

IFix cars dealership scenario.

1. Verification.

1.1. Feasibility study. **Business Analysts team.**

Input.

1.1.1. Scope - IFix cars dealership.

Population	Market competitors in car dealership industry.	Example: Autohaus24, CarWow, FirstMotors.
	Companies with similar functions.	Example: IAAI, Manheim, Ebay.
	Target population.	Users of competitor's services.
Audience	Target audience.	Males 20-60 y.o, car enthusiasts
Audience sampling.	Users.	10 clients of Autohaus24, 5 clients of CarWow, 10 clients of FirstMotors.
	Experts.	2 admins, 1 database expert, 3 backend experts, 2 frontend experts, 1 QA expert, 2 car experts.

1.1.2. Client and system specification.

Client specification.	Business type.	Digital platform for car dealership "IFix".
	Objective.	Gain more clients, increase competitive ability, increase income and sales.
System specification.	Software.	Database management system, payment API, measurements management.
	Hardware.	Database, banking system(server), server.
	Normal behavior.	NB is defined from Threshold --> response time < 5 sec.

1.1.3. List of operational and technical issues.

Operational issues.	Technical issues.
Examples of Operational issues: <ul style="list-style-type: none">• slow performance• security issues• customization problems	Examples of Technical issues: <ul style="list-style-type: none">• compatibility issues• database issues• network connectivity problems• payment API problems

Quantitative analysis – questionnaire to users.	Qualitative analysis – questionnaire to experts.
Examples of questions: <ul style="list-style-type: none"> • What users expect from our software? • What are features that users like/dislike in competitor's software products? • What features users want to see in our software? 	Examples of questions: <ul style="list-style-type: none"> • What can be done to make best possible system integration? • How to avoid fatal errors and app crashes? • What can be done to increase security? • What are the most important metrics we can measure to be aware of app's condition and for further improvement?

1.1.4. Budget.

Client defined budget = 50000\$. Divide: 20% (10000\$) - software; 40% (20000\$) - hardware; 40% (20000\$) - quality assurance.

Client agreed and confirmed to Medium level of quality for the development. Agile SDLC is chosen for development.

Output.

1.1.5. Joint report.

<i>Function</i>	<i>Operational issues</i>	<i>Technical issues</i>
Customer relationship management	Customer service quality, lost sales opportunities, decreased customer satisfaction	Data integration problems, security vulnerabilities
Reporting and analytics	Identifying operational inefficiencies, compliance monitoring	Database performance problems, Data integration problems, improper set measurements
Financial management	Financial errors, data inaccuracy, customer dissatisfaction	Payment API vulnerability, security issues, upgrade and migration problems
Inventory management	Stock visibility errors -> item not in stock, difficulty in tracking inventory, order fulfillment issues	Database performance problems, improper data integration, scalability issues

1.1.6. Keywords and recommendations.

Keywords: Market analysis, scope – market competitors, budget, user, admin, technology assessment, feasibility criteria, population, audience, normal behavior, MVP, actual process, scalability.

Recommendations: Evaluate the estimated costs of development, implementation, and maintenance; obtain client specification; identify potential risks such as technical challenges, market competition, and regulatory changes; establish feasibility criteria; consider building a prototype or Minimum Viable Product; consider scalability of the system; evaluate normal behavior by setting the threshold.

1.2. Requirements analysis. **Software Design team.**

Input.

1.2.1. Revised Specification level 1 – client specification.

Features and functions: Sign up, Login, Buy car, Sell car, Edit car data, Edit profile, Search, Search history, View car offer, Filters, Wish list, P2P chat, Add dealership's car, Send notifications, View users, View metrics.

1.2.2. Revised Specification level 2 – system specification.

User types: Normal user, Admin.

a) *Normal user.*

Main functions: Sign up, Login, Buy car, Sell car, Edit car data, Edit profile, Search, Search history, View car offer, Filters, Wish list, P2P chat.

b) *Admin.*

Main functions: Add dealership's car, Send notifications, View users, View metrics.

Simulations for user input/output.

<i>User.</i>	<i>Function.</i>	<i>User input.</i>	<i>Expected output.</i>
Normal user	Sign up	New user -> sign up -> fill in email, username, and password -> create account	New user instance is created in database successfully.
	Login	User enters email and password -> login	Authorization successful, user entered his account.
	Buy car	User decides to buy car and presses "Buy" button	Order is created, user is redirected to payment page, payment API activated
	Sell car	User wants to sell own car -> fill in the form with detailed information about car	New car instance is created in the database. New car is listed as offer for other users.
	Edit car data	User wants to change some attribute of data of his car listed for sale	Instance of user's car is found in database -> particular attribute rewritten in database
	Edit profile	User wants to change some attribute of his personal information	Instance of user's profile is found in database -> particular attribute rewritten in database
	Search	User enters some keyword and searches	Search result based on searched keyword is displayed
	Search history	User views search history	Search history is successfully retrieved from database

