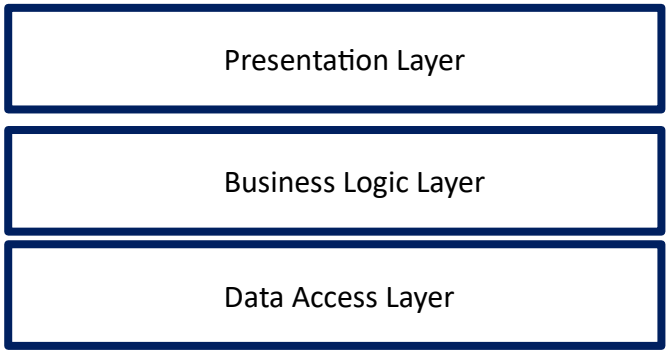
Design structure

The design structure of the ReserveWell Application is organized in a layered architecture, ensuring modularity, scalability, and maintainability. Each layer serves a specific purpose and collaborates seamlessly to achieve the overall goal of efficiently managing restaurant reservations.



Layered Architecture

Subsystems

The application has no subsystems.

Patterns

[Pattern 1-Make Reservation]

Overview

The Reservation pattern is tailored to streamline the reservation process within the ReserveWell Application, specifically targeting the "Make Reservation" use case. With a focus on enhancing efficiency and user convenience, this pattern aims to provide a dynamic and adaptable framework for managing reservations effectively.

Intent: The primary intent of the Reservation Workflow pattern in the context of the "Make Reservation" use case is to create a seamless, user-friendly experience for diners while ensuring optimal utilization of restaurant resources. This pattern intends to automate and optimize the steps involved in making a reservation, offering a responsive system that caters to varying diner preferences and restaurant capacities.

Motivation: Traditional reservation systems often involve manual steps and may lack adaptability to real-time changes in diner preferences and restaurant availability. The pattern is motivated by the need to address these challenges by introducing automation and dynamic decision-making. By doing so, the

pattern aims to improve the overall reservation process, benefiting both diners and restaurant managers.

Applicability: The pattern is particularly applicable to the "Make Reservation" use case within the ReserveWell Application. It caters to the interests of various stakeholders, including diners, restaurant managers, system administrators, the Payment Authorization Service, and the Development Team. The pattern is relevant in scenarios where:

- Diners seek an efficient and convenient reservation process.
- Restaurant managers aim to optimize table utilization and ensure a smooth dining experience.
- System administrators need a stable and scalable reservation system.
- The Payment Authorization Service requires seamless integration with the bank's system.

By adopting the pattern, the ReserveWell Application aims to provide a responsive, user-centric reservation experience while ensuring system stability, scalability, and adherence to best practices.

Structure

Diner:

- Role: Initiates the reservation process by interacting with the application's user interface.
- Relevant Data: Provides information such as the desired restaurant, the number of guests, preferred date and time, name, surname, email.
- Behavior: Triggers the reservation confirmation process by confirming the provided information.

Restaurant Manager:

- Role: Manages reservations effectively and ensures a seamless dining experience.
- Relevant Data: Accesses and updates reservation information, including table availability, to optimize the dining experience.
- Behavior: Utilizes tools and functionalities to oversee the reservation process and ensure a smooth flow of restaurant operations.

Waitstaff:

- Role: Provides efficient and high-quality service to customers, including arranging tables' physical availability based on reservation updates.
- Relevant Data: Receives real-time updates on reservations and allocates tables accordingly to enhance customer service.
- Behavior: Utilizes tools within the application to organize seating arrangements and deliver a positive dining experience.

Reservation System

- Role: Central component responsible for managing reservations.
- Relevant Data: Stores and processes diner-provided information, generates a unique reservation ID, and updates the system with new reservations.
- Behavior: Validates and records diner reservations, ensuring the availability of the reserved table or dining area during the specified date and time.

User Interface Modules:

- Role: Facilitates interaction between the diner and the reservation system.
- Relevant Data: Displays restaurant options, reservation details, and confirmation messages.
- Behavior: Presents a user-friendly interface, collects diner inputs, and communicates reservation status.

Behavior

Diner Initiates Reservation:

The diner interacts with the user interface, selects a restaurant, specifies the number of guests, and chooses a date and time for the reservation.

Data Submission

Diner enters personal details (name, surname, email, phone number) and submits the reservation request.

System Validation:

The reservation system validates diner inputs, checking for availability and adherence to business rules.

Unique Reservation ID Generation:

Upon successful validation, the reservation system generates a unique reservation ID and associates it with the diner's reservation.

Reservation Confirmation

The reservation system updates its records, marking the reserved table or dining area as unavailable during the specified date and time.

Confirmation Message:

The user interface modules present a confirmation message to the diner, indicating a successful reservation.

Example

Consider a scenario where a diner, Sarah, uses the ReserveWell Application to make a reservation for two at her favorite restaurant. After entering the necessary details and confirming the reservation, the system generates a unique ID, reserves a table, and sends a confirmation message to Sarah's device. If payment is required, the Payment Authorization Service ensures a secure transaction with the bank. This example illustrates a seamless and optimized reservation workflow facilitated by the pattern.

Requirement realizations

View of participants

Diner

- Behavior: Initiates the reservation process, provides necessary information, and confirms the reservation.
- Attributes: Name, surname, email, phone number, selected restaurant, number of guests, preferred date and time.
- Relationships: Interacts with the user interface, triggers reservation confirmation.

Restaurant Manager

- Behavior: Manages reservations effectively, oversees real-time updates, and ensures a smooth dining experience.
- Attributes: Access to reservation management tools, real-time updates on reservations and table availability.
- Relationships: Interacts with the system to optimize table allocation and enhance the dining experience.

Waitstaff

- Behavior: Provides efficient and high-quality service, organizes tables based on real-time reservation updates.
- Attributes: Access to tools for table arrangement, real-time updates on table availability.
- Relationships: Interacts with the system to optimize table arrangements and contribute to positive customer service.

Reservation System

- Behavior: Validates diner inputs, generates unique reservation IDs, updates system records.
- Attributes: Reservation ID, restaurant availability status, reservation details.
- Relationships: Receives inputs from the diner, interacts with the Payment Authorization Service if required.

User Interface Modules

- Behavior: Presents restaurant options, collects and displays reservation details, communicates confirmation messages.
- Attributes: User interface components, reservation status messages.
- Relationships: Facilitates interaction between the diner and the Reservation System.

Basic scenario

Diner Initiates Reservation

Diner interacts with the user interface, selects a restaurant, and specifies the number of guests, date, and time for the reservation.

Data Submission

Diner enters personal details, and the system collects and validates the reservation request.

System Validation

The Reservation System validates diner inputs, checks restaurant availability, and confirms the reservation's feasibility.

Unique Reservation ID Generation

Upon successful validation, the Reservation System generates a unique ID and associates it with the reservation.

