CSC 431 – <SPRING 2025>

<QuantEstate>

Software Requirements Specification (SRS)

<Team number>

Eddy Boris	Product Manager
Anish Guntreddi	Lead Developer
Diego Herrera	Scrum Master

Version History

Version	Date	Author(s)	Change Comments
1.0	02/20/25	Eddy Boris, Anish Guntreddi. and Diego Herrera	Created and Initialized our ideas
1.1	02/22/2025	Eddy Boris , Anish Guntreddi, and Diego Herrera	Changed some technicalities and system constraints
1.2	02/23	Eddy Boris, Anish Guntreddi, and Diego Herrera	Did a brief overview of final results and changed any errors or spelling mistakes

Table of Contents

1. System Requirements

1.1 Functional Requirements

< List all functional requirements in the following example format >

1.1.1 Data Collection and Integration

Title	Data Collection and Integration
Description	Pulling Real estate listing information and historical prices and trends from the housing market along with mortgage data and interest rate data
Priority	0
Precondition(s)	Need to set up a python virtual environment with all necessary dependencies for obtaining, cleaning, and training data.
Basic Flow	Use APIs to pull real estate listing data from Zillow or another housing source and use pandas to create a dataframe for this in python. Next use pandas to clean data and use numpy for numerical data on the housing market along with interest rates and mortgages to train ML models.
Postconditions(s)	We obtain the result of a data frame consisting of all important data needed to train a ML model or potentially a deep learning model.
Use Case Diagram	3.1.1

1.1.2 App Initialization

Title	App Initialization
Description	Next Step would be to initialize a reactJS or nextJs web application using node.js.
Priority	0
Precondition(s)	Need to setup NodeJS and then use either React or NextJS
Basic Flow	Install NodeJs and setup ReactJS or NextJS to get basic web template file and then work on creating an MVP frontend that meets basic needs for input data.
Postconditions(s)	This results in a working front end service for the user to be able to input data
Use Case Diagram	<Link or number, if present>

1.1.3 Data Analysis and ROI Computation

Title	Data Analysis and ROI Computation
Description	Analyze the cleaned real estate data to compute returns on investment (ROI) based on factors like price trends, interest rates, rental yields, and local market conditions
Priority	1
Precondition(s)	Processed dataset from Data Collection and Integration is available. Statistical and machine learning libraries (scikit-learn, TensorFlow)
Basic Flow	Retrieve the cleaned dataset from the storage. Apply ML model to identify trends and patterns. Calculate potential ROI based on various market scenarios (e.g., best case, worst case). Return computed ROI metrics to be displayed or stored.
Postconditions(s)	ROI metrics for each property are generated. These metrics are stored or cached for faster access by the recommendation engine.
Use Case Diagram	3.1.3

1.1.4 User Account Management

Title	User Account Management
Description	Allow users to sign up, log in, and manage their profiles, saving specific property preferences and recommended houses.
Priority	1
Precondition(s)	Database or authentication service is set up (Firebase Auth or custom Node.js-based auth). Front-end is initialized to handle user form entries.

Basic Flow	User navigates to the sign-up or login page. User enters credentials; the system validates or creates an account. Users can update personal information or property preferences (location, budget range). Auth tokens or sessions are generated for secure access to the rest of the app.
Postconditions(s)	Users are authenticated and can access personalized recommendations. User preferences are saved for future data analysis and recommendation refinement.
Use Case Diagram	3.1.2

1.1.5 Recommendation Engine

Title	Recommendation Engine
Description	Offer personalized housing recommendations based on user preferences, ROI computations, and market trends.
Priority	2
Precondition(s)	ROI metrics from the Data Analysis and ROI Computation requirement. User's location, budget, and preference data from User Account Management.
Basic Flow	Retrieve user's profile and preferences from the database. Filter available properties based on location, price range, and property type. Sort or rank properties using ROI metrics and any machine learning recommendations. Return top recommendations to the user interface.
Postconditions(s)	A curated list of properties with the highest potential return and relevance to the user is displayed.
Use Case Diagram	3.1.4

1.2 Non-Functional Requirements

< List all non-functional requirements in the following example format >

1.2.1 Performance and Reliability

Title	Performance and Reliability
Description	The system must handle data collection, analysis, and recommendation requests within acceptable time limits and operate reliably under varying loads.
Priority	0
Applicable FR(s)	Data Collection and Integration, Data Analysis and ROI Computation, Recommendation Engine

1.2.2 Security and Privacy

Title	Security and Privacy
Description	User credentials and personal data must be stored securely, ensuring confidentiality and integrity. Best practices for data security (e.g., encryption, HTTPS) must be followed.
Priority	0
Applicable FR(s)	User Account Management, Recommendation Engine (handles user data)

