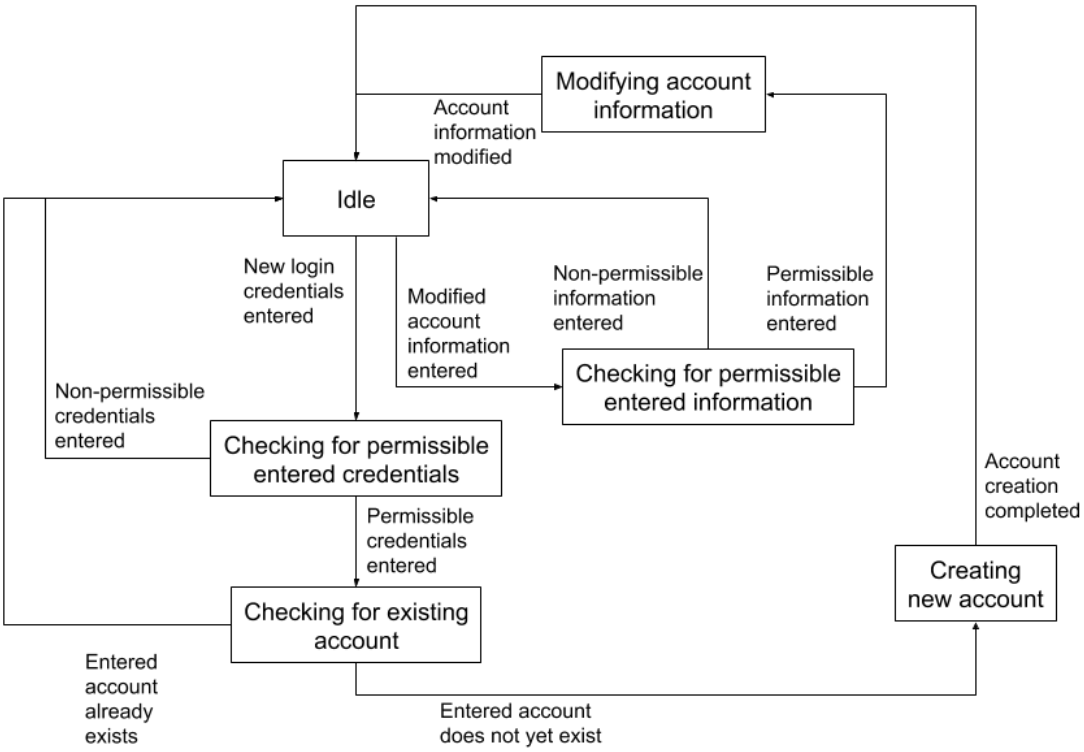


Figure 15: User Management STD

3.6 System Function 6 - Listing Modification

3.6.1 Description of Function

This process allows sellers to modify their property listing such as listing details, listing price, listing removal and finally also marking if the listing has been sold. The Level 1 and 2 data flow diagrams for this feature are shown in Figure 16.

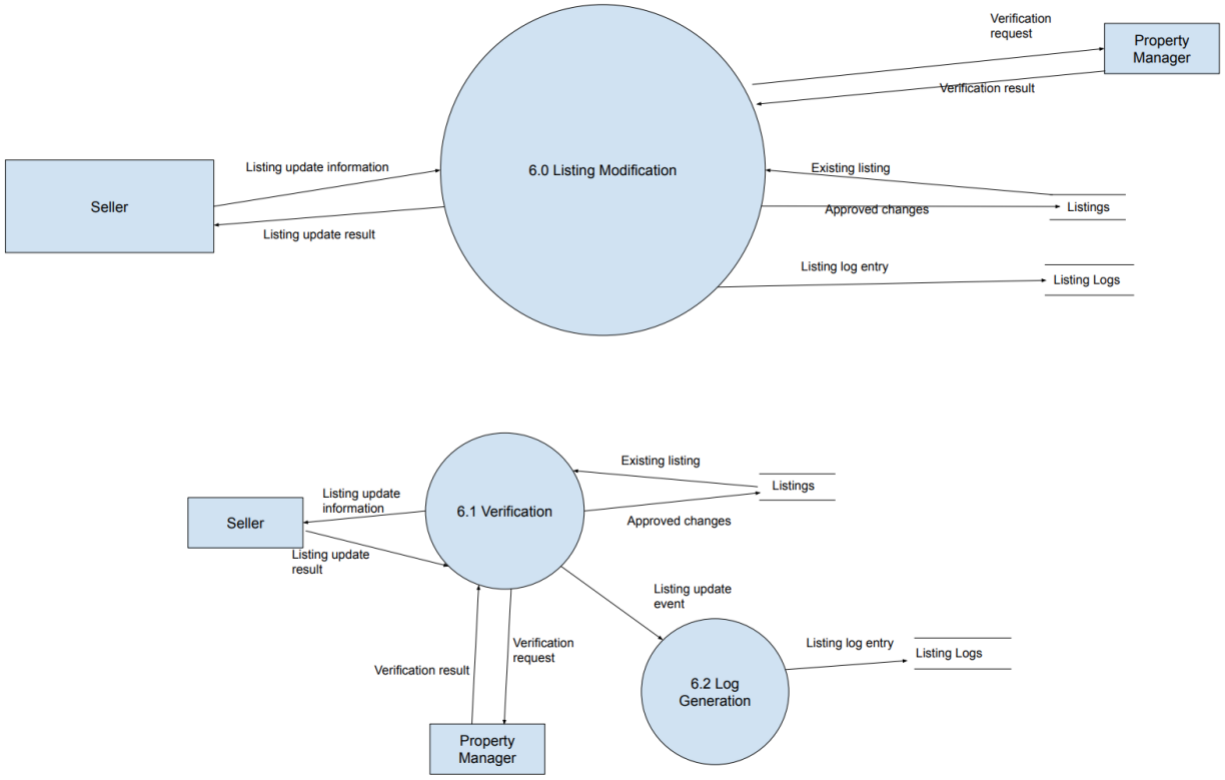


Figure 16: Listing Modification Level 1 & 2 DFD

3.6.2 Preconditions and Promises

Preconditions

- Needs an existing property listing from a registered and logged in seller.

Promises

- Generates a modification request for the listing which is sent to property manager for approval.
- Generates a listing log entry at each stage of the listing modification process (updates and verification).

3.6.3 Structured English Description

The workflow of this process is described in the following structured English.