	View car offer	User opens car offer to view	The page of this car offer is opened for view
	Filters	User implies some filters to the search	Search results refreshed based on the filtered attribute
	Wish list	User adds a car to wish list	New instance for wish list attribute is created in database in the user's instance
	P2P chat	User sends a message to another user via chat	Message is successfully delivered from one user to another. Second user receives notification about new message
Admin	Add dealership's car	Admin adds new car from dealership for sale	New car instance created in database. New car is listed as offer for users.
	Send notification	Admin sends in-app notification to all or some particular users	Notification is successfully delivered to users. Users can view the notification
	View users	Admin accesses user management section in database	Access granted, admin successfully enters section and views users
	View metrics	Admin views metrics of users or system	Access to database granted, admin is able to view metrics in database or log files

Necessary hardware: Database, banking system (server), web server.

Database: CPU - Intel Xeon, RAM – 64GB DDR5, Secondary storage – 20 TB HDD.

Web Server: CPU – AMD Ryzen, RAM – 32GB DDR5, Secondary storage – 5 TB HDD.

Banking Server: CPU – AMD EPYC, RAM – 128GB DDR5, Secondary storage – 3 TB SSD.

Quality control in Requirements analysis.

Level of Quality = Quality Planning (verify) + Quality Control (validate).

Quality characteristic: Availability -> Reliability, Performance, Accessibility, Security

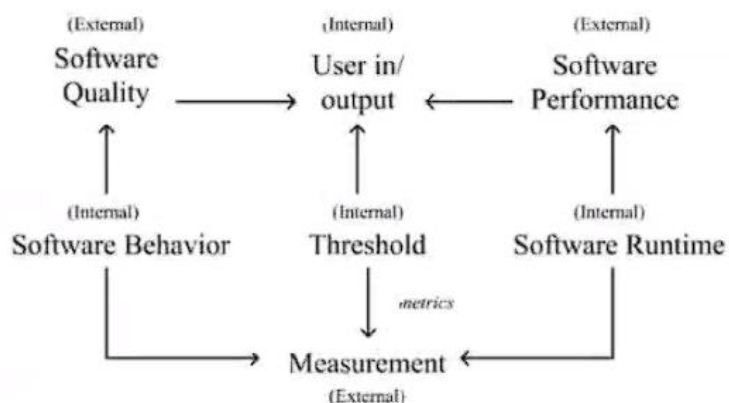
Quality planning -> requirement analysis -> verification with standard rules -> parameters -> metrics -> presume threshold to be measured -> code error -> solve errors.

Establish normal behavior. Functions are measured by:

- Soft operation – by session time (60 seconds per session).
- Hard operation – by threshold (0-5 seconds).

Batch processing table.

<i>User.</i>	<i>Function.</i>	<i>User Input.</i>	<i>Operation.</i>	<i>Database operation.</i>
Normal user	Sign up	Email, username, password	Soft – 10 min	Read, Write
	Login	Account ID	Hard – 5 sec	Read
	Buy car	Offer ID, account ID	Soft – 15 min	Read, Write
	Sell car	Account ID, new car instance	Soft – 5 min	Write
	Edit car data	Account ID, offer ID	Soft – 5 min	Read, Write
	Edit profile	Account ID	Soft – 5 min	Read, Write
	Search	Offer ID	Hard – 5 sec	Read
	Search history	Account ID	Soft – 10 min	Read
	View car offer	Offer page ID	Soft – 15 min	Read
	Filters	Offer attributes	Hard – 5 sec	Read
	Wish list	Account ID, wish list attribute	Hard – 5 sec	Write
	P2P chat	Sender account ID, receiver account ID	Hard – 3 sec	Read, Write
Admin	Add dealership's car	Admin ID, admin password, new car instance	Soft – 5 min	Write
	Send notification	Admin ID, admin password	Hard – 3 sec	Write
	View users	Admin ID, admin password, database access	Soft – 15 min	Read
	View metrics	Admin ID, admin password, database access	Soft – 15 min	Read



Software runtime and threshold are one of the main objectives of quality. Combined with Software behavior, they are main components of measurement.

Direct indicator of assessment is result of action. Indirect indicator – cause of an action.

Table of direct, indirect assessment and direct, indirect measurement.

