# Seminar Report:

### Introductory information:

| **Parameter:** | **Description:** |
| --- | --- |
| **Project:** | Development of an online system for order delivery |
| **Date of the event:** | 06.04.2025 |
| **Venue:** | https://salutejazz.ru/calls/h5o5qh?psw=OAEJBEoUExUbEQ4SFQAEBhQbEw |
| **Leading:** | combolyn |
| **Clerk:** | greenhih, melodyma |
| **Seminar format:** | Theoretical and debatable |
| **Objectives of the seminar:** | 1. Business requirements;  2. Roles of the system, Needs of roles, their problems as is;  3. Solutions to problems considered in the brainstorming to be;  4. Boundaries of the system in the form of functions performed by roles in the system. |

To conduct the seminar more effectively, the presenter compiled a list of key questions and theoretical information for each section, and also recorded all changes made, indicating their authors.

### Business requirements:

| **Business requirement:** | **Business rule:** |
| --- | --- |
| Provide customers with the ability to track the status of their delivery in a format convenient for the customer. | * Order tracking is only possible with a valid tracking number or customer authorization. * Status change history is saved for at least 30 days after delivery is completed. |
| Ensure high accuracy of delivery time forecast | * The delivery time forecast must be recalculated every time the delivery status or route changes. * If the forecast cannot be met, the delivery is transferred to the "Delay" status. * If the forecast change exceeds 30 minutes, the customer must be notified. |
| Flexible delivery to meet customer needs | * The customer can select a delivery interval from the available time windows when placing an order. * Intervals depend on the logistics capabilities of the region and the number of available couriers. * Changing the selected interval is possible no later than 2 hours before the start of the delivery window. * Delivery is possible only during working hours. |
| Transparency and accuracy of shipping cost calculations | * The shipping cost should be calculated automatically when placing an order. * If the weight, dimensions or shipping address change, the cost is recalculated automatically. * The minimum shipping cost is X (set by company policy), regardless of the conditions. |
| Possibility to change delivery parameters after placing an order | * Changing the delivery address is possible no later than 2 hours before the order is shipped from the supplier * All changes must be confirmed by the user via code/confirmation. * The change history must be saved and be available for viewing by support staff. |
| Increase customer awareness of the delivery process | * Notifications to the client must be sent when the following events occur:  1. Confirmation of order; 2. Handover to courier; 3. Delivery delay; 4. Arrival at destination; 5. Completion of delivery;  * The customer can select the preferred notification channel (email, SMS, push in the app) when placing an order. * If the customer does not confirm receipt of the notification, a re-send must be performed. |
| Increase motivation of couriers through income transparency | * The courier must have access to income details for each order. * Income is displayed only after delivery completion is confirmed and is available in real time. * The bonus is awarded for completing more than N deliveries per day/week (in accordance with company policy). |

### Corrections and answers to questions on the “Business Requirements” block:

| **Business requirement:** | **Business rule:** | **Initiator:** | **Note:** |
| --- | --- | --- | --- |
| Simplify interaction with suppliers | Suppliers receive a single electronic document with all orders once a day through an integrated system, without the need for manual confirmation on both sides. | greenhih | New requirement and rule added. |
| Automated control of couriers and automatic redistribution | If the courier does not confirm receipt of a new task within 3 minutes, the system automatically redistributes the order to another available courier. | melodyma | New requirement and rule added. |

| **Question:** | **Answer:** |
| --- | --- |
| **Option for technical implementation of the system** (*application / web version*)**?** | **Application -** for the client, courier;  **Web version -** dispatcher, administrator, supplier and all other roles and stakeholders. |
| **Accuracy of the forecast** (*time intervals sent to the client***) about delivery?** | **In normal mode** - 90% accuracy of time forecasts.  **During peak loads** - at least 70%. |
| **What will the loyalty and motivation system for couriers look like?** | The motivation system for couriers will increase their coefficient based on the demand for orders, weather conditions and personal performance in fulfilling orders |
| **Will the project's scaling in functionality be considered in the future, in the long term?** | Scaling issues will be considered after the effectiveness of the business model is confirmed. |
| **What notification channels will be used for clients?** *Should channel selection or multi-channel be provided?* | We use external APIs for (*SMS, mailings, calls*), in addition to internal push notifications.  We provide clients with custom notification settings / multi-channel by default. |
| **What delivery data should the customer see in real time?** | 1. Order status; 2. Composition; 3. Courier and contact information; 4. Time; 5. Courier location; |
| **How will the final shipping cost be calculated for the customer?** *What variables are taken into account: weight, volume, distance, urgency?* | **It is necessary to develop a formula for calculating the final cost, which should include:**  **1. Courier and his criteria;**   1. **Order cost for the supplier;** 2. **Order processing cost;** 3. **Company marginality (our %);** |
| **What metrics and reports will a business need? *For example: % of delays, average delivery time, courier efficiency.*** | 1. Number of completed orders;  2. Number of orders completed on time;  3. Average delivery time;  4. Number of couriers on the line;  5. Number of suppliers;  6. Number of repeat orders. |
| **What delivery scenarios should be considered: pick-up, delivery to a pick-up point, delivery by courier?** | Courier delivery only. |