BEGIN

With listing modification request received

Send verification request to property manager

WHILE Verification result not received

Show verification pending

IF Listing modification permitted

Update listing details

Notify seller of listing update

ELSE
Notify seller of rejected modification
ENDIF
END

3.7 System Function 7 - Login

3.7.1 Description of Function

This process allows sellers, managers and administrators to login to the system in order to access the features associated with their account type. The Level 1 data flow diagram for this feature is shown in Figure 17.

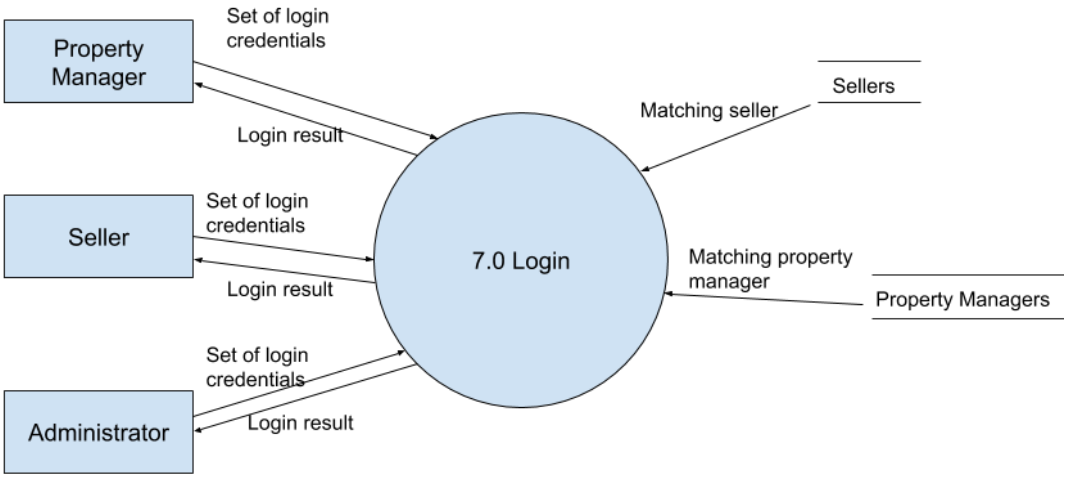


Figure 17: Login Level 1 DFD

3.7.2 Preconditions and Promises

Preconditions

- Needs a valid set of login credentials (username and password registered with the system).

Promises

- Provides the seller, manager or administrator with access to their dashboard with the features associated with their account type.

3.7.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 18.

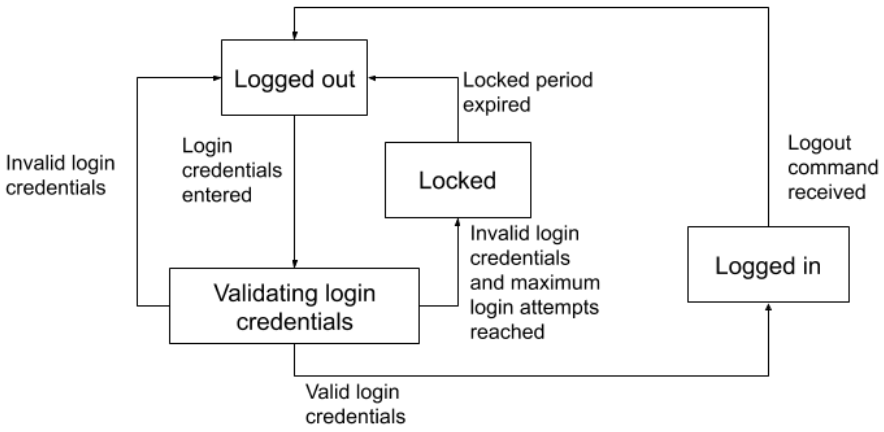


Figure 18: Login STD

3.8 System Function 8 - Listing Contract Renewal

3.8.1 Description of Function

This process allows sellers to extend their initial property listing contracts if the property has not been sold and seller chooses too. The Level 1 and 2 data flow diagrams for this feature are shown in .