Function.	Direct assessment	Indirect assessment	Direct measurement	Indirect measurement
Sign up	Connectivity, security, availability, accessibility, performance, reliability	Error rate: 0.02 Response time: 300 ms Data loss: None	Session time – 10 min Threshold – 5 sec	Error rate > 2% = error -> defect Data loss > 0% = error -> defect Session time > 10 min = error -> defect Threshold > 5 sec = error -> defect
Login	Connectivity, security, availability, accessibility, performance, reliability	Error rate: 0.05 Response time: 600 ms Data loss: None	Threshold – 5 sec	Error rate > 5% = error -> defect Data loss > 0% = error -> defect Threshold > 10 sec = error -> defect
Buy car	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0 Response time: 100 ms Data loss: None	Session time – 15 min Threshold – 5 sec	Error rate > 0% = error -> defect Data loss > 0% = error -> defect Session time > 15 min = error -> defect Threshold > 5 sec = error -> defect
Sell car	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0 Response time: 100 ms Data loss: None	Session time – 5 min	Error rate > 0% = error -> defect Data loss > 0% = error -> defect Session time > 5 min = error -> defect
Edit car data	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.05 Response time: 600 ms Data loss: None	Session time – 5 min Threshold – 5 sec	Error rate > 5% = error -> defect Data loss > 0% = error -> defect Session time > 5 min = error -> defect Threshold > 5 sec = error -> defect
Edit profile	Connectivity, security, availability, usability, accessibility,	Error rate: 0.05 Response time: 600 ms Data loss: None	Session time – 5 min Threshold – 5 sec	Error rate > 2% = error -> defect Data loss > 0% = error -> defect

	performance, reliability			Session time > 10 min = error -> defect Threshold > 5 sec = error -> defect
Search	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.04 Response time: 500 ms Data loss: None	Threshold – 5 sec	Error rate > 4% = error -> defect Data loss > 0% = error -> defect Threshold > 5 sec = error -> defect
Search history	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.05 Response time: 600 ms Data loss: None	Session time – 10 min	Error rate > 5% = error -> defect Data loss > 0% = error -> defect Session time > 10 min = error -> defect
View car offer	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.02 Response time: 300 ms Data loss: None	Session time – 15 min	Error rate > 2% = error -> defect Data loss > 0% = error -> defect Session time > 15 min = error -> defect
Filters	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.02 Response time: 300 ms Data loss: None	Threshold – 5 sec	Error rate > 2% = error -> defect Data loss > 0% = error -> defect Threshold > 10 sec = error -> defect
Wish list	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.03 Response time: 400 ms Data loss: None	Threshold – 5 sec	Error rate > 3% = error -> defect Data loss > 0% = error -> defect Threshold > 10 sec = error -> defect
P2P chat	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0 Response time: 100 ms Data loss: None	Session time – 15 min Threshold – 3 sec	Error rate > 0% = error -> defect Data loss > 0% = error -> defect Session time > 15 min = error -> defect Threshold > 3 sec = error -> defect
Add dealership's car	Connectivity, security, availability, usability,	Error rate: 0.01 Response time: 200 ms Data loss: None	Session time – 5 min	Error rate > 1% = error -> defect Data loss > 0% = error -> defect

	accessibility, performance, reliability			Session time > 5 min = error -> defect
Send notification	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.04 Response time: 500 ms Data loss: None	Threshold – 3 sec	Error rate > 4% = error -> defect Data loss > 0% = error -> defect Threshold > 5 sec = error -> defect
View users	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.02 Response time: 300 ms Data loss: None	Session time – 15 min	Error rate > 2% = error -> defect Data loss > 0% = error -> defect Session time > 15 min = error -> defect
View metrics	Connectivity, security, availability, usability, accessibility, performance, reliability	Error rate: 0.01 Response time: 200 ms Data loss: None	Session time – 15 min	Error rate > 1% = error -> defect Data loss > 0% = error -> defect Session time > 15 min = error -> defect

Table of users mapped to functions and utilizing technical and behavioral measurements. Behavioral – activities performed by users; technical – systems respond on those activities.

Main measurement indicating system's state is Utilization factor = (No. attempts + No. queries)/Max load (above 1 is system overload; between 0-1 normal condition). Attempt – user's try of making action; query – successful attempt.

User.	Function.	Technical measurement.	Behavioral measurement.
Normal user	Sign up	Max load: 600 Utilization factor: $(300+100)/600 = 0.67$	No. attempts: 300 No. queries: 100
	Login	Max load: 1200 Utilization factor: $(600+400)/1200 = 0.83$	No. attempts: 600 No. queries: 400
	Buy car	Max load: 300 Utilization factor: $(100+50)/300 = 0.5$	No. attempts: 100 No. queries: 50
	Sell car	Max load: 200 Utilization factor: $(80+60)/200 = 0.7$	No. attempts: 80 No. queries: 60
	Edit car data	Max load: 400 Utilization factor: $(200+80)/400 = 0.7$	No. attempts: 200 No. queries: 80
	Edit profile	Max load: 600 Utilization factor: $(350+200)/600 = 0.91$	No. attempts: 350 No. queries: 200