### Roles of the system and their needs As is:

| **Role:** | **Requirements / Responsibilities:** | **Access:** | **The role problem As is: (at the time of system development)** |
| --- | --- | --- | --- |
| 1. **Courier** | 1. View available orders; 2. Book an order; 3. Mark on order receipt; 4. Mark on order delivery; 5. View accrued payment; | Limited access | 1. It is difficult to find a stable job - you need to constantly look for orders on different platforms . 2. You have to negotiate with each client separately. 3. There is no navigation and tracking - it is difficult to plan a route and estimate delivery time. 4. Income is irregular, payment is often delayed. 5. There are no transparent working conditions, often there are no contracts or guarantees. |
| 1. **Administrator** | 1. Registration of couriers; 2. Access rights within the system; 3. Monitoring; 4. Blocking users; 5. Handling problem situations / Reports on system operation; | **Full access** | 1. Lack of a centralized database — all information is stored in different places: paper records, Excel, instant messengers. 2. . Manual processing of applications — you have to manually check, sort and send orders. 3. Errors in data entry — there is no automatic verification, which leads to errors in information about clients and couriers. 4. Lack of analytics — there are no tools for tracking the efficiency of the system, tracking errors or problems. 5. Ineffective resource management — problems with managing couriers, orders, since there are no tools for automatic coordination. |
| 1. **Client** | 1. Placing an order; 2. Delivery status; 3. Selecting delivery time; 4. Cancelling an order; 5. Payment; | Limited access | 1. Ordering food or delivery means phone calls, messages, and a lot of manual actions. 2. No certainty that the order will arrive at all, and when exactly. 3. Difficult to pay: only cash or transfer by number. 4. . No support if something goes wrong 5. It is impossible to track who will arrive and in what condition the goods are. |
| 1. **Dispatcher** | 1. Order information; 2. Assignment to couriers; 3. Order control; 4. Receiving calls / Reassignment; 5. Cancellation of orders; | Limited access | 1. Manual coordination of orders - all order data comes via calls and messages, you have to manually assign couriers.  2. No way to track order statuses - you have to check with the courier or customer each time where the order is.  3. Problems with multitasking - it is difficult to control several orders at the same time, there are errors in distribution.  4. Inefficient routing - no tools for route optimization, you have to calculate manually, which leads to delays.  5. Difficulties in communicating with couriers - all interaction occurs via phone and messengers, misunderstandings arise, especially in case of changes in orders. |
| 1. **Supplier** | 1. Submits orders; 2. Delivery status; 3. Report on work done; 4. Calculation; | Limited access | 1. No automation: orders are accepted by phone, paper, messengers. 2. High risk of errors, losses, duplicate orders. 3. Impossible to forecast demand and plan logistics. 4. You have to independently search for sales points or clients. |
| 1. **Student company**   **(development team)** | 1. System development; 2. Analysis of current problems and adjustment of functionality; 3. Control over cash flow; | **Full access** | 1. No live product - nothing to test or show.  2. It is difficult to collect real requirements from users.  3. The team has no experience of launching projects "from scratch".  4. It is unclear who the target audience is and what their real pain points are.   1. It is impossible to start development without understanding the processes. |
| 1. **Accountant** | 1. Data on completed orders; 2. Calculation of courier income; 3. Interaction with the supplier; 4. Reporting to the state 5. Financial reports to the student team; | Limited access | 1. All payments are made manually - tables, transfers, errors.  2. Settlements with suppliers and couriers - via messengers, without accounting.  3. No automatic report generation.  4. It is difficult to keep track of receipts and payments.  5. Tax and financial reporting. |
| 1. **IT staff**   **(development team)** | 1. System status monitoring; 2. Updates; 3. Update documentation; 4. Feedback / Bug and vulnerability fixes; | **Full access** | 1. Nothing to administer — no system. 2. Everything works in disparate services — Google Sheets, messengers, phones. 3. It is impossible to build redundancy, security or monitoring. 4. When a technical need arises, everything has to be created "from scratch". 5. No documentation, architecture, standards — complete improvisation. |