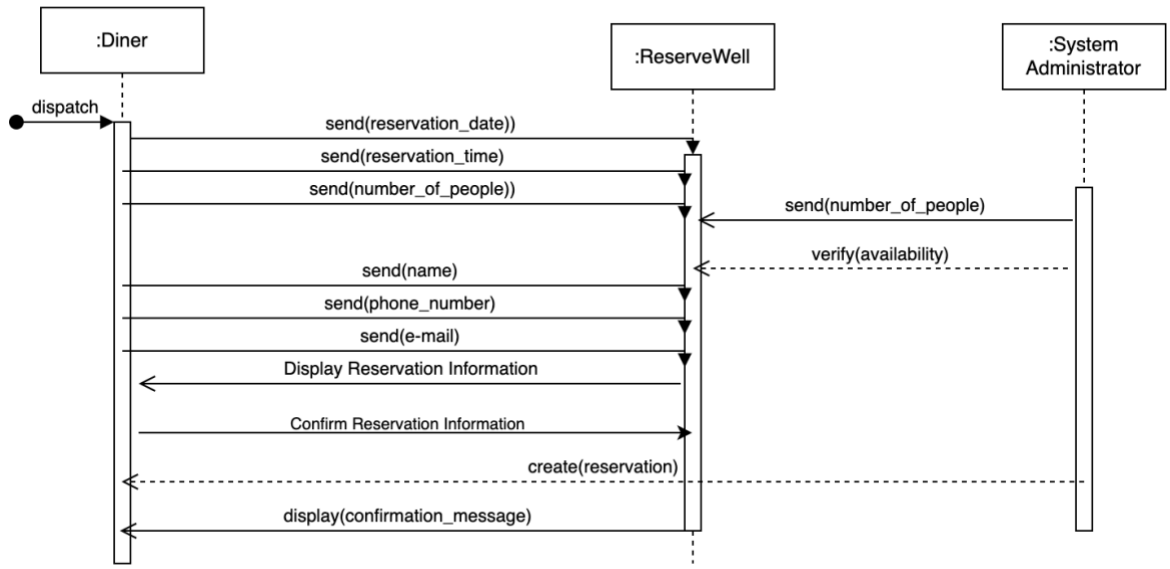
Reservation Confirmation

The Reservation System updates its records, marking the reserved table or dining area as unavailable during the specified date and time.

Confirmation Message

The User Interface Modules present a confirmation message to the diner, indicating a successful reservation.

Interaction Diagram For Make Reservation



Additional scenarios

Scenario 1: System Failure and Recovery

In the event of a system failure, the diner restarts the system, triggering a request for recovery of the prior state. The system then reconstructs the prior state, ensuring a seamless continuation of the reservation process.

Scenario 2: Anomalies preventing recovery

If the system detects anomalies preventing recovery, it signals an error to the diner, records the error, and enters a clean state. The diner is then prompted to start a new reservation, ensuring data integrity and system stability.

Scenario: Diner Entered Incorrect Information

In case the diner enters incorrect e-mail address, the system notifies the diner and prompts the diner to re-enter the information until successfully verified.

[Pattern 2-Register As A Diner]

Overview

The "Register as a Diner" pattern facilitates the process by which users, specifically diners, can create accounts within the ReserveWell application. This pattern ensures a streamlined and user-friendly registration experience, allowing diners to access personalized features such as making reservations, providing feedback, and enjoying a tailored user interface.

Structure

Diner:

Attributes: Name, email, password.
Responsibility: Initiating the registration process, entering personal details, confirming registration.

System:

Attributes: User database, communication module, authentication logic.
Responsibility: Orchestrating the registration process, interacting with user interface and communication modules.

Behavior

From a dynamic perspective, the diner collaborates with the System to enters her information and to get verified by the system.

Example

John, a new user, opens the ReserveWell application and clicks on the "Register As A Diner" button. He fills out the registration form with his name, email and password. After clicking "Register," he completes the registration process, and John can now log in as a registered diner.

Requirement realizations

View of participants

Diner: Interacts with the user interface to input registration details.
System: Facilitates the registration process, manages user accounts, and ensures email address verification.

Basic Scenario

Initiation

The user initiates the registration process by clicking on the "Register As a Diner" button on the main screen of the ReserveWell application. The system responds by presenting the user with the registration form, prompting them to enter the required information.

Data Entry

The enters her information, including name, surname, e-mail. The system validates the entered data in real-time, providing feedback on the correctness of the information. It ensures that mandatory fields are filled, the email format is correct, and the password meets security requirements.

Confirmation and Registration

The user reviews the entered information and clicks the "Register" button to confirm their registration. The system processes the registration request, creating a new user account in the database. The system generates a unique user ID.

Completion

The user is now successfully registered and can log in using their credentials. The system acknowledges the successful registration, updating the user interface to reflect the new user's registration status. The user is now a recognized and authenticated member of the ReserveWell application.

Additional scenarios

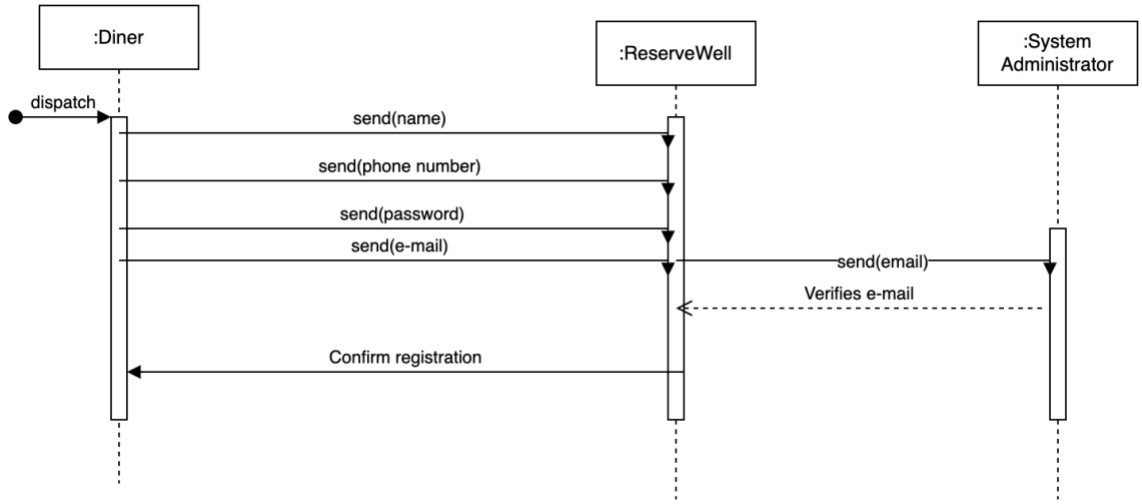
Scenario 1: System Failure and Recovery

In the event of a system failure, the diner restarts the system, triggering a request for recovery of the prior state. The system then reconstructs the prior state, ensuring a seamless continuation of the reservation process.

Scenario 2: Diner's Email Address Cannot Be Verified

If the system detects that the confirmation link is either expired, invalid, or does not match the user account. An error message is displayed on the user interface, Until the diner enters verified e-mail, system keeps to Show same warning.

Interaction Diagram for “Register As A Diner”



[Pattern 3-Register As A Restaurant Manager]

Overview

The "Register as a Restaurant Manager" pattern enables individuals responsible for restaurant management to create accounts within the ReserveWell application. This process allows restaurant managers to access and utilize the platform's features for effective reservation and dining experience management.

Structure

Restaurant Manager:

Attributes: Name, email, password and restaurant information.
Responsibility: Initiating the registration process, entering personal details and restaurant information.

System:

Attributes: User database, communication module, authentication logic.
Responsibility: Facilitates the registration process, manages restaurant manager accounts.

Behavior

From a dynamic perspective, the restaurant manager collaborates with the System to enters her information, restaurant information and to get verified by the system.

Example

Consider a scenario where a new restaurant manager, John Smith, wants to register with the ReserveWell application to manage reservations for his restaurant. John follows the user interface prompts, enters his information and his restaurant's information. The system updates its records, and John Smith gains access to the platform's features as a registered restaurant manager.