	Search	Max load: 1000 Utilization factor: $(600+300)/1000 = 0.9$	No. attempts: 600 No. queries: 300
	Search history	Max load: 500 Utilization factor: $(200+100)/500 = 0.6$	No. attempts: 200 No. queries: 100
	View car offer	Max load: 2000 Utilization factor: $(900+600)/2000 = 0.75$	No. attempts: 900 No. queries: 600
	Filters	Max load: 700 Utilization factor: $(400+200)/700 = 0.85$	No. attempts: 400 No. queries: 200
	Wish list	Max load: 400 Utilization factor: $(150+100)/400 = 0.625$	No. attempts: 150 No. queries: 100
	P2P chat	Max load: 1100 Utilization factor: $(500+400)/1100 = 0.81$	No. attempts: 500 No. queries: 400
Admin	Add dealership's car	Max load: 100 Utilization factor: $(30+20)/100 = 0.5$	No. attempts: 30 No. queries: 20
	Send notification	Max load: 400 Utilization factor: $(200+120)/400 = 0.8$	No. attempts: 200 No. queries: 120
	View users	Max load: 600 Utilization factor: $(360+140)/600 = 0.83$	No. attempts: 360 No. queries: 140
	View metrics	Max load: 600 Utilization factor: $(400+150)/600 = 0.91$	No. attempts: 400 No. queries: 150

1.3. Requirements definition. **Software Design team.**

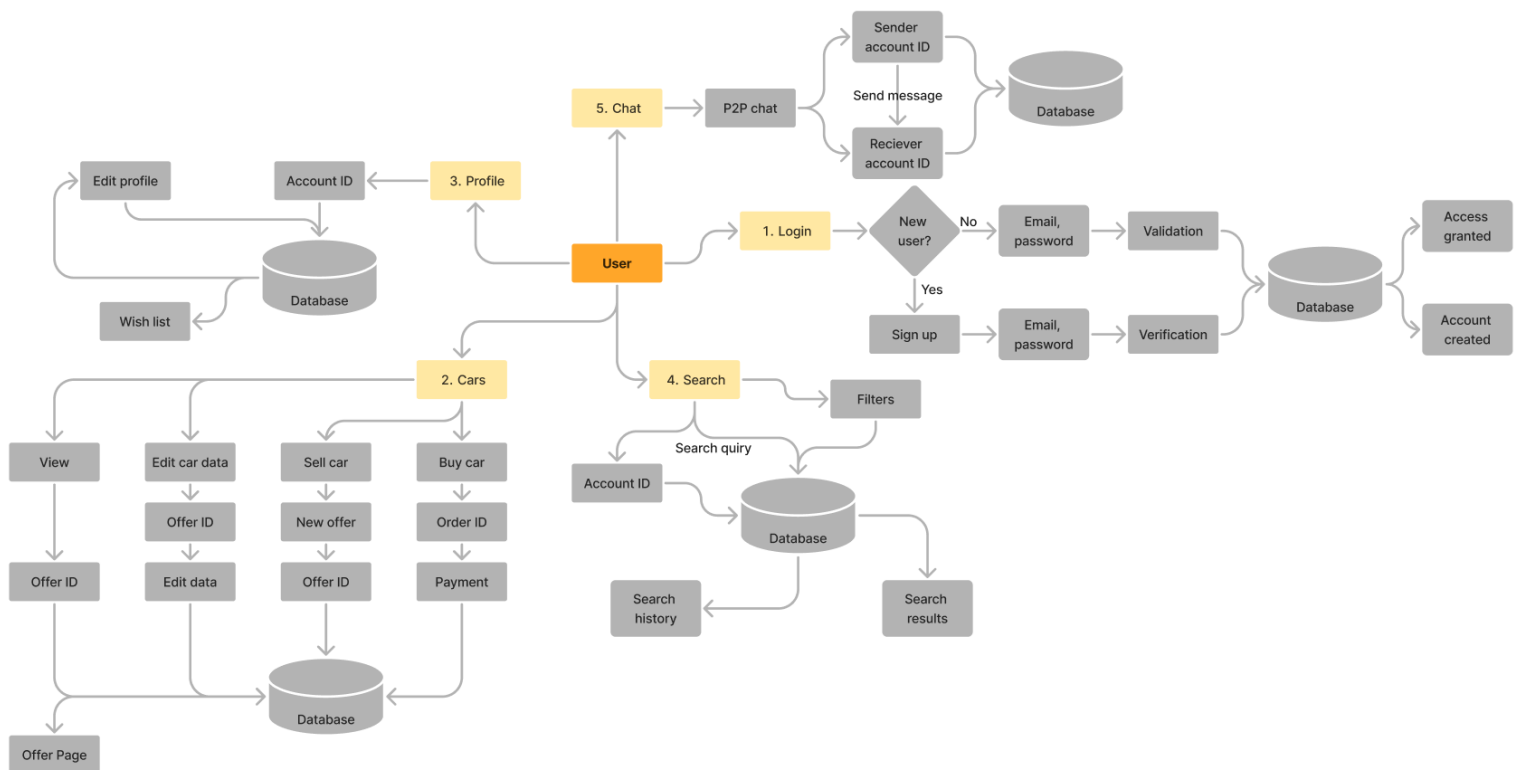
Input.