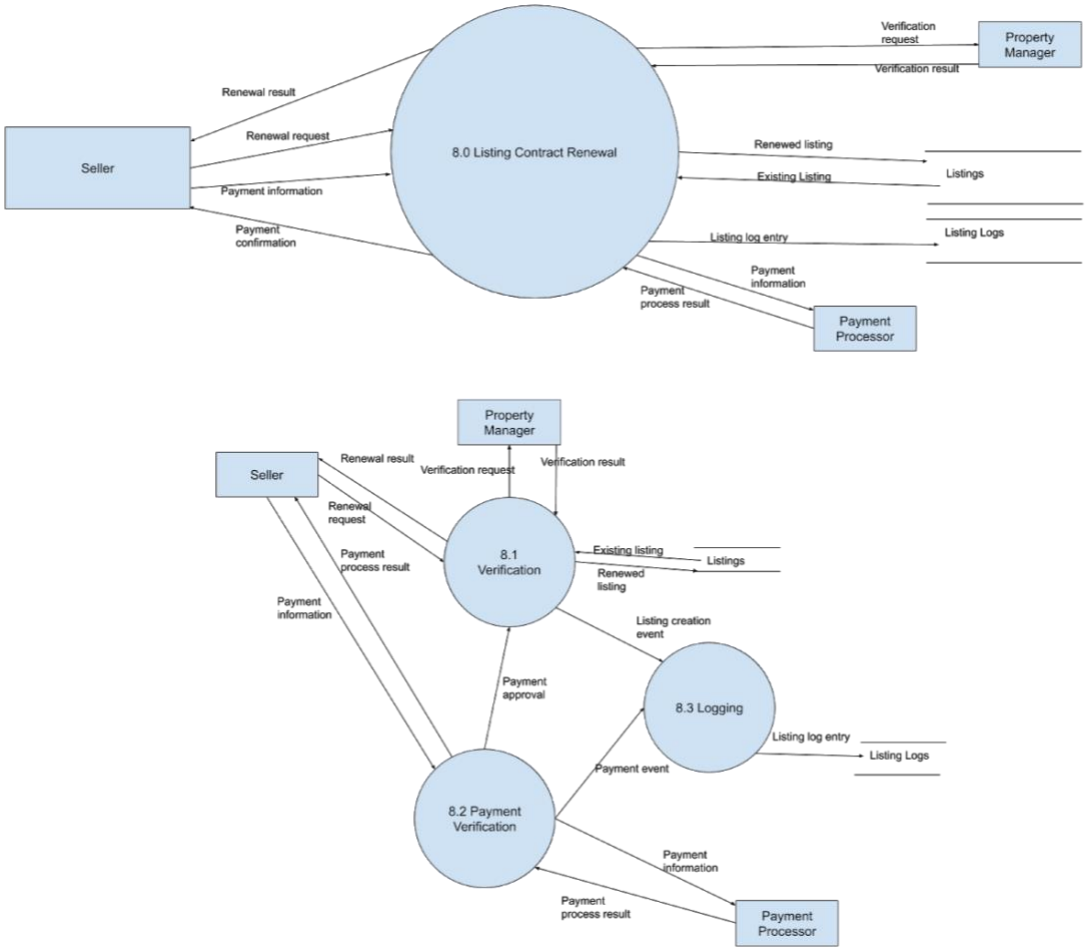


Figure 19: Listing Contract Renewal Level 1 & 2 DFD

3.8.2 Preconditions and Promises

Preconditions

- Needs an existing property listing from a registered and logged in seller.
- Needs a valid listing that is still available for sale and not marked as sold.

Promises

- Generates a renewal request for the listing which is sent to property manager for approval given that payment is approved.
- Generates a listing log entry at each stage of the listing renewal process.

3.8.3 Supplemental Diagram

The state transition diagram for this feature is shown in Figure 20.

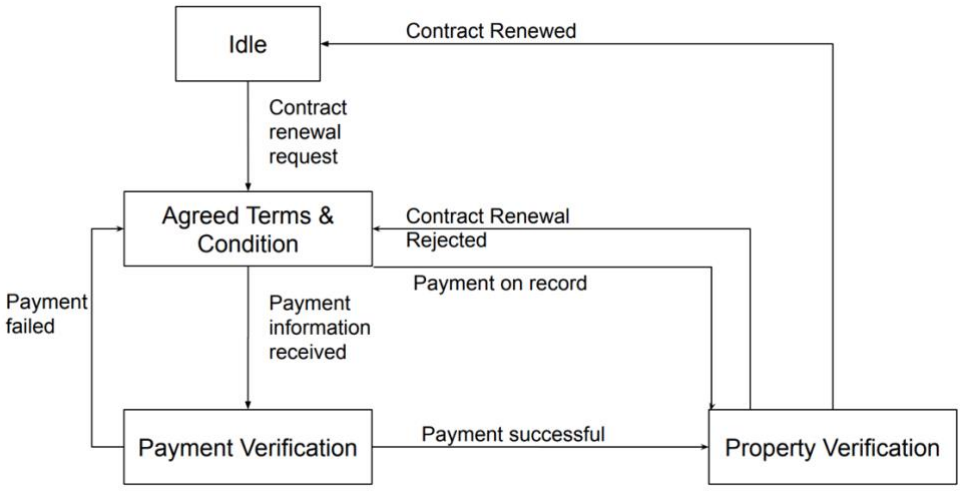


Figure 20: Listing Contract Renewal STD

4.0 External Interface Requirements

4.1 User Interfaces (UI)

The Interfaces for OBRM have been mostly inspired and derived from the purplebricks.ca, an industry leader in owner listed and fixed fee property listing platform.

4.1.1. User Registration

The first step of the application is User Registration. In this step sellers can register for an account on the application, and only through their registered account will they receive access to create and manage property listings. Property managers and administrators do not need to register for an account through the application, but property managers have their accounts created and modified by administrators directly. Sellers must provide a valid email and password in order to create an account.

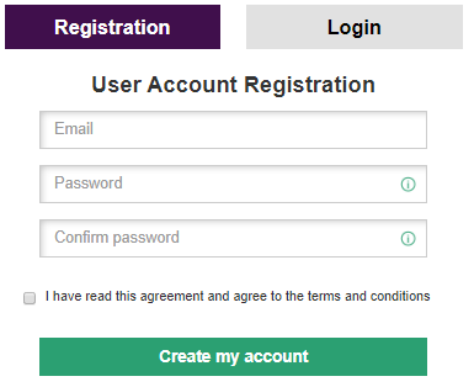


Figure 21: User Account Registration UI

4.1.2. Login

Once sellers have successfully registered for an account, they can use this feature to log in to the application. If the login is successful, the seller is redirected to the seller dashboard page. The login feature is identical for sellers, property managers and administrators. However, property managers and administrators are not required to register through the previous user registration interface as they are externally assigned their predefined user ID and password, which they can use here on the login page to access their own account to gain access to features such as property verification and report generation.

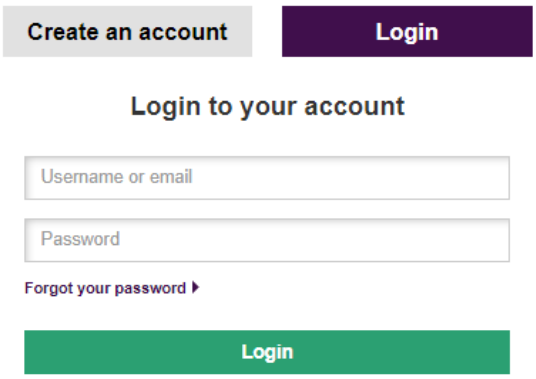


Figure 22: User Login UI

4.1.3 Property Listing Creation

The property listing creation interface is where a seller can create a listing and add property details as required/desired. The application has multiple tabs on this page including Contact Information, Location, Description, Property Images, Rooms and Cost. There are various mandatory fields such as address and cost of property that sellers are required to provide when creating a property listing, in addition to numerous optional fields that can be filled in as desired.