Requirement realizations

View of participants

Diner: Interacts with the user interface to input registration details.
System: Facilitates the registration process, manages user accounts, and ensures email address verification.

Basic Scenario

Initiation

The user initiates the registration process by clicking on the "Register As a Diner" button on the main screen of the ReserveWell application. The system responds by presenting the user with the registration form, prompting them to enter the required information.

Data Entry

The enters her information, including name, surname, e-mail. Also, he/she enter restaurant information. The system validates the entered data in real-time, providing feedback on the correctness of the information. It ensures that mandatory fields are filled, the email format is correct, and the password meets security requirements.

Confirmation and Registration

The user reviews the entered information and clicks the "Register" button to confirm their registration. The system processes the registration request, creating a new user account in the database. The system generates a unique user ID.

Completion

The user is now successfully registered and can log in using their credentials. The system acknowledges the successful registration, updating the user interface to reflect the new user's registration status. The user is now a recognized and authenticated member of the ReserveWell application.

Additional scenarios

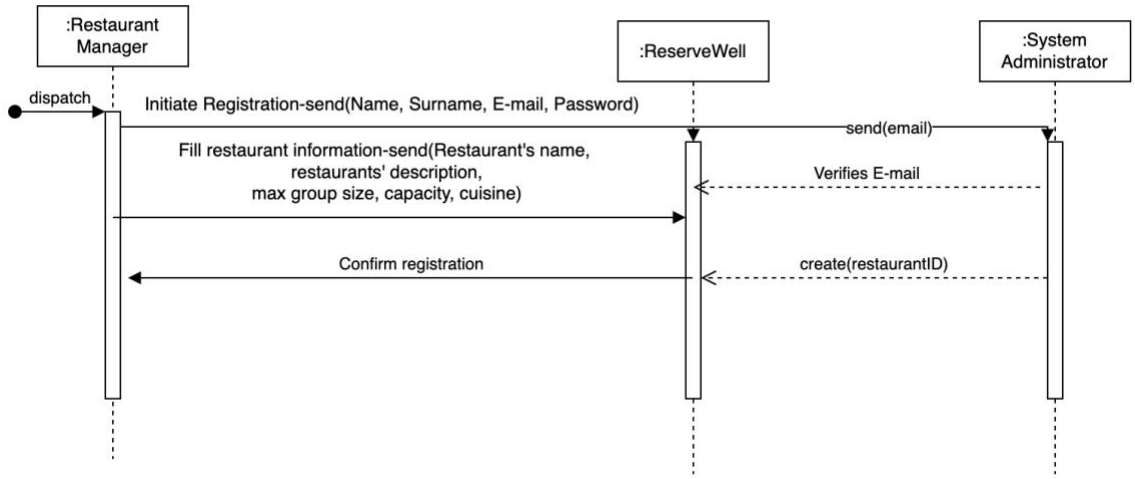
Scenario 1: System Failure and Recovery

In the event of a system failure, the diner restarts the system, triggering a request for recovery of the prior state. The system then reconstructs the prior state, ensuring a seamless continuation of the reservation process.

Scenario 2: Restaurant Manager's Email Address Cannot Be Verified

If the system detects that the confirmation link is either expired, invalid, or does not match the user account. An error message is displayed on the user interface, Until the restaurant manager enters verified e-mail, system keeps to show same warning.

Interaction Diagram for “Register As A Restaurant Manager”



[Pattern 4- Edit Capacity]

Overview

The Capacity Edit pa.ern facilitates the dynamic adjustment of seatng capacity in response to changing demand conditons, optmizing operatonal efficiency, and supportng workforce allocaton planning.

Structure

Restaurant Manager:

Role: Initates and manages the process of editng seatng capacity.

Relevant Data: Accesses and updates seatng capacity informaton, including the current capacity and new values for adjustment.

Behavior: Utlizes tools and functonalites to oversee the seatng capacity editng process and ensure smooth restaurant operatons.

Waitstaff:

Role: Receives real-tme updates on seatng capacity changes to optmize table arrangements.

Relevant Data: Accesses updated seatng capacity informaton for fair distributon of seatng assignments.

Behavior: Utlizes tools within the applicaton to organize seatng arrangements based on capacity changes.

Diners (Customers):

Role: Affected by changes in the restaurant's seatng capacity, requires awareness of any modificatons.

Relevant Data: Needs informaton about any changes in the restaurant's seatng capacity to plan visits accordingly.

Behavior: Relies on accurate and up-to-date seatng capacity informaton provided by the system to reduce inconvenience.

Restaurant Owners:

Role: Has an interest in accurate seatng capacity edits for cost-effectve operatons, adaptability to market trends, and maintaining a positve reputaton.

Relevant Data: Accesses historical data and logs to review changes in seatng capacity for operatonal and strategic decision-making.

Behavior: Monitors the accuracy of seatng capacity updates to ensure optmal restaurant performance.

Development Team:

Role: Ensures the reliability and stability of the system during and aNer seatng capacity edits.

Relevant Data: Concerned with system reliability, real-tme updates, mobile responsiveness, and security measures during seatng capacity adjustments.

Behavior: Implements measures to safeguard sensitve data and maintain system stability during dynamic changes in seatng capacity.

Reservation System:

Role: Central component responsible for managing seatng capacity edits.

Relevant Data: Stores and processes informaton related to current and updated seatng capacity.

Behavior: Validates and records changes in seatng capacity, ensuring that the updated capacity is reflected across the system for upcoming transactons.

User Interface Modules:

Role: Facilitates interaction between the Restaurant Manager and the system for editng seating capacity.

Relevant Data: Displays current seating capacity, customization options, and confirmaton messages.

Behavior: Presents a user-friendly interface for the Restaurant Manager to input new seating capacity, confirms changes, and communicates the success of the update.

Behavior

From a dynamic perspectve, the Restaurant Manager collaborates with the System to adjust seatng capacity, ensuring adaptability to market trends and operatonal optmizaton.

Example

Consider a scenario where a restaurant experiences a sudden increase in reservatons. The Capacity Edit pattern allows the Restaurant Manager to promptly adjust seating capacity to accommodate the higher demand.

Requirement Realizations

View of participants

Restaurant Manager: Initates and confirms capacity changes.
System: Facilitates capacity adjustments and updates related data in real-tme.

Basic scenario

Restaurant Manager Initiates Capacity Edit:

The restaurant manager interacts with the user interface to edit seatng capacity and the system displays the existng capacity.

Data Submission:

The Restaurant Manager enters the new capacity and submits the update request, and the system enables adjustng the capacity based on the provided values.

System Validation:

The system validates the working hours adjustments, ensures feasibility with the existng reservatons

Confirmation and Update:

Upon system validaton, the restaurant manager to confirm the changes, and the system updates its records with the new seatng capacity.

Confirmation Message:

The user interface modules present a confirmaton message to the Restaurant Manager, indicatng a successful seatng capacity update, which will be valid for upcoming transactons.

Additional Scenarios

Scenario 1: Abandon Process - Restaurant Manager Quits the Page

If the Restaurant Manager chooses to abandon the capacity editng process by quiting the page, the system offers optons to discard, review, save changes, or cancel quiting. Discarding

changes reconstructs the prior state, reviewing changes allows discarding or saving, and canceling quiting returns to the edit capacity page.

Scenario 2: Abandon Process - Restaurant Manager Saves the Changes

If the Restaurant Manager saves changes before quiting, the system confirms the update, displays a "Capacity Updated" message, and returns to the home page.