### Corrections and answers to questions on the block “System roles and their problems As is”:

| **Role:** | **Correction:** | **Initiator:** | **Note:** |
| --- | --- | --- | --- |
| **Operator** | Role added | greenhih | Forgot to mention this when originally drafted. |
| **Supplier** | Problem:  It is necessary to have your own staff of couriers | melodyma | The problem was not included in the original version. |

| **Question:** | **Answer:** |
| --- | --- |
| **Are all roles listed in this table?** | Yes |
| **Are the party's issues relevant or should they be supplemented or adjusted?** | No, it's not worth it. |
| **Are changes in access required for certain roles?** | No, it's not worth it. |
| **Are all key system requirements and party responsibilities listed in the table?** | Yes, they are listed. |

3. Solutions to the problems discussed during the brainstorming session To be:

| **Role:** | **Possible problems:** | **Solution to the problem To be:** |
| --- | --- | --- |
| 1. **Courier** | 1. No filters or sorting (e.g. by distance, delivery type, urgency). 2. Orders may be updated with a delay, causing couriers to lose current orders. 3. Conflicts are possible when booking the same order simultaneously with different couriers. 4. No mechanism for automatic confirmation or short-term reservation (an order may "disappear" if not confirmed quickly). 5. The ability to book multiple orders at the same time, which may lead to cancellations or delays. 6. There is not always a clear indication of the delivery conditions (weight, dimensions, restrictions). | 1. Developing the order card. Specifying the necessary parameters.  2. Developing the order card. Specifying the necessary parameters.  3. System limitation on the number of orders taken.  4. Adding the necessary algorithms for auto-confirmation.  5. Fall into a separate internal interface "Booking", exit from it "Active orders", which are in the interface "Order". |
| 1. **Administrator** | 1. High administrator load, lots of manual work.  2. Data entry errors (e.g. incorrectly entered accounts).  3. No automatic notification of critical events (e.g. system crashes, mass errors).  4. No centralized access policy management (e.g. automatic deletion of inactive users). | 1. Solved by internal integration and CRM setup.  2. Solved by automation (upon registration, rights are issued automatically).  3. Solved by adding this mechanism.  4. Solved by adding functionality to the system.  Now the administrator can manage access directly without delays. Also, inactive users are automatically deleted from the system after a set period of time. |
| 1. **Client** | .1. Lack of transparency in real-time order tracking.  2. No clear communication with the courier (for example, the client cannot call him directly).  3. Long wait for support in case of delivery issues.  4. Limited choice of payment methods (for example, no option to pay with cryptocurrency or via PayPal).  5. No loyalty program for regular customers | 1. Added functionality with available suppliers. An internal interface with a number of levels for the user has been developed to minimize his interaction with third-party resources where the system is an intermediary.  2. But without canceling integration with third-party services.  3. Also, the interface of this side provides for all interactions with support and the courier.  4. The ability to make a call through the application. |
| 1. **Dispatcher** | 1. Manually checks the status of orders, no automatic notifications about problems.  2. Possible delays in the transfer of information between the dispatcher and the courier.  3. Errors in assigning couriers (for example, if you assign one courier to two orders at the same time).  4. No integration with maps to calculate the optimal delivery route.  5. Calls are not made through the system, but through external communication. | 1. Setting up CRM. All interactions take place within the system.  2. Adding an algorithm that automatically adds and removes the responsible person in case of reassignment.  3. Algorithms and restrictions in the system itself in order to avoid problematic situations (dual purpose).  4. Setting up integration with maps, an algorithm for calculating the route (own or integrated).  5. Automatic notifications in case of order downtime. |
| 1. **Supplier** | 1. There may be errors in transmitting information about orders (for example, lost or duplicate orders).  2. There is no real tracking of the order during delivery.  3. There is no convenient analytics for completed and cancelled orders.  4. Couriers may not arrive at the appointed time, because there is no clear synchronization with the system. | 1. Develop an API or web platform for automatic order acceptance to eliminate manual work.  2. Create a personal account or a bot in a messenger where clients can track the status of orders.  3. Automate the generation of reports and sending them on a schedule or in real time via email or a notification system. |
| 1. **Student company**   **(development team)** | 1. Development is slow due to lack of resources (time, specialists, knowledge).  2. There is no automated collection of metrics on the system's operation.  3. It is difficult to maintain a balance between development, user support and business processes.  4. Limited budget, which complicates the development of the project. | 1. Hire an analyst who will help analyze the product needs, compensate for the lack of knowledge of specialists in the areas where it is observed.  2. Solved by adding a mechanism for collecting information on the work of the project.  3. Solved by adding a mechanism for financial transactions without the need to get into other external systems. The solution is in the maximum automation of these needs. |
| 1. **Accountant** | 1. Possible calculation errors due to manual data entry.  2. Delays in payments to couriers and suppliers due to untimely information updates.  3. Lack of automated reporting and integration with banking systems.  4. Complicated tax planning due to insufficient transparency of payment transactions. | 1. The algorithm reads delivery data in real time. It calculates according to the formula and uploads it to the CRM. It requires confirmation.  2. The transfer of funds occurs automatically. |
| 1. **IT staff**   **(development team)** | 1. Long response time to technical failures.  2. No automatic backup and version rollback systems.  3. No test environment, which is why updates can break the system.  4. High load on a small team due to constant fixes and improvements. | 1. Develop a planned work schedule and follow it, observing deadlines.  2. Also, in order to avoid breakdowns, thoroughly test the product before release and use gitlab version control systems.  3. It is solved by adding a special algorithm to the system that, when a new patch is released, will interact with the user in an interactive format, depending on who the interested party is, and inform about new features. |