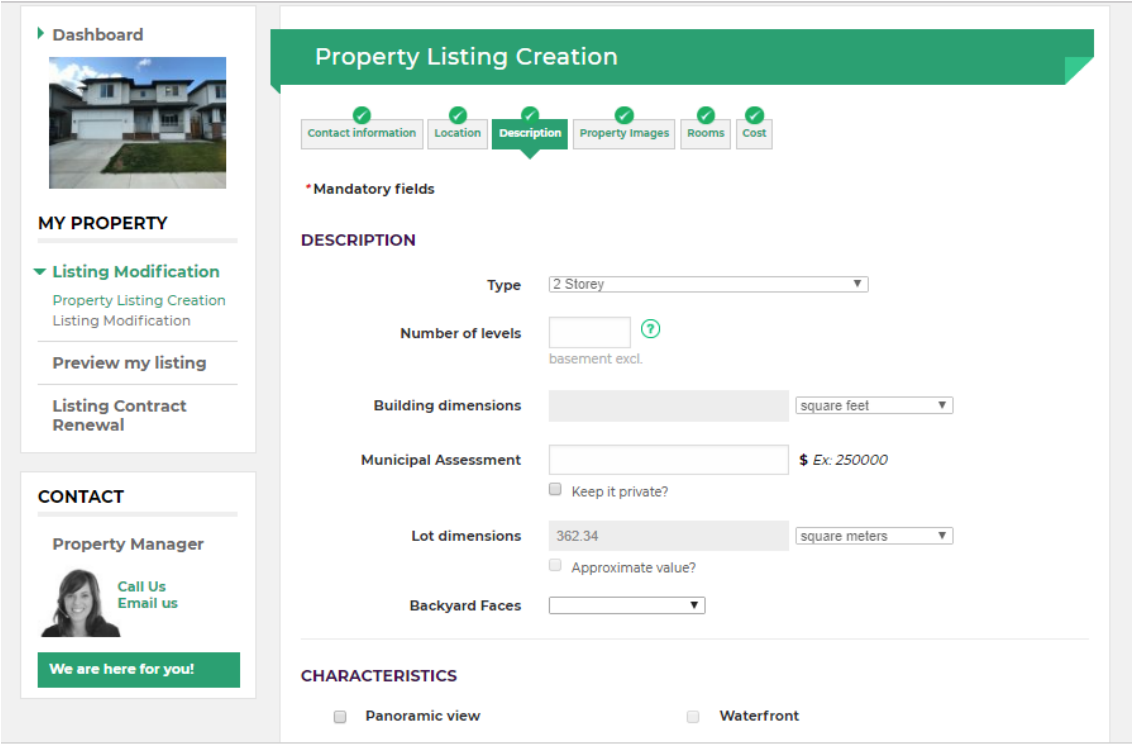


Figure 23: Property Listing Creation UI

4.1.4. Listing Modification

This interface provides sellers with the option of modify an existing property listing. A seller can update information through this interface such as address, image, description and cost of the property. If the seller has updated any critical information such as the address, the listing must be re-verified by a property manager before the listing will be re-published on the application.

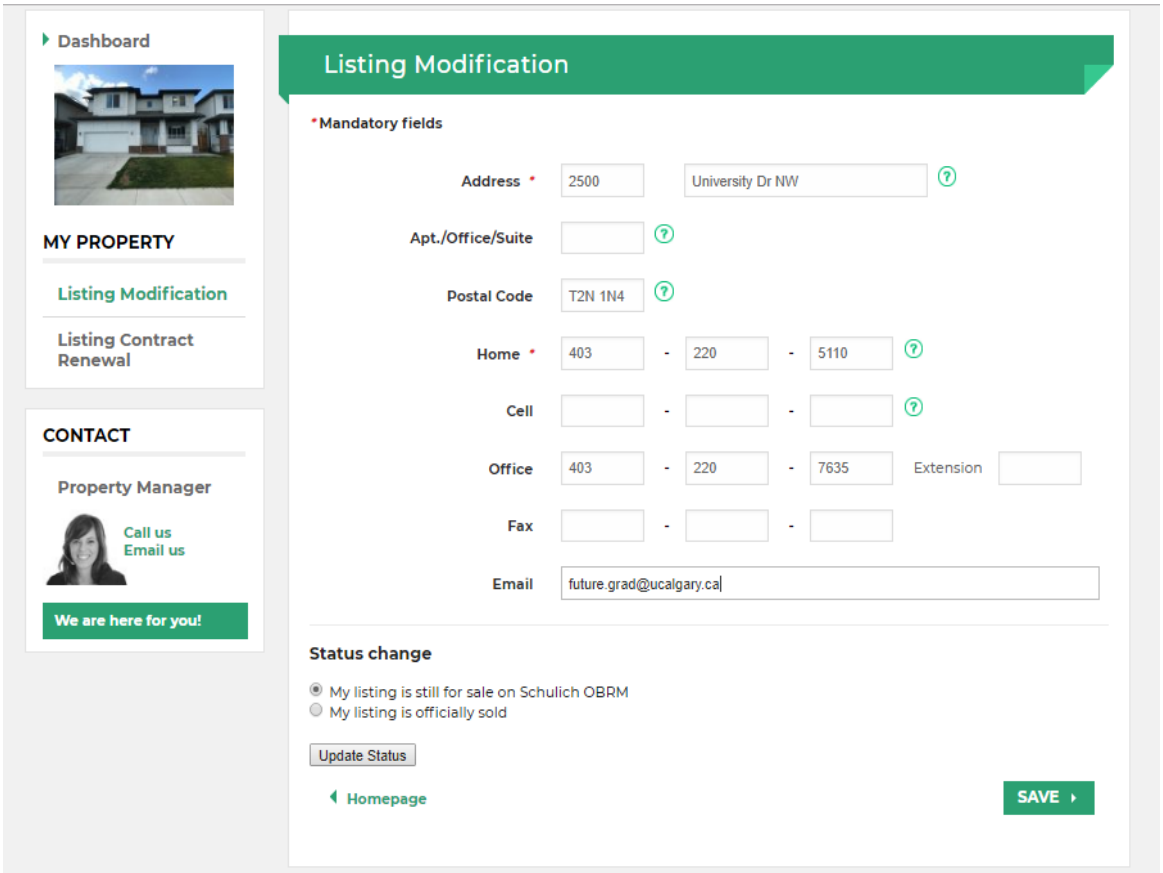


Figure 24: Listing Modification UI

4.1.5. Listing Contract Renewal

The interface for listing contract renewal is one which allows sellers to extend the duration of their listing contract. The initial contract for a listing is established when a seller creates a property listing and can be anywhere in the range of six months to one year, where the listing will be published and will be visible to buyers only for the duration of this contract. Through this interface a seller can see information about their contract start and end date and have an option to extend the contract. If a seller chooses to renew their contract, their contract will be extended to a selected duration and they will be charged payment corresponding to the extended contract duration.