1.2.3 Usability and Accessibility

Title	Usability and Accessibility
Description	The user interface should be intuitive, with consistent layout and design. The application should comply with common accessibility standards.
Priority	1
Applicable FR(s)	App Initialization, User Account Management, Recommendation Engine

1.2.4 Scalability

Title	Scalability
Description	The system should be able to handle an increasing number of users and expanding datasets (more properties, larger historical data) without significantly degrading performance.
Priority	2
Applicable FR(s)	App Initialization, User Account Management, Recommendation Engine

2. System Constraints

2.1 Tool Constraints

< List all tool constraints in the following example format >

2.1.1 Python Environment

Title	Python Environment
Description	We must use Python 3.8+ with libraries like pandas, numpy, scikit-learn and requests for data processing and analysis.
Priority	0

2.2 Language Constraints

< List all language constraints in the following example format >

2.2.1 Required Languages

Title	Required Languages
Description	The system must be developed using Python (for data processing, ml development and analysis) and JavaScript (React/Next.js) for the front end with Node.js for server-side or API endpoints
Priority	0

2.3 Platform Constraints

< List all platform constraints in the following example format >

2.3.1 Web Based Application

Title	Web Based Application
Description	The solution must run in modern web browsers (Chrome, Firefox, Edge, Safari) and be accessible over the internet.
Priority	1

2.4 Hardware Constraints

< List all hardware constraints in the following example format >

2.4.1 Minimum Server Requirements

Title	Minimum Server Requirements
Description	A server with at least 4 CPU cores, 8GB RAM, and sufficient storage to handle the data volume. GPU acceleration optional but helpful for large scale ML development/
Priority	1

2.5 Network Constraints

< List all network constraints in the following example format >

2.5.1 Internet Connectivity

Title	Internet Connectivity
Description	Reliable high-speed internet is required for data collection (APIs and web scraping) and for users to access the web app.
Priority	1

2.6 Deployment Constraints

< List all deployment constraints in the following example format >

2.6.1 Cloud Deployment

Title	Cloud Deployment
Description	The system must be deployable to a cloud environment (AWS, GCP, or Azure) to allow scaling, efficient resource allocation and metrics.
Priority	2

2.7 Transition & Support Constraints

< List all transition & support constraints in the following example format >

2.7.1 Maintenance and Hand Off

Title	Maintenance and Hand Off
Description	Clear documentation, version control, and CI/CD pipelines much be established to allow easy transition to new development teams or maintenance teams as company scales.
Priority	2

2.8 Budget & Schedule Constraints

< List all budget & schedule constraints in the following example format >

2.8.1 Development Timeline and Funding

Title	Development Timeline and Funding
Description	The project must be completed within the set timeframe and within the allocated budget for API subscriptions, cloud computing costs and developer hours
Priority	0