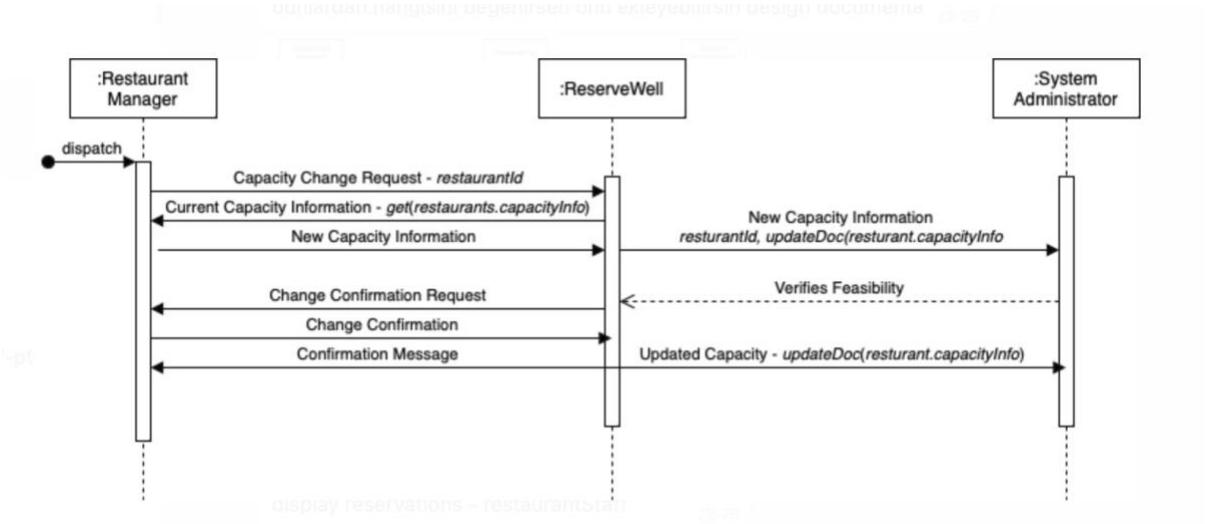
Scenario 3: Abandon Process - Restaurant Manager Cancels Quiting

Choosing to cancel quiYng gracefully returns the system to the edit capacity page, enabling the Restaurant Manager to contnue seamlessly.

Scenario 4: System Failure and Recovery

In case of a system failure, the Restaurant Manager restarts the system, triggering recovery. The system reconstructs the prior state unless anomalies preventng recovery are detected. In such cases, an error is signaled, recorded, and sent for review.

Interaction Diagram For Edit Capacity



[Pattern 5-Register As Restaurant Manager]

Overview

The "Register as Restaurant Manager" pattern facilitates the process by which individuals sign up to become restaurant managers within the ReserveWell application and add his restaurants to the application. This pattern aims to streamline the onboarding experience for new restaurant managers, ensuring they can efficiently create accounts and access the system's management features.

Structure

Restaurant Manager:

Attributes: Restaurant’s name, cuisine, short description, capacity, max group size, and restaurant manager’s name, email, password.

Responsibility: Initiates the registration process by providing necessary details.

System:

Attributes: User database, communication module, authentication logic.
Responsibility: Manages the registration process, data storage, and authentication.

User Interface Modules:

Role:Facilitates interaction between the Restaurant Manager and the system.
Relevant Data: Displays fields for user to fill the information.
Behavior: Presents a user-friendly interface for the Restaurant Manager to process registration.

Behavior

From a dynamic perspective, the restaurant manager collaborates with the System to enters her information, and restaurant's information and to get verified by the system.

Example

Jane, a restaurant manager, decides to join ReserveWell to efficiently handle reservations for her restaurant. She navigates to the registration page, provides accurate details, and submits the form. After verifying her email, Jane's account and restaurant are successfully created, allowing her to log in and manage her restaurant's reservations using the ReserveWell application.

Requirement realizations

View of participants

Restaurant Manager: Interacts with the user interface to input registration details.
System: Facilitates the registration process, manages user accounts, and ensures email address verification.

Basic Scenario

Initiation

The user initiates the registration process by clicking on the "Register As a Restaurant Manager" button on the main screen of the ReserveWell application. The system responds by presenting the user with the registration form, prompting them to enter the required information.

Data Entry

The restaurant manager enters her information, including name, surname, e-mail, password. Also, he/she provides information of his/her restaurant. The system validates the entered data in real-time, providing feedback on the correctness of the information. It ensures that mandatory fields are filled, the email format is correct, and the password meets security requirements.

Confirmation and Registration

The user reviews the entered information and clicks the "Register" button to confirm their registration. The system processes the registration request, creating a new user account and restaurant in the database. The system generates a unique user and restaurant ID.

Completion

The user is now successfully registered and can log in using their credentials. The system acknowledges the successful registration, updating the user interface to reflect the new user's registration status. The user is now a recognized and authenticated member of the ReserveWell application.

Additional scenarios

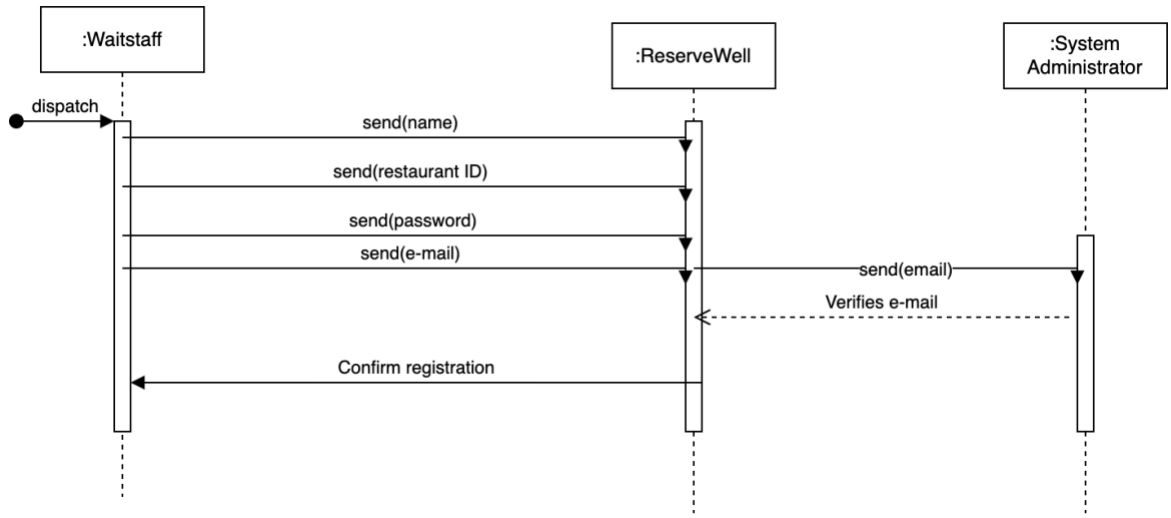
Scenario 1: System Failure and Recovery

In the event of a system failure, the restaurant manager restarts the system, triggering a request for recovery of the prior state. The system then reconstructs the prior state, ensuring a seamless continuation of the reservation process.

Scenario 2: Restaurant Manager's Email Address Cannot Be Verified

If the system detects that the confirmation link is either expired, invalid, or does not match the user account. An error message is displayed on the user interface, Until the restaurant manager enters verified e-mail, system keeps to show same warning.