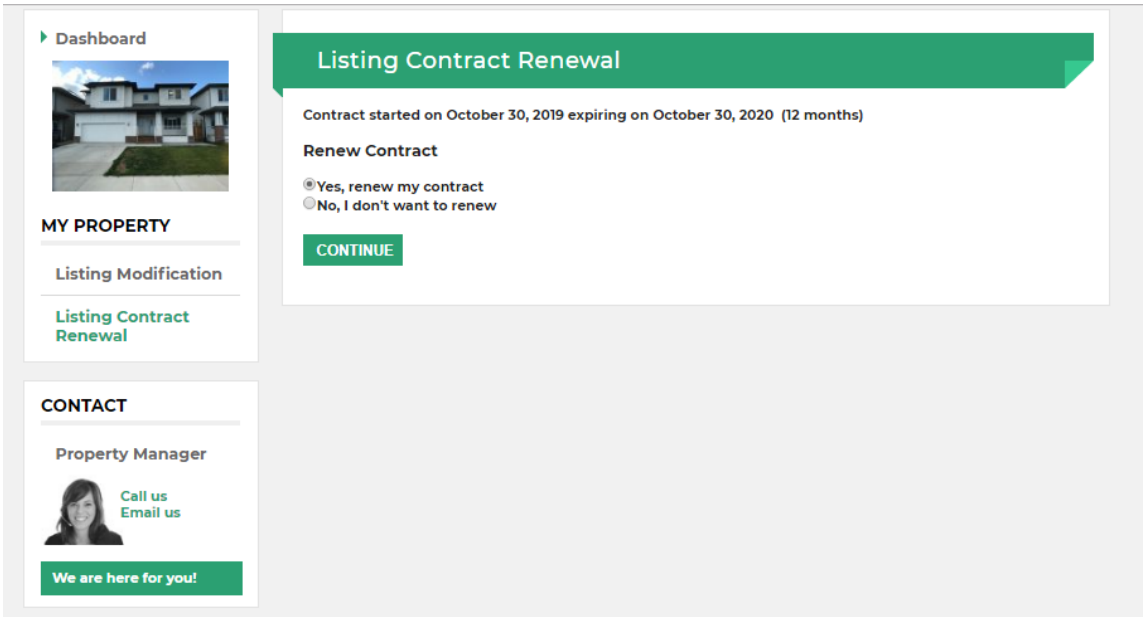


Figure 25: Listing Contract Renewal UI

4.1.6. Property Search Engine

The property search interface allows buyers to browse and filter properties through a search engine available on the homepage of the application. The buyer can provide specific search criteria such as the type of property, number of bedrooms and bathrooms, size of living area and other parameters in order to filter search results. The application could also be extended in the future to connect to an MLS, and include this as a search parameter.

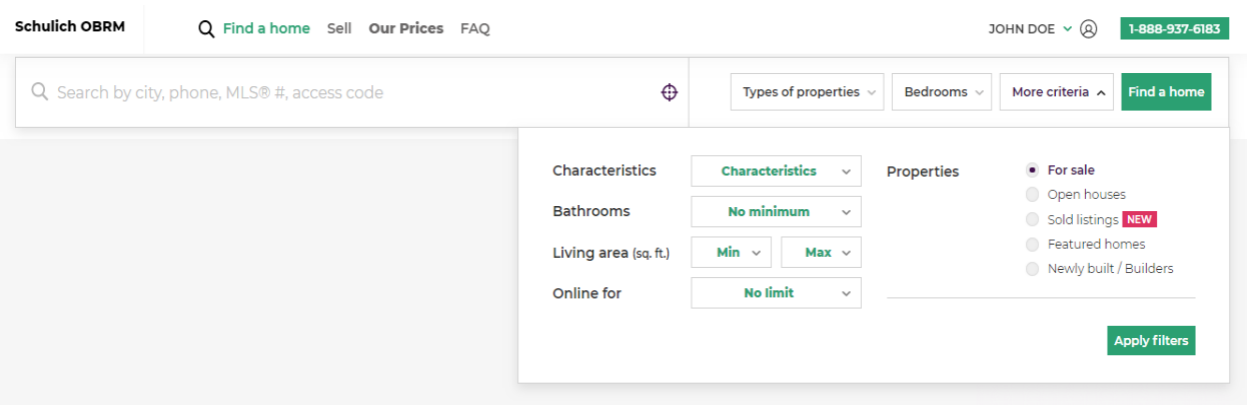


Figure 26: Property Search Engine UI

4.1.7. Property Listing Interface

The property listing interface consists of a displayed list of properties. Each property is shown has a displayed card containing an image, the cost, address and description of the property.

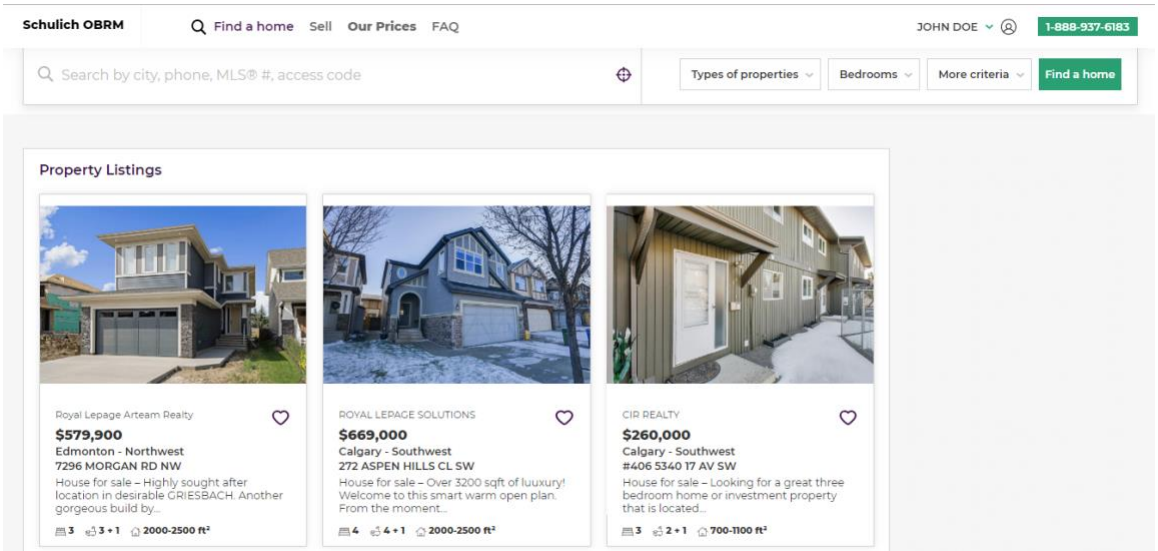


Figure 27: Property Listing UI

4.1.8. Property Showcase

The property showcase interface will be used by both sellers and buyers. When a buyer selects a property listing to view from the property listing interface, they will be taken to a showcase page containing more details on the property based on the information provided by the seller in the listing. When sellers create or manage their property listing, they will have an option to preview it to view this interface, allowing them to see how the listing will be shown to buyers through the same showcase page.