1.3.1. Workflow.

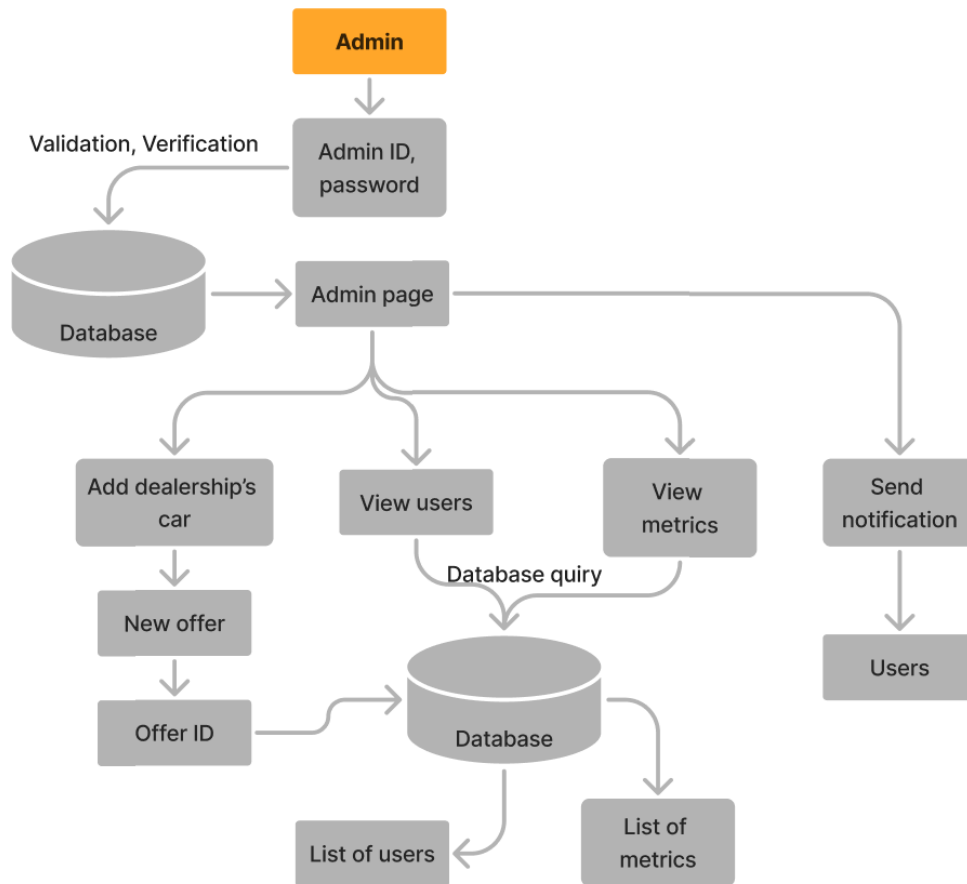
To proceed with Workflow diagram, sub scenarios are defined.

1. Login -> Login and Sign up.
2. Cars -> Buy car, Sell car, Edit car data, View car offer.
3. Profile -> Edit profile, Wish list.
4. Search -> Search, Search history, Filters.
5. Chat -> P2P chat.
6. Admin -> Add dealership's car, Send notification, View users, View metrics.

User Workflow diagram.



Admin Workflow diagram.



1.3.2. Mapping Classes to Objects with functions, data structures and metadata.

Class.	Objects.	Attributes.	Function.	Data structure.	Metadata.
User	User1, User2, User3.....	Email, username, password, Account ID, offer ID, order ID	Sign up	Table -> varchar	Email, username, password
			Login	Table -> varchar	Email, username, password
			Buy car	Table -> unique int	Account ID, offer ID, order ID
			Sell car	Table -> unique int	Account ID, new car instance
			Edit car data	Table -> unique int	Account ID, offer ID

			Edit profile	Table -> unique int	Account ID
			Search	Linear	Keyword
			Search history	Table -> unique int; Linear	Account ID, keywords
			View car offer	Table -> unique int	Account ID, offer ID
			Filters	Table -> varchar	Offer attributes
			Wish list	Table -> unique int	Account ID, offer ID
			P2P chat	Table -> unique int	Sender account ID, receiver account ID
Admin	Admin1, Admin2.	Admin ID, admin password	Add dealership's car	Table -> unique int, varchar	Admin ID, admin password, new car instance
			Send notification	Table -> unique int, varchar	Admin ID, admin password, account ID
			View users	Table -> unique int, varchar	Admin ID, admin password, account ID
			View metrics	Table -> unique int, varchar	Admin ID, admin password, log files
Car	BMW, Porsche, Audi, Mercedes.....	Type, year, engine, price	Functions are implemented on Car by User and Admin	Table -> varchar	Car ID, offer ID.

Output.