Interaction Diagram for “Register As A Restaurant Manager”



[Pattern 6-Register As A Waitstaff]

Overview

John, a new waiter at a restaurant, needs to use ReserveWell for efficient table management. He navigates to the registration page, provides accurate details, and submits the form. After verifying his email, John's account is successfully created, enabling him to log in and utilize ReserveWell features to enhance his service at the restaurant.

Structure

Waitstaff:

Attributes: The waitstaff's name, email, password.
Responsibility: Initiates the registration process by providing necessary details.

System:

Attributes: User database, communication module, authentication logic.
Responsibility: Manages the registration process, data storage, and authentication.

User Interface Modules:

Role:Facilitates interaction between the Waitstaff and the system for editing seating capacity.
Relevant Data: Displays fields for user to fill the information.
Behavior: Presents a user-friendly interface for the waitstaff to process registration.

Behavior

From a dynamic perspective, the waitstaff collaborates with the System to enters her information and to get verified by the system.

Example

Jane, a restaurant manager, decides to join ReserveWell to efficiently handle reservations for her restaurant. She navigates to the registration page, provides accurate details, and submits the form. After verifying her email, Jane's account and restaurant are successfully created, allowing her to log in and manage her restaurant's reservations using the ReserveWell application.

Requirement realizations

View of participants

Waitstaff: Interacts with the user interface to input registration details.
System: Facilitates the registration process, manages user accounts, and ensures email address verification.

Basic Scenario

Initiation

The user initiates the registration process by clicking on the "Register As a Waitstaff" button on the main screen of the ReserveWell application. The system responds by presenting the user with the registration form, prompting them to enter the required information.

Data Entry

The waitstaff enters her information, including name, surname, e-mail and password. The system validates the entered data in real-time, providing feedback on the correctness of the information. It

ensures that mandatory fields are filled, the email format is correct, and the password meets security requirements.

Confirmation and Registration

The user reviews the entered information and clicks the "Register" button to confirm their registration. The system processes the registration request, creating a new user account. The system generates a unique user ID.

Completion

The user is now successfully registered and can log in using their credentials. The system acknowledges the successful registration, updating the user interface to reflect the new user's registration status. The user is now a recognized and authenticated member of the ReserveWell application.

Additional scenarios

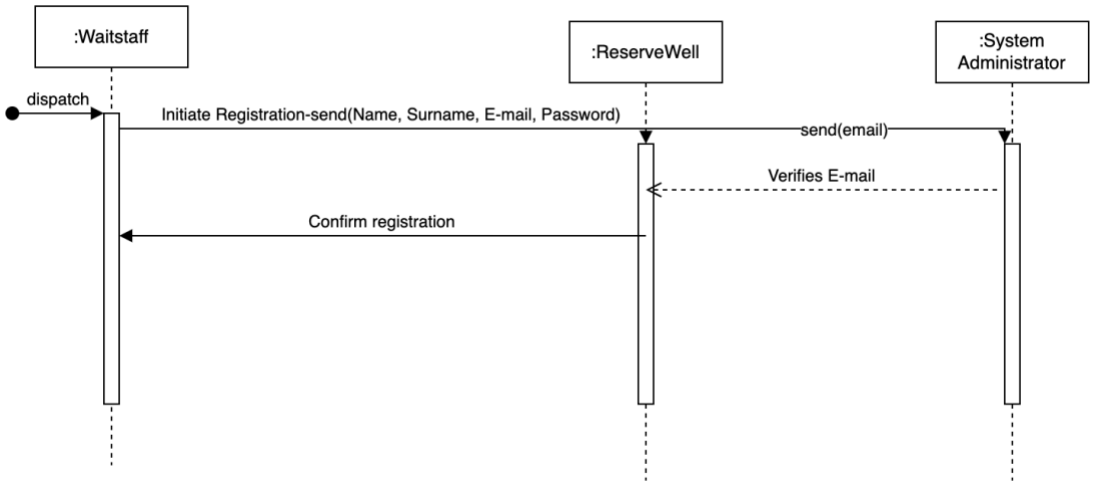
Scenario 1: System Failure and Recovery

In the event of a system failure, the waitstaff restarts the system, triggering a request for recovery of the prior state. The system then reconstructs the prior state, ensuring a seamless continuation of the reservation process.

Scenario 2: Waitstaff's Email Address Cannot Be Verified

If the system detects that the confirmation link is either expired, invalid, or does not match the user account. An error message is displayed on the user interface, Until the waitstaff enters verified e-mail, system keeps to show same warning.

Interaction Diagram for “Register As A Waitstaff”



[Pattern 7-Manage Reservations]

Overview

The Reservation Management pattern facilitates accurate and efficient reservation handling, ensuring real-time updates to reservations, restaurant capacity, and waitlist information. The pattern streamlines the reservation management process, providing a user-friendly interface for the Restaurant Manager.

Structure

- **Restaurant Manager:**
  - Role: Initiates and manages the reservation management process.
  - Relevant Data: Accesses and updates reservation details, including reservation ID, number of guests, special requests, reservation hours, and date.
  - Behavior: Utilizes tools and functionalities to oversee reservation changes and ensure a smooth reservation management process.
- **Waitstaff:**
  - Role: Receives real-time updates on reservation changes for efficient table arrangements.
  - Relevant Data: Accesses updated reservation information to organize table assignments based on reservation updates.
  - Behavior: Utilizes tools within the application to coordinate table arrangements with the latest reservation changes.
- **Diners (Customers):**
  - Role: Affected by changes in reservation details, requires awareness of any modifications.
  - Relevant Data: Needs information about reservation updates to plan visits accordingly.
  - Behavior: Relies on accurate and up-to-date reservation information provided by the system to reduce inconvenience.
- **Restaurant Owners:**
  - Role: Has an interest in accurately recorded reservations for operational efficiency and customer satisfaction.
  - Relevant Data: Accesses historical data and logs to review changes in reservations for strategic decision-making.
  - Behavior: Monitors the accuracy of reservation updates to ensure optimal restaurant performance.

- **Development Team:**
  - Role: Ensures the reliability and stability of the system during reservation updates.

    - Relevant Data: Concerned with system reliability, real-time updates, scalability, security measures, and adherence to best practices during reservation changes.
    - Behavior: Implements measures to safeguard sensitive data and maintain system stability during dynamic changes in reservations.
- **Reservation System:**
    - Role: Central component responsible for managing reservation updates.
    - Relevant Data: Stores and processes information related to current and updated reservations.
    - Behavior: Validates and records changes in reservations, ensuring that the updated reservations are reflected across the system for upcoming transactions.
- **User Interface Modules:**
    - Role: Facilitates interaction between the Restaurant Manager and the system for managing reservations.
    - Relevant Data: Displays current reservation details, customization options, and confirmation messages.
    - Behavior: Presents a user-friendly interface for the Restaurant Manager to input reservation changes, confirm updates, and communicate the success of the changes.

Behavior

From a dynamic perspective, the Restaurant Manager collaborates with the system to manage reservations efficiently, ensuring real-time updates and streamlined table allocation.

Example

Consider a scenario where a restaurant experiences last-minute changes in reservation details. The Reservation Management pattern allows the Restaurant Manager to promptly adjust reservation information, ensuring optimal table allocation and a positive dining experience for customers.

Requirement Realizations

View of Participants

- Restaurant Manager: Initiates and confirms reservation changes.
- System: Facilitates reservation adjustments and updates related data in real-time.

Basic Scenario

1. **Restaurant Manager Initiates Reservation Management:**

The restaurant manager interacts with the user interface to manage reservations, and the system displays the existing reservations.