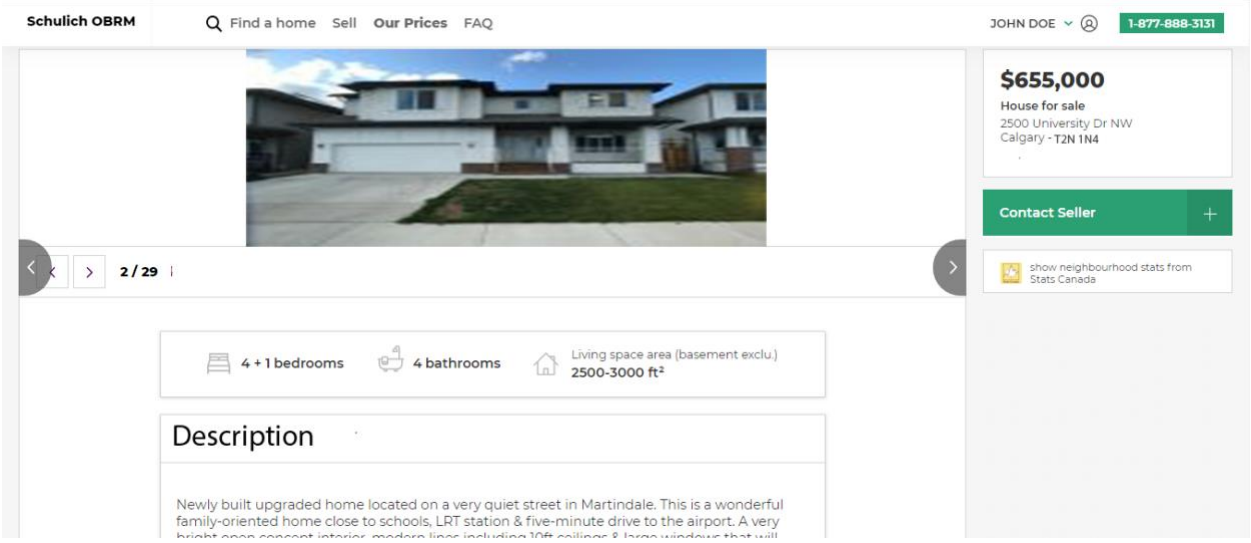


Figure 28: Property Showcase UI

4.2 Hardware Interfaces

The application will run on both Computer (Web) and Mobile platforms. Since the application's back-end components are hosted on AWS (Cloud platform), in-house physical servers are not required. An active internet connection is required for the application, so communication devices such as a modem and routers will be required when sellers, buyers, property managers and administrators use the application.

4.3 Software Interfaces

For the front-end components, HTML 5, CSS 3 and JavaScript version 2018 will be used. For the back-end components, AWS will be used, including Amazon's EC2 which provides secure and resizable computing capacity, AWS Lambda to run functions over the cloud in real time without making any changes in the front-end application, AWS Identity and Access Management (IAM) service to differentiate between user roles such as Seller, Buyer, Property Manager and Administrators in the application, AWS Relational Database (RDS) to store all data of the application and AWS S3 bucket to store images, videos and other files. The front-end and back-end components communicate with each other over HTTP. The application will also use third party API services from Statistics Canada to obtain metrics displayed in property listings. Most of the data that will be shared across software components will be in the JSON format and will be pertaining to properties.

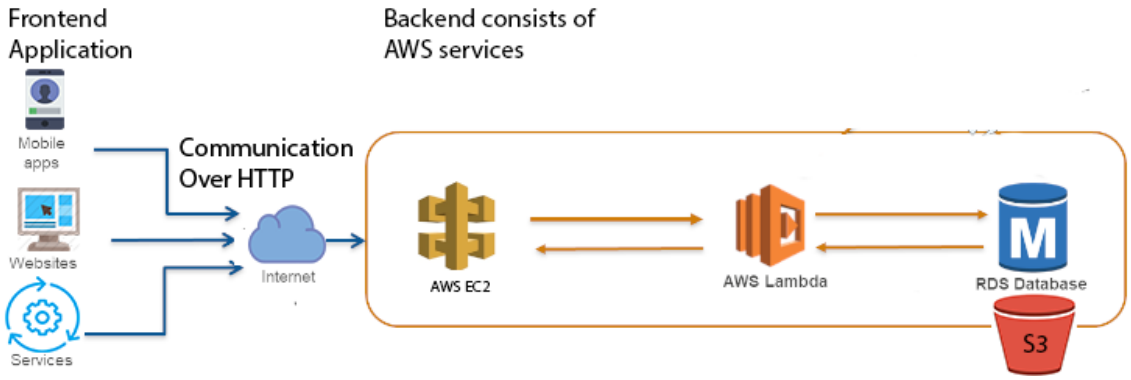


Figure 29: Software Interface Application

4.4 Communications Interfaces

The application will use Email and SMS services to notify buyers and sellers, which can be accomplished by AWS SNS notification services. As noted previously, the application uses web browsers and Android/iOS platforms to run. In order to protect the seller's critical information such as property address and contact information, the application encrypts data prior to saving it in the database or transmitting it between the front-end and back-end components. Since the application uses AWS S3 bucket for data storage, the data transfer rate will be faster than other

typical servers as Amazon automatically uses a server/data centre region spatially near the user's request location.

5.0 Non-Functional Requirements/Quality Requirements

Since this is an application designed to be used by the general public for buying and selling properties, the UI design and overall flow of use must be competitive with other, similar, applications. Interfaces that interact with the user must be straightforward, uncluttered, but have thorough explanations when requested by the user to minimize confusion and misunderstanding between the user and the system.

Security of data transfers during the payment transaction phase during property listing creation and contract renewal are of utmost importance. Since this is a system hosted on AWS and interacts with payments using Stripe, all designs regarding payment must comply fully with AWS and Stripe API guidelines to avoid unauthorized access of customers' payment information. Security of data transfers during the user login stage is also of utmost importance as sensitive information is stored and accessed during this stage. Sensitive information such as payment info and user passwords must NOT be logged.