### Corrections and answers to questions on the block “Solving problems considered during the brainstorming session To be”:

| **Role:** | **Problem:** | **Solution:** | **Initiator:** |
| --- | --- | --- | --- |
| **Operator** |  | Removing from the system as a role. | greenhih |
| **IT Staff** | **As is:**  High workload for a small team due to constant fixes and improvements. | **To be:**  A task prioritization system has been introduced (for example, Kanban + SLA), external contractors/volunteers have been connected for some tasks.  A tech debt sprint has also been introduced - a week each month dedicated only to refactoring and improvements. | melodyma |

| **Question:** | **Answer:** |
| --- | --- |
| **1. Are all the problem solutions filled in correctly? Have we missed anything?** | **The key issues have been resolved and agreed upon.** |
| **2. What result do we want to achieve after implementing the changes?** | **An optimized and stable functioning system.** |
| **3. What functions should the new system perform?** | **Optimization of business processes.** |
| **4. What processes should be automated or optimized?** | **Everything, as far as possible.** |
| **5. How will user interaction with the system change?** | **The process of delivering goods to customers will be accelerated, a lot of unnecessary problems with redirection and paperwork for users of the system on the part of our organization will go away.** |
| **6. What new roles or actions will appear in the system?** | **Unnecessary roles will be removed, Operator, since there is no need for him and his functions can be fully covered by the system.** |
| **7. What interfaces will be required (user, internal, external)?** | **Mobile app:**  **Custom for client and courier.**  **Web version:**  **all other interfaces (including courier and client)** |
| **8. What metrics will be used to evaluate the effectiveness of the To Be model?** | **The same as when listing business requirements, we do not add any new ones.** |
| **9. What risks may arise during the transition to To Be?** | **1. Lack of resources for implementation;**  **2. Non-working economic model;**  **3. Competition;**  **4. Backwardness of technological implementation of the system;** |
| **10. What resources (people, technology, budget) will be required to implement To Be?** | **1. Consult a lawyer;**  **2. Consult an organization to check the system for security vulnerabilities;**  **3. Contact an organization to test the application to find bugs.** |