2. **Data Submission:**

The Restaurant Manager enters the reservation ID to view and update its details, and the system enables adjusting reservation details based on the provided values.

3. **System Validation:**

The system validates the reservation adjustments, ensuring feasibility with the existing reservations.

4. **Confirmation and Update:**

Upon system validation, the restaurant manager confirms the changes, and the system updates the reservation data in real-time and adjusts available capacity accordingly.

5. **Notification:**

The system sends notifications to diners about the reservation updates, and waitlist availability notifications are sent to customers on the restaurant's waitlist.

Additional Scenarios

Scenario 1: Abandon Process - Restaurant Manager Quits the Page

- If the Restaurant Manager chooses to abandon the reservation management process by quitting the page, the system offers options to discard, review, save changes, or cancel quitting.

Scenario 2: Abandon Process - Restaurant Manager Saves the Changes

- If the Restaurant Manager saves changes before quitting, the system confirms the update, displays a "Reservation Updated" message, and returns to the home page.

Scenario 3: Abandon Process - Restaurant Manager Cancels Quitting

- Choosing to cancel quitting gracefully returns the system to the reservation management page, enabling the Restaurant Manager to continue seamlessly.

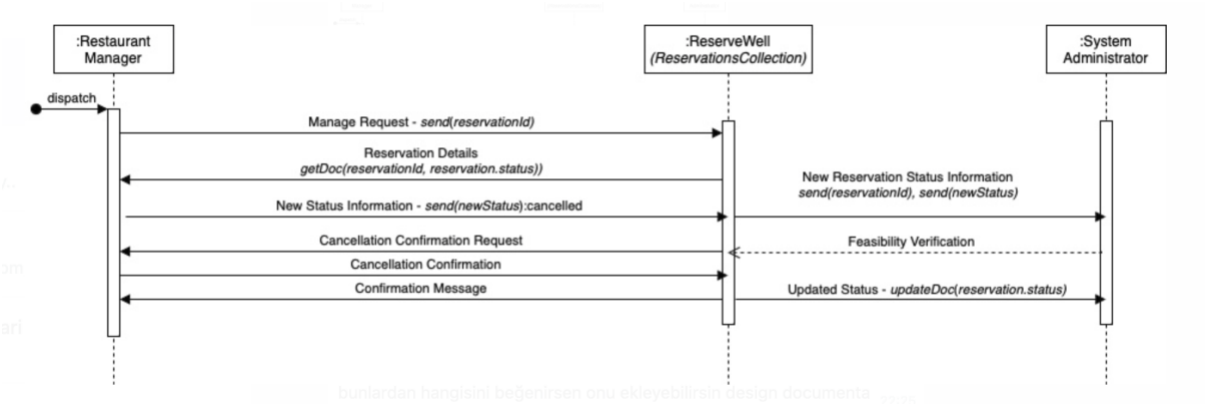
Scenario 4: System Failure and Recovery

- In case of a system failure, the Restaurant Manager restarts the system, triggering recovery. The system reconstructs the prior state unless anomalies preventing recovery are detected. In such cases, an error is signaled, recorded, and sent for review.

Scenario 5: Various Reservation Details Changes

- The restaurant manager can edit various reservation details, including the number of guests, special requests, reservation hours, date, and cancel the reservation. The system updates the respective information accordingly.

Interaction Diagram For Manage Reservations



[Pattern 8-Update Reservations]

The Update Reservation pattern for Diners focuses on providing flexibility for diners to modify reservations based on changing plans or preferences. It ensures accurate and efficient reservation management, updating the reservation system database, and reflecting the changes in the user interface of the application.

Structure

- **Diner:**
  - Role: Initiates and manages the process of updating reservations.
  - Relevant Data: Accesses and updates reservation details, including date, time, party size, and special requests.
  - Behavior: Utilizes tools and functionalities to edit reservations and ensure a smooth reservation updating process.
- **Restaurant Manager:**
  - Role: Wants accurate and efficient reservation management and optimized table allocation.
  - Relevant Data: Monitors reservation updates for operational decision-making.
  - Behavior: Expects a clear and organized display of updated reservations to ensure efficient restaurant management.
- **Waitstaff:**
  - Role: Wants to provide efficient and high-quality service to customers.
  - Relevant Data: Receives real-time updates on reservation changes for efficient table arrangements.
  - Behavior: Utilizes tools within the application to coordinate table arrangements with the latest reservation updates.
- **Development Team:**
  - Role: Ensures the reliability and stability of the system during reservation updates.

    - Relevant Data: Concerned with system reliability, real-time updates, scalability, security measures, and adherence to best practices during reservation changes.
    - Behavior: Implements measures to safeguard sensitive data and maintain system stability during dynamic changes in reservations.
- **Reservation System:**
    - Role: Central component responsible for managing and updating reservation details.
    - Relevant Data: Stores and processes information related to current and updated reservations.
    - Behavior: Validates and records changes in reservations, ensuring that the updated reservations are reflected across the system.
- **User Interface Modules:**
    - Role: Facilitates interaction between the Diner and the system for updating reservations.
    - Relevant Data: Displays current reservation details, customization options, and confirmation messages.
    - Behavior: Presents a user-friendly interface for the Diner to input reservation changes, confirm updates, and receive notifications.

Behavior

From a dynamic perspective, the Diner collaborates with the system to efficiently update reservations, ensuring real-time updates and smooth communication with the restaurant.

Example

Consider a scenario where a diner needs to modify a reservation due to a change in plans. The Update Reservation pattern allows the diner to promptly edit reservation details, confirm changes, and receive notifications, ensuring a positive dining experience.

Requirement Realizations

View of Participants

- Diner: Initiates and confirms reservation changes.
- System: Facilitates reservation adjustments and updates related data in real-time.

Basic Scenario

1. **Access Personal Reservations:**

The diner wants to update reservations and accesses their existing reservations by clicking the 'My Reservations' button.

2. **Display Personal Reservations Page:**

The system displays a personal reservations page.

3. **Select Reservation for Update:**

The diner needs to update a specific reservation and selects the reservation they wish to update.

4. **Enable Editing:**

The system enables editing.

5. **Modify Reservation Details:**

The diner modifies the reservation details, such as date, time, party size, or special requests.

6. **Confirmation and Update:**

The system shows the change info and asks for confirmation.

7. **Diner Confirms Changes:**

The diner confirms the changes to be updated in the system.

8. **Update Reservation and Notify:**

The system updates the reservation, notifies the restaurant, and sends a confirmation to the diner.

Extensions (or Alternative Flows)

Scenario 1: Abandon Process - Diner Quits the Page

- If the diner chooses to abandon the reservation update process by quitting the page, the system offers options to discard, review, save changes, or cancel quitting.

Scenario 2: System Failure and Recovery

- In case of a system failure, the diner restarts the system, logs in, and requests recovery of the prior state. The system reconstructs the prior state.

Scenario 3: Invalid Reservation Selection

- If the diner selects an invalid reservation, the system informs the diner and returns them to the personal reservations page.