6.0 Prioritization and Release Plan

6.1.1 Functional Requirement Prioritization Method

In order to prioritize the functional requirements of this OBRM system, a simple stakeholders' voting approach method is used, where the priority of each feature is defined as either high, medium or low based on its business importance. This approach is also combined with consideration of the mutual dependencies of each feature, as many of the features described in Section 3 have preconditions that depend on the working functionality of other features.

6.1.2 Feature Priorities

The priority of each functional requirement is listed in Table 1.

Table 1: Functional Requirement

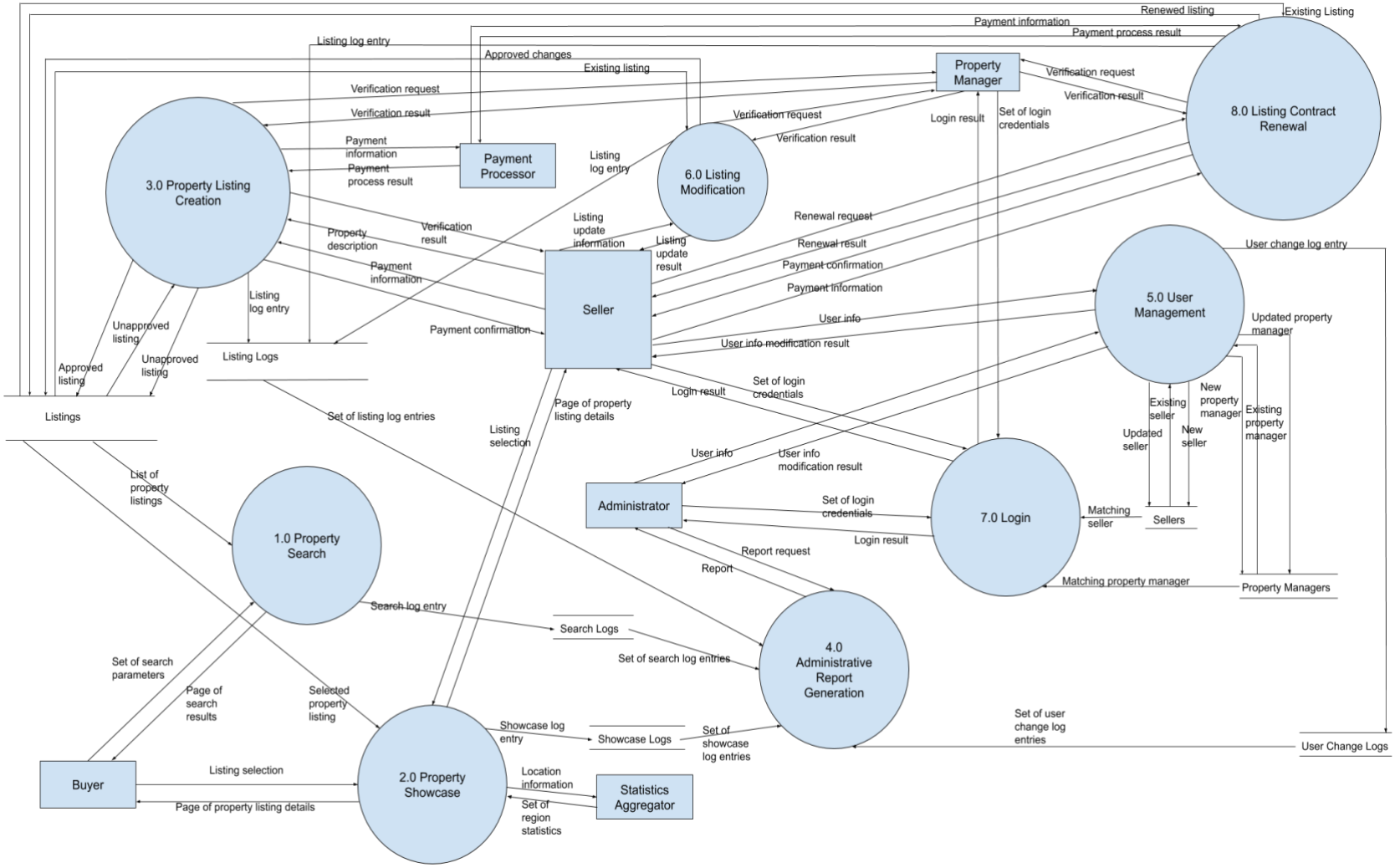
System Feature	Priority Level	Feature Type	Release Date	Remarks
1 - Property Search	High	F	February 14, 2020	The property search feature is a high priority feature as it provides the primary method through which buyers can access listing

				showcases via feature 2. Although it may be possible to have a workaround where buyers would simply browse a list of all property listings prior to having the search feature implemented, this would not be very useful for many buyers, which in turn would reduce demand for sellers to create listings via feature 3.
2 - Property Showcase	High	F	January 31, 2020	The property showcase feature is a high priority feature as buyers having access to this feature provides the primary motivation for sellers to list properties via feature 3.
3 - Property Listing Creation	High	F	January 10, 2020	The property listing creation feature is a high priority feature as it is the core functionality of the system which produces revenue.
4 - Administrative Report Generation	Medium	F	March 1, 2020	The administrative report generation feature is a medium priority feature as it is possible to have the system function without it, but it will still be very important in detecting system issues and for future extensions to the business application of the system (e.g. targeted marketing based on viewing statistics).
5 - User Management	High	F	December 1, 2019	The user management feature is a high priority feature as feature 7 requires a registered account to login to.
6 - Listing Modification	Medium	F	March 12, 2020	The listing modification feature is considered a medium priority feature as the seller's ability to modify or remove their listings is important in keeping listing showcases relevant and correct for buyers, but can be worked around initially until developed (e.g. having the search feature display newer listings first to reduce the number of