1.3.3. Functional and non-functional requirements.

<i>Function.</i>	<i>Metrics.</i>	<i>Functional requirements.</i>	<i>Non-functional requirements.</i>
Sign up	Error rate < 2% Data loss < 0% Session time < 10 min Threshold < 5 sec	User enters sign up page, writes Email and password, clicks button "Submit"	Accessibility to database, read from database, verification, validation, security, connection, visibility, error handling, redirect.
Login	Error rate < 5% Data loss < 0% Threshold < 5 sec	User enters login page, writes Email and password, clicks button "Login"	Accessibility to database, write to database, security, password encryption, connection, visibility.
Buy car	Error rate < 0% Data loss < 0% Session time < 15 min Threshold < 5 sec	When User finds car offer from which wants to buy a car, clicks "Buy car" button	Security, data protection, encryption, redirect, database record, payment systems integration, verification and validation, connection, visibility, payment API integration.
Sell car	Error rate < 0% Data loss < 0% Session time < 5 min	User has a car that wants to sell. Clicks "Sell car" button, fills in information about car and clicks "Submit"	Verification, database record, security, data protection, car listings update, connection, visibility, payment API integration.
Edit car data	Error rate < 5% Data loss < 0% Session time < 5 min Threshold < 5 sec	When viewing information about car that user is selling, user can press "Edit car data" button to change information about car. When editing data finished -> "Save changes" button	Performance, security (secure connection, data protection), mobile responsiveness, scalability, error handling, data privacy, integration with authentication systems, database accessibility, database read and write.
Edit profile	Error rate < 2% Data loss < 0% Session time < 10 min Threshold < 5 sec	When viewing profile, user can press "Edit profile" button to change personal information.	Performance, security (secure connection, data protection), mobile responsiveness, scalability, error handling, data privacy, integration with authentication

		When editing data finished -> "Save changes" button	systems, database accessibility, database read and write.
Search	Error rate < 4% Data loss < 0% Threshold < 5 sec	In search bar, user enters some search query and presses "Search" button	Security (secure connection, data protection), redirect, database, mobile adaptation, search and filter efficiency (using keywords), error handling.
Search history	Error rate < 5% Data loss < 0% Session time < 10 min	To view search history, user presses "Search history" button -> redirected to search history page	Accessibility to database, read from database, security, connection, visibility, verification.
View car offer	Error rate < 2% Data loss < 0% Session time < 15 min	On the main page or search results page user find car offer -> clicks on it -> redirected to car offer page	Database accessibility, security, visibility, verification, connection, availability, encryption,.
Filters	Error rate < 2% Data loss < 0% Threshold < 5 sec	When searching, user clicks "Filters" button, chooses some filters and press "Apply filters"	Accessibility to database, write to database, security, password encryption, connection, visibility.
Wish list	Error rate < 3% Data loss < 0% Threshold < 5 sec	When viewing car offer, user clicks Wish list icon -> car offer added to user's Wish list	Accessibility to database, write to database, security, connection, visibility, verification.
P2P chat	Error rate < 0% Data loss < 0% Session time < 15 min Threshold < 3 sec	User clicks Chat icon to enter P2P chat page. There chooses another user -> chats with this user	Security (secure connection, data encryption), user-friendly interface, mobile responsiveness, scalability, error handling, chat history storage, data privacy, network connection, database.
Add dealership's car	Error rate < 1% Data loss < 0% Session time < 5 min	Admin wants to add new car offer from dealership, clicks "Add dealership's car" button, fills in car information -> "Submit" button	Performance, security (secure connection, data protection), mobile responsiveness, scalability, error handling, data privacy, integration with authentication systems, database accessibility, database read and write.
Send notification	Error rate < 4% Data loss < 0%	Admin needs to send notification to users ->	Security, data protection, mobile responsiveness,

	Threshold < 3 sec	“Send notification” -> writes text of notification, chooses user to receive notification -> “Submit” button	error handling, real-time notification delivery, storage in database, privacy and consent management, cross-platform compatibility
View users	Error rate < 2% Data loss < 0% Session time < 15 min	Admin needs to see list of users in database. Before that admin enters Admin ID and password to access admin account. Then “View users” -> views users list in database	Performance, security (secure connection, data protection), mobile responsiveness, scalability, error handling, data privacy, integration with authentication systems, database accessibility, database read and write.
View metrics	Error rate < 1% Data loss < 0% Session time < 15 min	Admin needs to see metrics in database. Before that admin enters Admin ID and password to access admin account. Then “View metrics” -> views list of metrics in database	Performance, security (secure connection, data protection), mobile responsiveness, scalability, error handling, data privacy, integration with authentication systems, database accessibility, database read and write.

2. Design. **Software Design team.**

2.1. Workflow, class object.

In requirement definition phase objects were defined: Email, username, password, Account ID, offer ID, order ID, Admin ID, admin password.

Using these objects, we defined main classes: normal user and admin.

From objects and classes, we defined functions and metadata of objects.

Mapping classes and objects, helped to achieve low coupling and high cohesion.

Main scenario is the main page of application. From main scenario, we identified sub scenarios:

Login -> Login and Sign up.