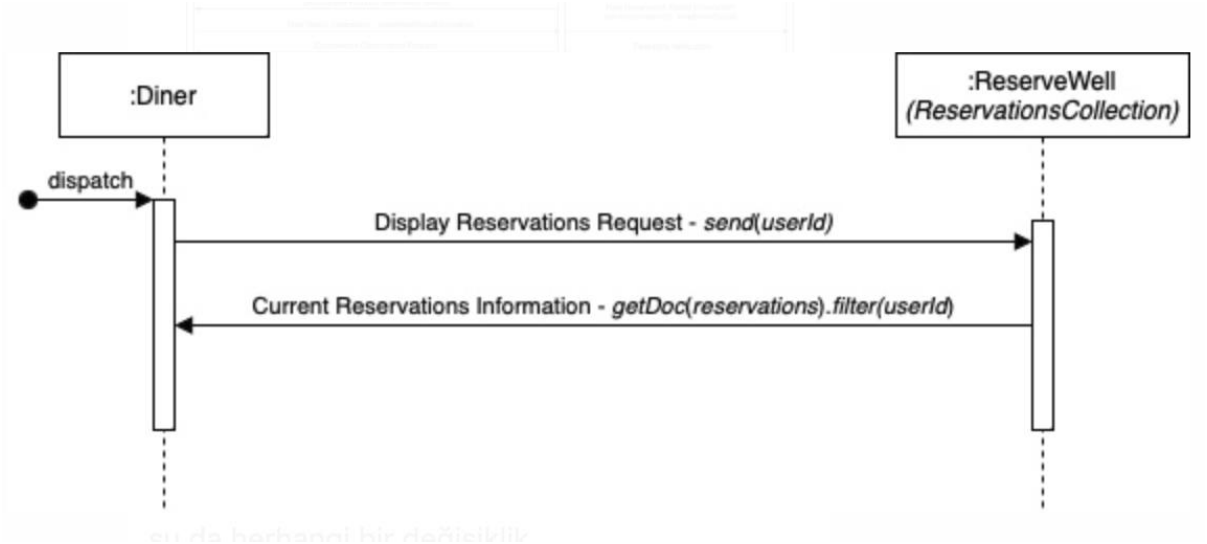
Scenario 4: Invalid Reservation Update Information

- If the diner tries to update the reservation with invalid information (e.g., exceeding capacity), the system informs the diner, and the diner revises the changes.

Special Requirements

- The reservation system database should be updated with the changes.
- The changes should be reflected in the user interface of the application.

Interaction Diagram For Update Reservations



[Pattern 9-Display Reservations/Waitstaff]

Overview

The Display Reservations pattern for Waitstaff facilitates the clear visualization of reservation data and real-time updates on reservation statuses. It provides a user-friendly interface for waitstaff to efficiently view reservations and coordinate table arrangements based on timely reservation information.

Structure

- **Waitstaff:**
  - Role: Accesses and views reservation information promptly.
  - Relevant Data: Views reservation details, including reservation ID, number of guests, and hour details.
  - Behavior: Utilizes tools and functionalities to navigate through the reservations page, apply filters, and confirm realized reservations.
- **Restaurant Manager:**
  - Role: Wants clear visualization of reservation data and real-time updates on reservation statuses.
  - Relevant Data: Monitors reservation information for operational decision-making.
  - Behavior: Expects a clear and organized display of reservations to ensure efficient restaurant management.
- **Diners (Customers):**
  - Role: Affected by changes in reservation details, requires awareness of any modifications.
  - Relevant Data: Needs information about reservation updates to plan visits accordingly.

    - Behavior: Relies on accurate and up-to-date reservation information provided by the system to reduce inconvenience.
- **Restaurant Owners:**
    - Role: Has an interest in accurately recorded reservations for operational efficiency and customer satisfaction.
    - Relevant Data: Accesses historical data and logs to review changes in reservations for strategic decision-making.
    - Behavior: Monitors the accuracy of reservation updates to ensure optimal restaurant performance.
- **Development Team:**
    - Role: Ensures the reliability and stability of the system during reservation display.
    - Relevant Data: Concerned with system reliability, real-time updates, scalability, security measures, and adherence to best practices during reservation information retrieval.
    - Behavior: Implements measures to safeguard sensitive data and maintain system stability during dynamic reservation display.
- **Reservation System:**
    - Role: Central component responsible for managing and displaying reservation information.
    - Relevant Data: Stores and processes information related to current reservations and displays it to waitstaff in real-time.
- **User Interface Modules:**
    - Role: Facilitates interaction between the Waitstaff and the system for displaying reservations.
    - Relevant Data: Displays current reservation details and provides options for applying filters and confirming realized reservations.
    - Behavior: Presents a user-friendly interface for the Waitstaff to navigate through reservation information.

Behavior

From a dynamic perspective, the Waitstaff collaborates with the system to efficiently view reservations, apply filters, and confirm realized reservations.

Example

Consider a scenario where the waitstaff needs to quickly access reservation details during a busy period. The Display Reservations pattern allows the waitstaff to promptly view reservations, apply filters, and confirm realized reservations, ensuring smooth collaboration among staff members.

Requirement Realizations

- Waitstaff: Accesses and views reservation information promptly.

- System: Facilitates the display of reservations, applies filters, and confirms realized reservations.

Basic Scenario

1. **Access Reservation Display Screen:**
    - The waitstaff wants to view reservations and accesses the reservation display screen by clicking the "Display" button.
2. **Display Reservations Page:**
    - The system displays the reservations page.
3. **View Reservations:**
    - The waitstaff views a list of existing reservations with reservation ID, number of guests, and hour details in an increasing hour order.

Extensions (or Alternative Flows)

Scenario 1: Abandon Process - Waitstaff Quits the Page

- If the waitstaff chooses to abandon the reservation display process by quitting the page, the system displays the home page.

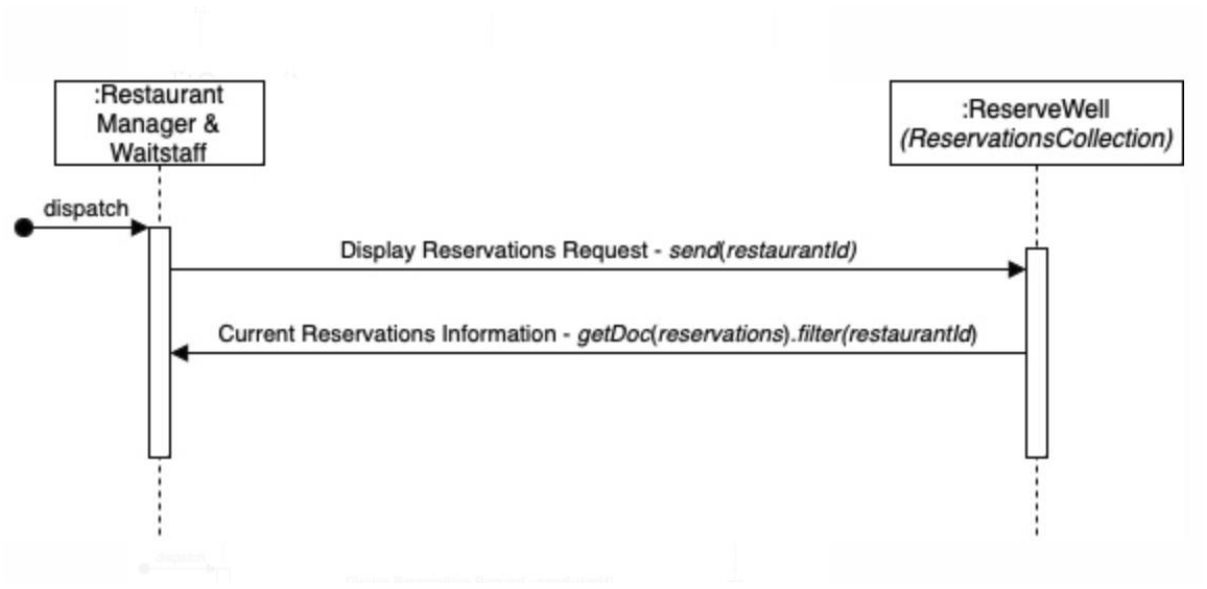
Scenario 2: System Failure and Recovery

- In case of a system failure, the waitstaff restarts the system. The system displays the login page, and after logging in, it displays the home page.