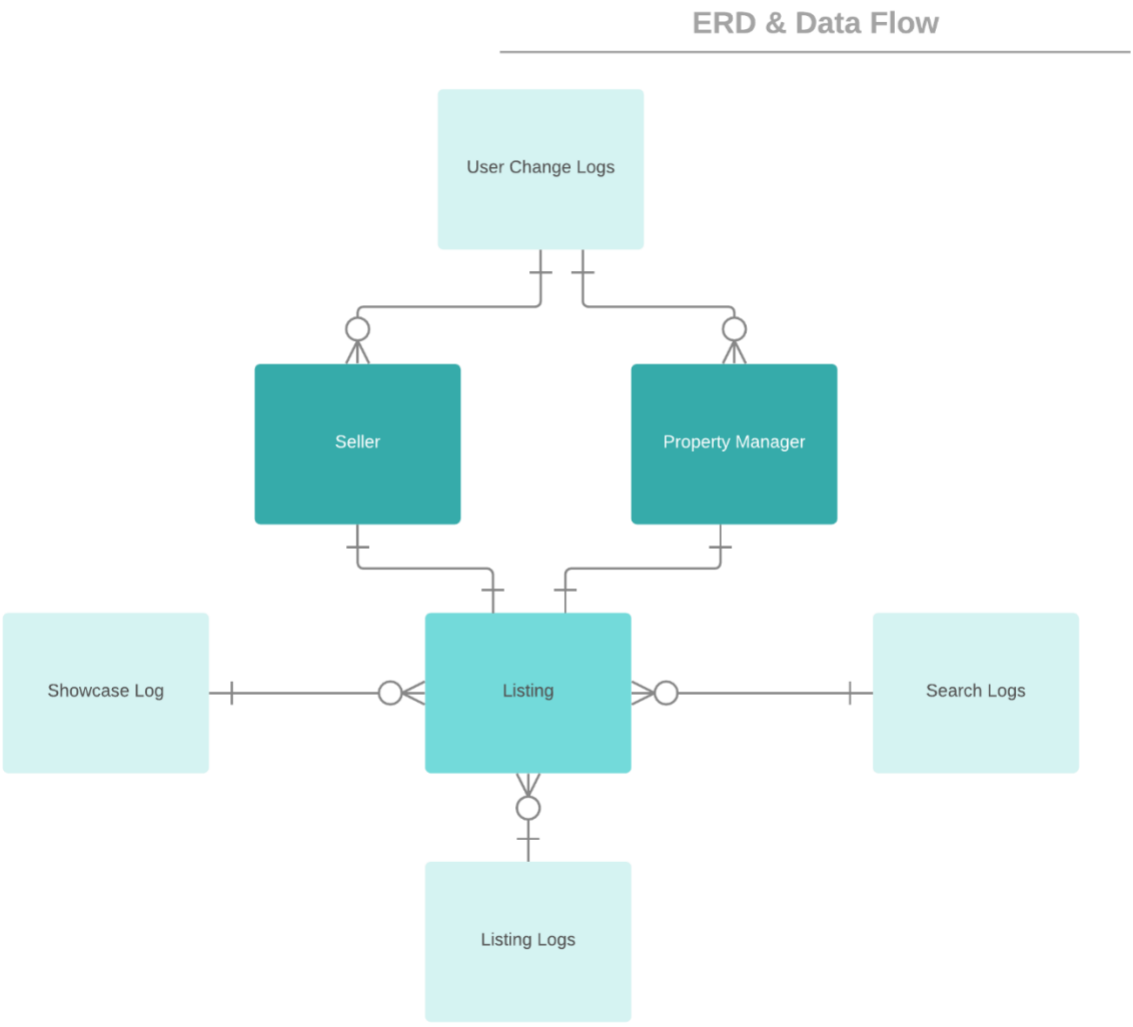
				outdated listings buyers encounter; having the listing creation feature notify sellers that they will be initially unable to modify their listing information)
7 - Login	High	F	December 14, 2019	The login feature is a high priority feature as features 3, 4, 6 and 8 depend on having a logged in seller or manager account.
8 - Listing Contract Renewal	Low	F	March 29, 2020	The listing contract renewal feature is a low priority feature as it provides a user experience enhancement in allowing sellers to keep property listings posted without having to re-create them but is not a necessary system function.

7.0 Appendices

Appendix A: System Level 1 DFD



Appendix B: Entity Relation Diagram



Appendix C: Data Dictionary

System Feature	Data	Description
	Contact information	Text: First name Text: Last name Text: Phone number Text: Email address
	Payment information	Integer: Credit card number (16 digits) Text: First name Text: Last name Integer: Expiry date (4 digits, MMYY) Integer: Security code (3 digits)
	User info	Contact information: User contact information User type: Seller, Manager, or Administrator Text: Username Text: Password
	Property Listing	Text: Location (Address) Text: Description Integer: Number of bedrooms Integer: Size of property (sq.ft.) Text: More Features Contact information: Seller contact information Photo [ ]: Photos of property Integer: Price Text (hidden): Listing approval state (unapproved, modified, approved, expired, etc.)
	Payment process result	Text: Current time Text: Transaction ID Text: Payment successful/failed
1	Set of search parameters	Text: City (or locality definition, could be Community, Postal Code, etc.) Text: Type of Property (House, Townhouse, Condo, etc.) Integer: Number of bedrooms Integer: Size of property (sq.ft.) Text: More Features (Walkout basement, finished basement, etc.)

		Price Range: Integer: Lower limit Integer: Upper limit
1	List of property listings	Property Listing: Listings that matched keys generated by search parameters and is currently approved
1	Page of search results	Search result: Summary of a property received in list of property listings
	Search result	Text: Location (Address) Integer: Number of bedrooms Integer: Price Text: Description (First 200 characters) Photo: First photo in the list of photos in the property listing
1	Search event	Text: IP address of access Text: A text representation of Set of Search Parameters
1	Search log entry	Text: Current time Text: A text representation of the search event generated for log entry
3	Property description	Property Listing: A rough draft of a property listing.
3	Payment confirmation	Payment process result: the payment process result forwarded from the payment processor
3	Payment approval	Boolean: Payment successful/failed Text: Transaction ID
3	Unapproved Listing	Property Listing: A property listing not yet approved by a property manager
3	Verification request	Text: Property listing address Contact information: Seller contact information
3	Verification result	Text: Property listing address Contact information: Seller contact information Contact information: Property manager contact information Text: Listing approved/rejected Text: Description of rejection, empty if listing approved

3	Approved Listing	Property Listing: A property listing approved by a property manager
3	Listing creation event	Text: IP address of access
3	Payment event	Text: IP address of access Text: Access username (seller) Text: Transaction ID Text: Payment successful/failed
3	Verification event	Text: IP address of access Text: Access username (manager) Text: Listing approved/rejected
3	Listing log entry	Text: Current time Text: A text representation of the listing creation event, payment event, or verification event generated for log entry
5	Updated Seller	User info: Updated user info
5	New Seller	User info: New user info
5	Existing Seller	User info: User info with matching username to search
5	User info modification result	User info: Modified user info Text: Modification successful/failed
5	User info modification event	Text: IP address of access Text: Modified username Text: Modification action successful/failed
5	User change log entry	Text: Current time Text: A text representation of the user info modification event generated for log entry