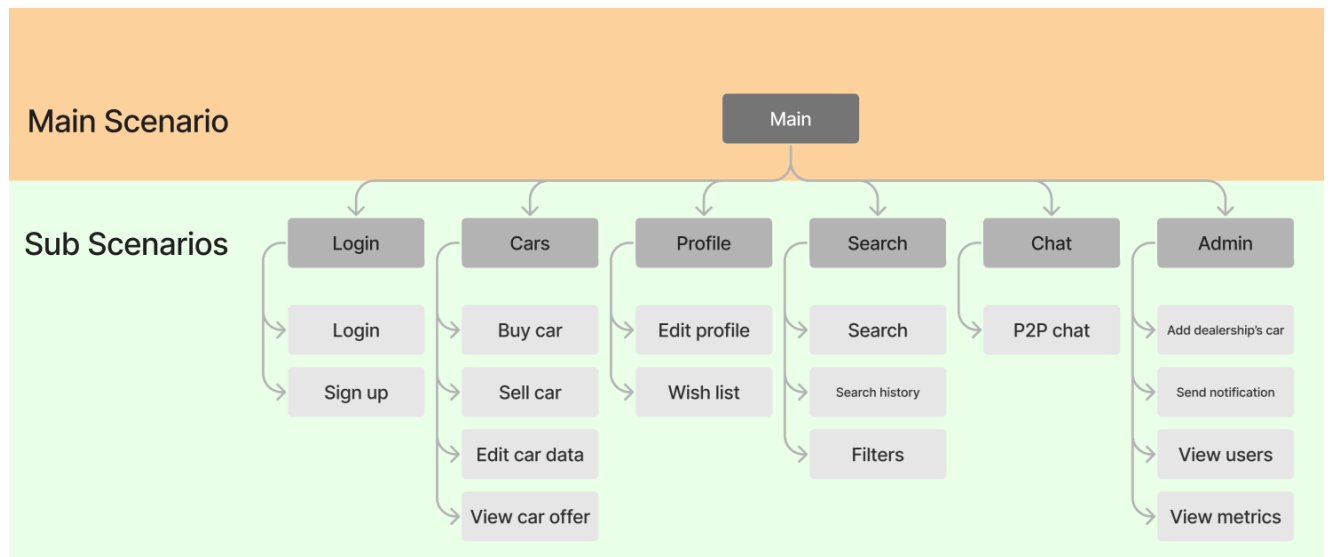
Cars -> Buy car, Sell car, Edit car data, View car offer.

Profile -> Edit profile, Wish list.

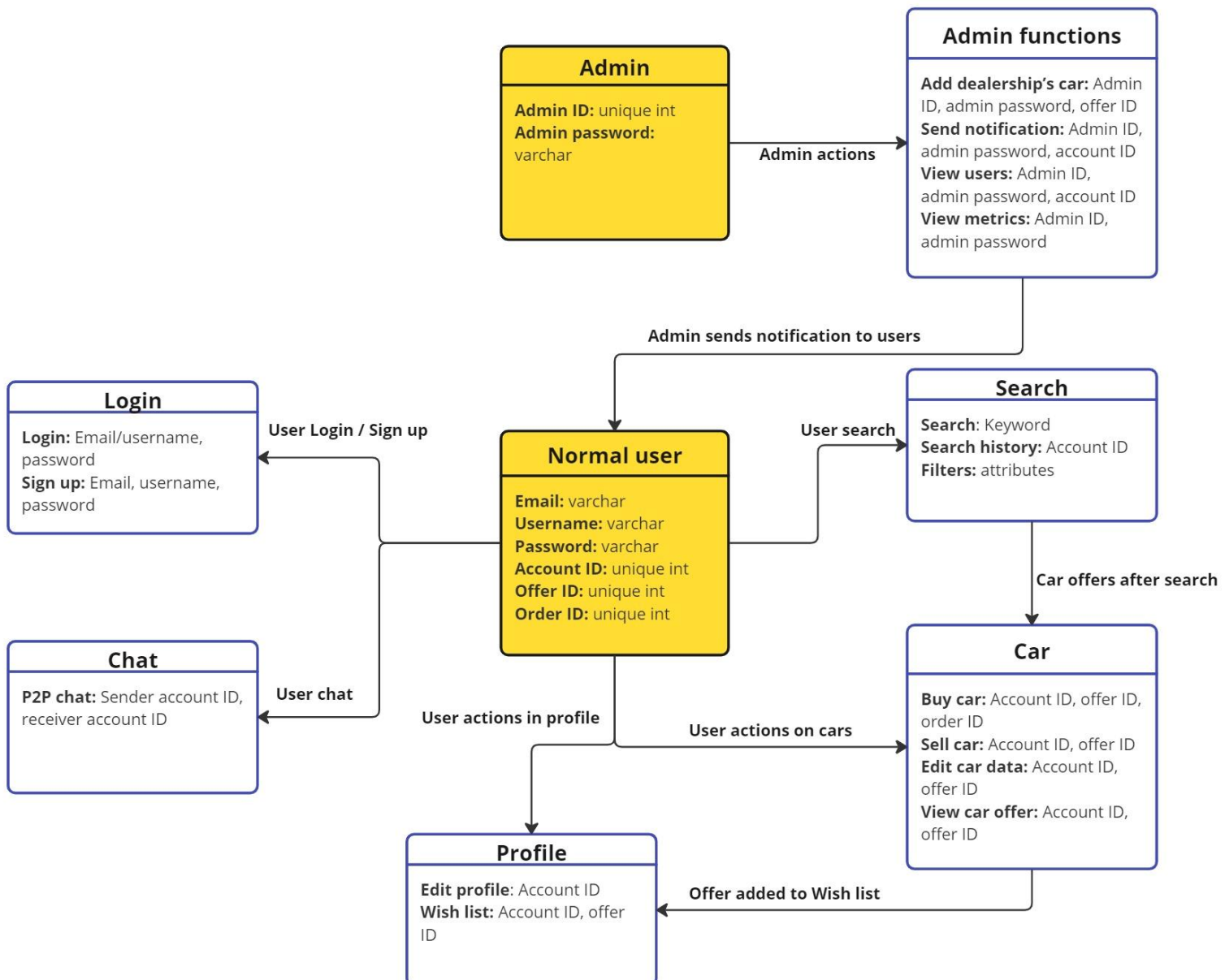
Search -> Search, Search history, Filters.