Scenario 3: Filtering Reservations

- If the waitstaff needs to filter reservations, they select conditions from the filter lists (date, hour) and select apply. The system applies selected filters and narrows the reservations list

Interaction Diagram For Display Reservations/Waitstaff]



[Pattern 10-Display Reservations/Restaurant Manager]

Overview

The Display Reservations pattern for Restaurant Managers focuses on providing a clear visualization of reservation data and real-time updates on reservation statuses. It ensures efficient management of reservations, timely communication with waitstaff, and access to reports and analytics for strategic decision-making.

Structure

- **Restaurant Manager:**
  - **Role:** Initiates and manages the process of displaying reservations.
  - **Relevant Data:** Accesses and views reservation details, including reservation ID, number of guests, and hour details.
  - **Behavior:** Utilizes tools and functionalities to navigate through the reservations page, apply filters, and manage reservations efficiently.
- **Waitstaff:**
  - **Role:** Wants to provide efficient and high-quality service to customers.
  - **Relevant Data:** Receives real-time updates on reservation changes for efficient table arrangements.
  - **Behavior:** Utilizes tools within the application to coordinate table arrangements with the latest reservation updates.
- **Diners (Customers):**
  - **Role:** Wants updating reservations and fast service with minimal effort.
  - **Relevant Data:** Needs information about reservation updates to plan visits accordingly.
  - **Behavior:** Relies on accurate and up-to-date reservation information provided by the system to reduce inconvenience.
- **Restaurant Owners:**
  - **Role:** Has an interest in accurately recorded reservations for operational efficiency and customer satisfaction.
  - **Relevant Data:** Accesses historical data and logs to review changes in reservations for strategic decision-making.
  - **Behavior:** Monitors the accuracy of reservation updates to ensure optimal restaurant performance.
- **Development Team:**
  - **Role:** Ensures the reliability and stability of the system during reservation displays.
  - **Relevant Data:** Concerned with system reliability, real-time updates, scalability, security measures, and adherence to best practices during reservation information retrieval.
  - **Behavior:** Implements measures to safeguard sensitive data and maintain system stability during dynamic reservation display.

- **Reservation System:**
    - **Role:** Central component responsible for managing and displaying reservation information.
    - **Relevant Data:** Stores and processes information related to current and updated reservations.
    - **Behavior:** Ensures real-time updates of reservation statuses and provides a clear display for the restaurant manager.
- **User Interface Modules:**
    - **Role:** Facilitates interaction between the Restaurant Manager and the system for displaying reservations.
    - **Relevant Data:** Displays current reservation details, customization options, and confirmation messages.
    - **Behavior:** Presents a user-friendly interface for the Restaurant Manager to input reservation display preferences, apply filters, and manage reservations.

Behavior

From a dynamic perspective, the Restaurant Manager collaborates with the system to efficiently view reservations, apply filters, and manage reservations, ensuring real-time updates and strategic decision-making.

Example

Consider a scenario where the restaurant manager needs to quickly access reservation details during a busy period. The Display Reservations pattern allows the restaurant manager to promptly view reservations, apply filters, and efficiently manage reservations, ensuring smooth collaboration among staff members.

Requirement Realizations

View of Participants

- **Restaurant Manager:** Initiates and confirms reservation displays, applies, filters, and manages reservations.
- **System:** Facilitates reservation adjustments, applies filters, and provides real-time updates to the Restaurant Manager.

Basic Scenario

1. **Restaurant Manager Initiates Reservation Display:**
    - The system displays the home page.
2. **Access Reservation Display Screen:**
    - The restaurant manager wants to view reservations and access the reservation display screen by clicking the "Display" button.
3. **Display Reservations Page:**
    - The system displays the reservations page.
4. **View Reservations:**

- The restaurant manager views a list of existing reservations with reservation ID, number of guests, and hour details in an increasing hour order.

5. **Exit Display Page:**
    - The restaurant manager exits the display page.

Extensions (or Alternative Flows)

Scenario 1: Abandon Process - Restaurant Manager Quits the Page

- If the restaurant manager chooses to abandon the reservation display process by quitting the page, the system displays the home page.

Scenario 2: System Failure and Recovery

- In case of a system failure, the restaurant manager restarts the system. The system displays the login page, and after logging in, it displays the home page.

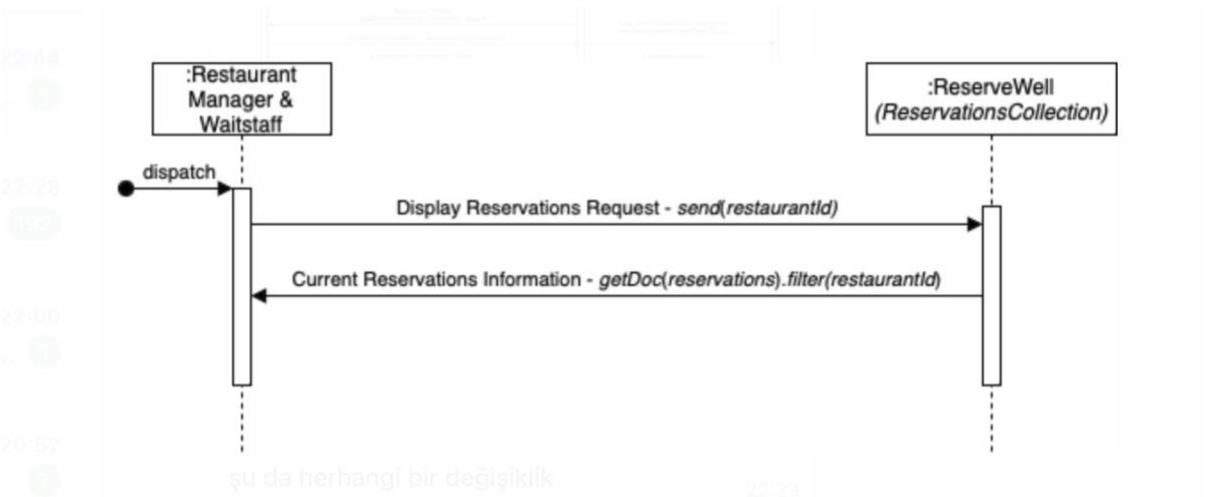
Scenario 3: Filtering Reservations

- If the restaurant manager needs to filter reservations, they select conditions from the filter lists (date, hour) and select apply. The system applies selected filters and narrows the reservations list.

Scenario 4: Managing Reservations

- The "Display Reservations" page should support managing reservations by the restaurant manager, allowing them to efficiently coordinate table arrangements and make strategic decisions based on reservation data.

Interaction Diagram For Display Reservations/Restaurant Manager



Version	Date	Description
v1.0	26.11.2023	-
v1.1	25.11.2023	• According to advisor feedback, subsystems part is updated. • Additional scenario part is revised.
v1.2	16.12.2023	• 3 new patterns are added to the document.
v1.3	1.01.2024	• New patterns are added to the document according to realized use cases at fourth iteration. • Instead of sequence diagrams, interaction diagrams are added.