### 4. System boundaries:

| **Role** | **Input flow**  ***(across the border):*** | **Output flow**  ***(across the border):*** | **System boundary**  ***(within the system)*:** | **Outside the system**  ***(not within the system*):** |
| --- | --- | --- | --- | --- |
| Client | 1. Makes an order;  2. Confirmation of order delivery;  3. Online payment for the order;  4. Report on possible problems related to the system;  5. Details for automatic write-off;  6. Request in case of problems with delivery;  7. Request to cancel the order; | 1. Order status data;  2. Order cost data;  3. Order and courier location data;  4. Data on updated system documentation;  5. Payment request;  6. Request for confirmation of order delivery; | Interacting with the application via the interface | Transfer of funds, interaction with the bank |
| Administrator | 1. Filling in information about the courier;  2. Granting access rights;  3. Blocking the user;  4. Reports on the operation of the system;  5. Report on possible problems associated with the system;  6. Resolving a conflict situation; | 1. General information about the system operation;  2. Information about updated documentation for the system;  3. Information about system users;  4. Incoming requests in case of problems; | Management via admin interface | Monitoring external systems, handling external incidents |
| Dispatcher | 1. Assigns a responsible person (courier) to the order;  2. Reassigns the responsible person;  3. Records cancelled orders and generates a report on them;  4. Reports on possible problems associated with the system;  5. Reassigns to the administrator / solves the problem;  6. Records the cancellation of the order. | 1. Order information;  2. Incoming request from the client;  3. Data on updated documentation for the system; | Interaction with the planning system | - |
| Courier | 1. Order reservation;  2. Mark of order receipt from the supplier;  3. Mark of order delivery;  4. Courier location while en route;  5. Report of possible problems related to the system;  6. Details;  7. Balance request; | 1. Booking confirmation;  2. Available orders details;  3. Balance details;  4. Updated system documentation details;  5. General order details, customer number, address, etc. | Receiving tasks via mobile application | Contact with customers during delivery |
| Developers | 1. Ready-made instructions for reorganization (of something in the system);  2. Requests for work done;  3. Information on current rates and commissions; | 1. Reports on system operation/finances/feedback/general statistics; 2. Data on updated documentation for the system; | Working with reports and improvements | Interacting with external services for improvements |
| IT staff | 1. Update documentation;  2. System updates;  3. Response to request; | 1. Information about existing errors and vulnerabilities from system users;  2. Feedback on the update;  3. Request for work done; | Interaction with infrastructure and systems  Interaction with the order adding system | Working with external hosting, third-party services |
| Supplier | 1. Order information;  2. New order;  3. Confirmation of order delivery to the courier;  4. Payment details; | 1. Information about the order status/completion;  2. Information about the crediting of funds to the balance;  3. Information about updated documentation for the system; |  | External contracts, logistics and shipments |
| Accountant | 1. Calculation of couriers;  2. Calculation of suppliers;  3. Report on request of student team / regulator; | 1. Data on completed orders;  2. Request for reporting;  3. Data on updated system documentation;  4. Information on current rates and commissions;  5. Information on current details of couriers/suppliers; | Working with the accounting and reporting system | Transfers of funds, tax reporting, interaction with banks |

### Corrections and answers to questions on the “System Boundaries” block:

| **Role:** | **Correction:** | **Initiator:** | **Note:** |
| --- | --- | --- | --- |
| **Client** | **Input flow (add-on):**  Refund status request  **Output flow (add-on):**  Refund/denial information | melodyma |  |
| **Accountant** | **Input flow (add-on):**  Request for financial discrepancies.  **Output flow (add-on):**  Financial reporting for management | greenhih |  |

| **Question:** | **Answer:** |
| --- | --- |
| **What functions should the system perform, and which should it not?** | Everything related to the delivery of goods to the client, from order placement to receipt. |
| **What processes are inside the system, and which occur in the external environment?** | All those related to delivery and staff work, all those not related to the processes above. |
| **Who interacts with the system, but is not part of it (e.g. external users, suppliers)?** | A. Media;  B. Competitors;  C. Regulators;  D. Banks;  Outstaffing company that provides couriers from outside; |
| **What external systems or services are involved in the interaction?** | 1. Payment gateways;   2. CRM;  3. Mail APIs. |
| **Where does the area of responsibility of our system end?** | **Examples:**  1. Speed of return of funds;  2. Consumer characteristics of the product. |
| **What actions should the user perform independently outside the system?** | **Client:**  Pick up delivery from courier.  **Courier:**  Pick up order from supplier.  **Client/courier:**  Confirm details. |
| **What resources or services do we use, but do not control (e.g. external databases, courier services)?** | 1. Outstaffing company; 2. Weather services; 3. Payment gateways; 4. Navigation services. |
| **What external systems do users interact with through the system interfaces?** | **Client**: navigation, bank, API.  **Courier**: navigation, bank.  **Accountant**: bank, regulators and their systems, CRM.  **Supplier**: bank, navigation.  **Dispatcher** / **Administrator**: CRM.  **Student team**: CRM. |
| **Are there processes that should not be automated or managed by the system?** | 1. Recruitment of new couriers;  2. Initial debiting of funds from the client's card for confirmation;  3. Connecting new suppliers. |

### Conclusion:

By the end of the workshop, all participants had reached a common understanding of the project, which ensured consistency of views and understanding of its goals and objectives. During the discussion, inaccuracies were identified and eliminated, which improved the concept of the project.   
  
Business requirements were also clearly formulated and defined, and key features of the project were reviewed before the start of development, which allowed eliminating possible misunderstandings at the preparatory stage.