2.9 Miscellaneous Constraints

< List all miscellaneous constraints in the following example format >

2.9.1 Legal and Regulatory Compliance

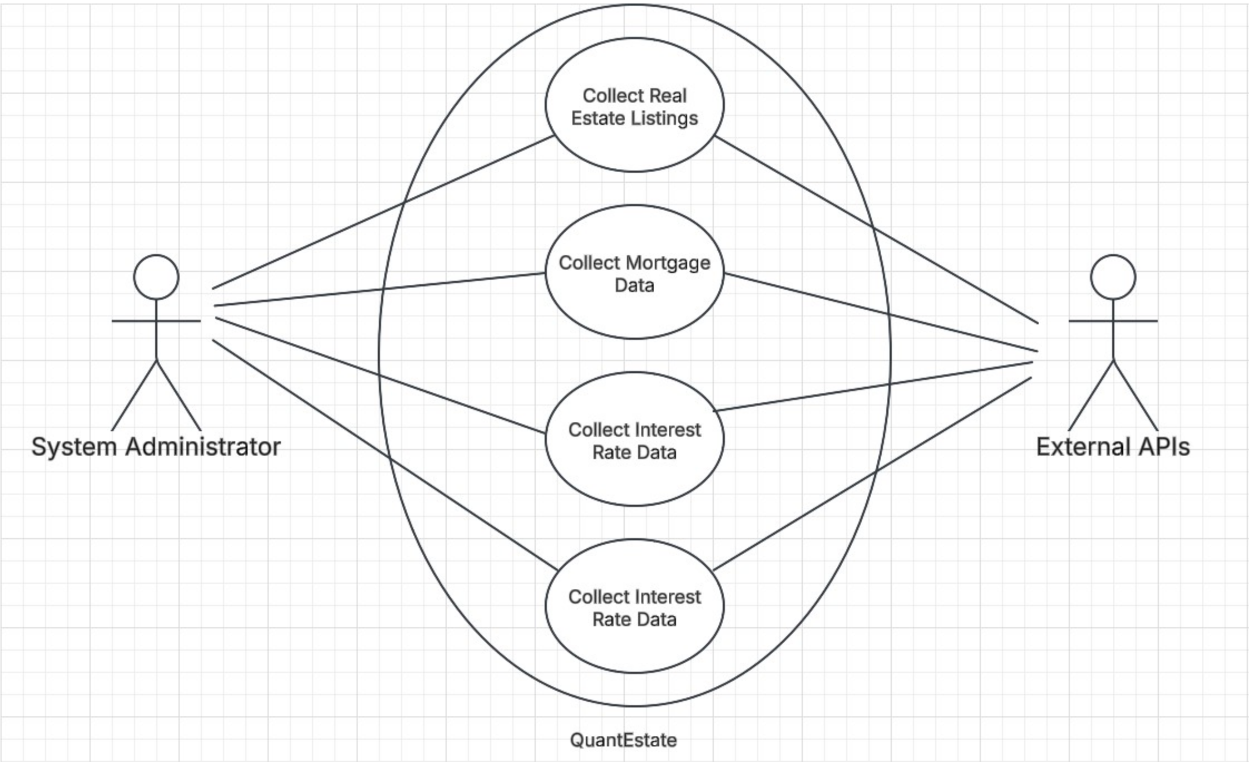
Title	Legal and Regulatory Compliance

Description	The system must comply with relevant data privacy laws and usage constraints for each data provider’s terms of service
Priority	0