Chat -> P2P chat.

Admin -> Add dealership's car, Send notification, View users, View metrics.



Workflow UML diagram.



2.2. Development. **Software Development team.**

2.2.a. Programming.

Choose programming language(s) for the development. For example, Python is chosen as main programming language.

2.2.b. Coding.

Coding of the application following common code writing rules.

2.2.c. Tools.

For example, GitHub, Visual Studio Code, Docker.

2.2.d. Libraries.

For example, Flask, Django, BeeWare.

2.3. Prototype.

Prototype of software product is created.

3. Validation.

3.1. Specification.

Revise Specification level 1 and 2 again.

3.2. Testing. **Software Testing team.**

Test plan is created. Scope: to evaluate software, its functionality and check if it all works as it was planned to work. Test plan is implemented through series of tests – unit test, subsystem test and integration test, which all combined give the alpha test. To make the test acquired measurement units are compared to the expected values of measurements.

3.2.a. Unit Test – white box (WB).

Testing functions or units of software product separately.

<i>Function.</i>	<i>Expected.</i>	<i>Result.</i>
Sign up	Error rate < 2% Data loss < 0% Session time < 10 min Threshold < 5 sec	Pass
Login	Error rate < 5% Data loss < 0% Threshold < 5 sec	Pass
Buy car	Error rate < 0% Data loss < 0% Session time < 15 min Threshold < 5 sec	Pass
Sell car	Error rate < 0% Data loss < 0% Session time < 5 min	Pass

Edit car data	Error rate < 5% Data loss < 0% Session time < 5 min Threshold < 5 sec	Pass
Edit profile	Error rate < 2% Data loss < 0% Session time < 10 min Threshold < 5 sec	Pass
Search	Error rate < 4% Data loss < 0% Threshold < 5 sec	Pass
Search history	Error rate < 5% Data loss < 0% Session time < 10 min	Pass
View car offer	Error rate < 2% Data loss < 0% Session time < 15 min	Pass
Filters	Error rate < 2% Data loss < 0% Threshold < 5 sec	Pass
Wish list	Error rate < 3% Data loss < 0% Threshold < 5 sec	Pass
P2P chat	Error rate < 0% Data loss < 0% Session time < 15 min Threshold < 3 sec	Pass
Add dealership's car	Error rate < 1% Data loss < 0% Session time < 5 min	Pass
Send notification	Error rate < 4% Data loss < 0% Threshold < 3 sec	Pass
View users	Error rate < 2% Data loss < 0% Session time < 15 min	Pass
View metrics	Error rate < 1% Data loss < 0% Session time < 15 min	Pass

3.2.b. Subsystem test – white box (WB).

Testing the function as part of the subsystem, their connection between each other.

<i>Subsystem.</i>	<i>Functions.</i>	<i>Connection between functions.</i>	<i>Result.</i>
Login	Login, Sign up	Established, stable	Pass
Cars	Buy car, Sell car, Edit car data, View car offer	Established, stable	Pass
Profile	Edit profile, Wish list	Established, stable	Pass

Search	Search, Search history, Filters	Established, stable	Pass
Chat	P2P chat	Established, stable	Pass
Admin	Add dealership's car, Send notification, View users, View metrics	Established, stable	Pass

3.2.c. Integration test – black box (BB).

In the integration test all subsystems are put together and tested as a whole, one single system. Connection and work correctness between subsystems is tested.

In the end, all these 3 tests (unit, subsystem and integration) put together and represent the alpha test of the software product.

WB + BB = alpha test.

Output of Testing = Exit report of all tests performed.