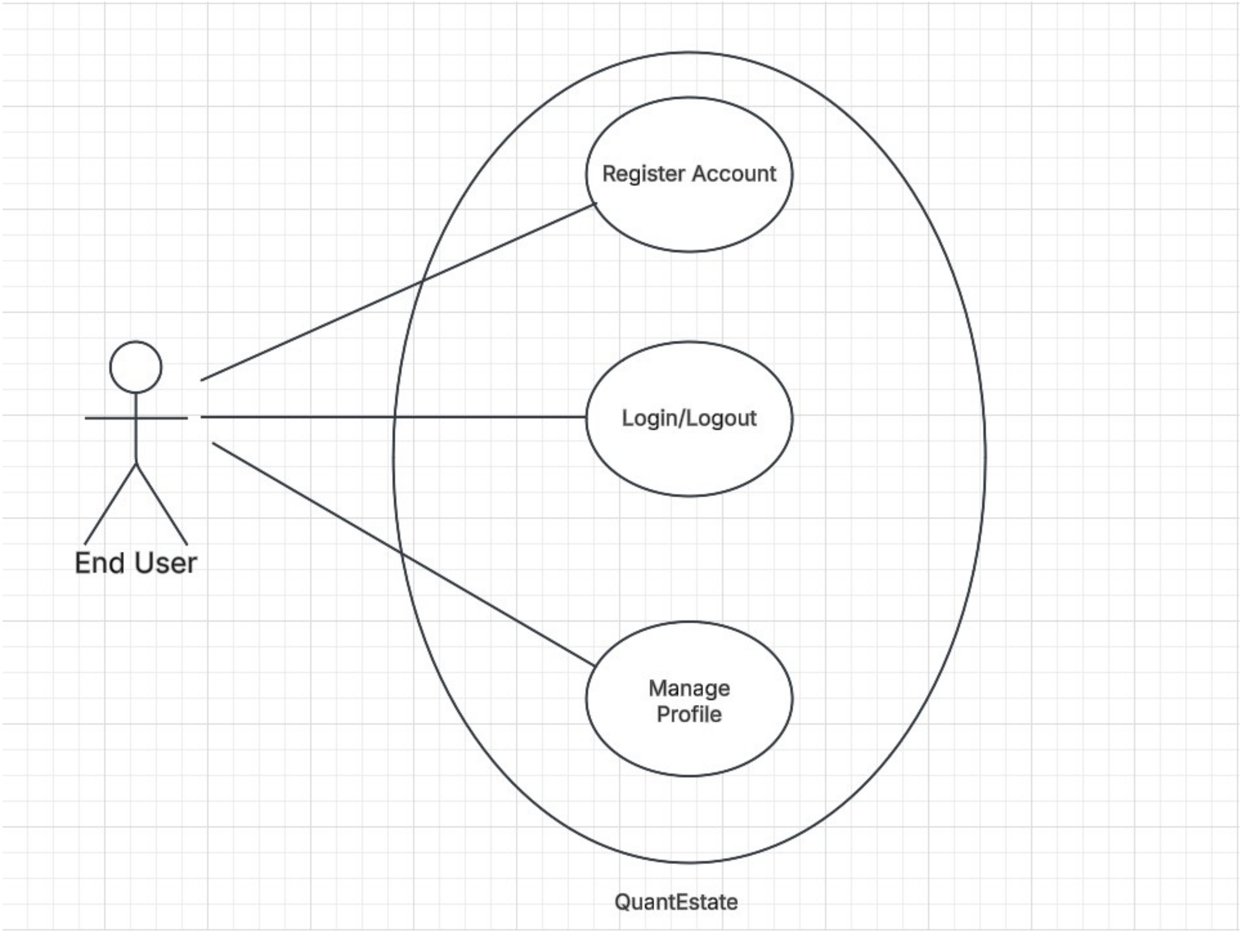
3. Requirements Modeling

< List all Use-case diagrams for the functional requirements in the following format>

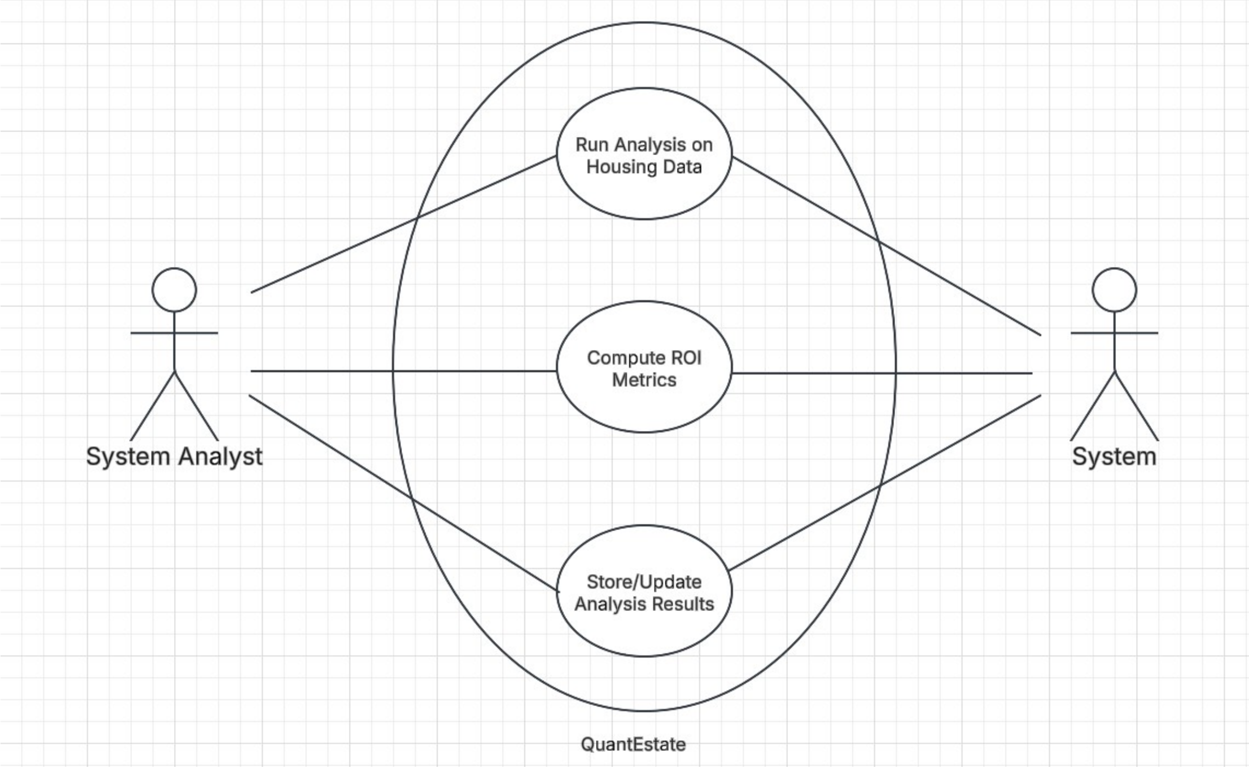
3.1.1 Data Collection and Integration Use-case


Collect Real Estate Listings
Collect Mortgage Data
Collect Interest Rate Data
Collect Interest Rate Data
System Administrator
External APIs
QuantEstate

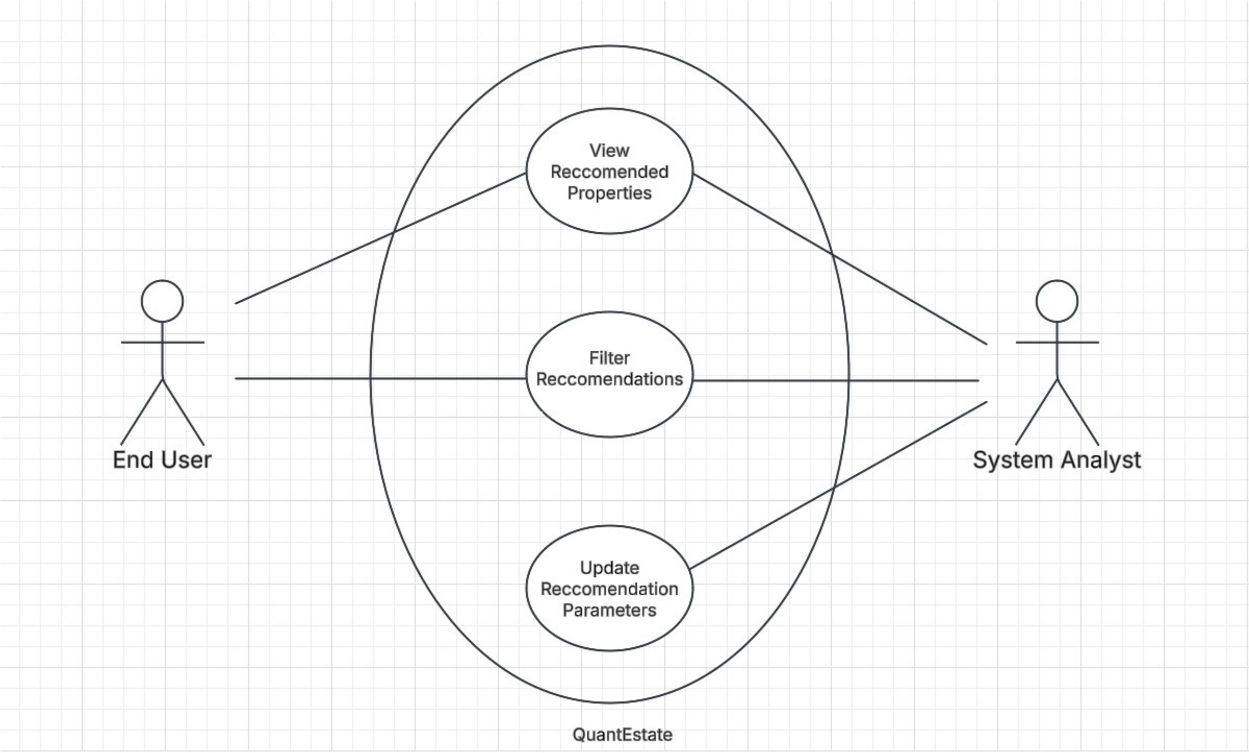
3.1.2 User Account Management Use-case



3.1.3 Data Analysis and ROI Computation Use-case



3.1.4 Recommendation Engine Use-case



4. Evolutionary Requirements

4.1 Functional Requirements

< List all functional requirements in the following example format >

4.1.1 Advanced Predictive Analytics

Title	Advanced Predictive Analytics
Description	Future updates may include advanced time-series forecasting and deep learning models for improved ROI Predictions
Priority	3
Precondition(s)	A large historical dataset is accumulated to train deeper models. Additional processing power or specialized hardware (GPU) is available
Postconditions(s)	More accurate or robust house price forecasts and ROI estimates.
Use Case Diagram	<Link or number, if present>

4.1.2 Broadening Product Offerings

Title	Broadening Product Offerings
Description	Future updates may include broadening the options available for users on our web app. This includes offerings such as land, cars, planes, etc.
Priority	3
Precondition(s)	A new dataset detailing interest rates and statistics to train additional ML models on new product offerings
Postconditions(s)	Increase in uses within a single web application.
Use Case Diagram	<Link or number, if present>

4.2 Non-Functional Requirements

< List all non-functional requirements in the following example format >

4.2.1 Multi Language Support

Title	Multi Language Support
Description	The application may need to support multiple languages for wider accessibility.
Priority	4
Applicable FR(s)	App Initialization, User Account Management, Recommendation Engine

4.2.1 Increased Security Services

Title	Increase Security Services
Description	As the application grows the security of our users must remain scalable. With increases in revenue investments should be made into increased security to ensure users remain trusting of the application.
Priority	4
Applicable FR(s)	App Initialization, User Account Management