

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Звіт
до етапу №6 курсової роботи
із дисципліни «Бази даних та інформаційні системи»
на тему
«Оформлення шлюбу»

Виконав:
студент групи КМ-33
Шарапов В.В.

Керівник:
Терещенко І.О.

ЗМІСТ

ВСТУП.....	3
1 ОСНОВНА ЧАСТИНА.....	4
1.1 Діаграма потоків даних нульового рівня.....	4
1.2 Діаграма потоків даних першого рівня.....	5
1.3 Діаграма потоків даних другого рівня для процесу Authenticate User.....	6
1.4 Діаграма потоків даних другого рівня для процесу Manage Artists.....	7
1.5 Діаграма потоків даних другого рівня для процесу Manage Restaurants.....	8
1.6 Діаграма потоків даних другого рівня для процесу Manage Ceremony.....	9
1.7 Діаграма потоків даних другого рівня для процесу Manage Payments.....	10
1.8 Опис структур даних.....	12
ВИСНОВКИ.....	13

ВСТУП

Для інформаційної системи “Оформлення шлюбу” розробити діаграму потоків даних за стандартом DFD до другого рівня включно.

1 ОСНОВНА ЧАСТИНА

1.1 Діаграма потоків даних нульового рівня

Діаграму потоків даних нульового рівня зображено на рис.1.1.

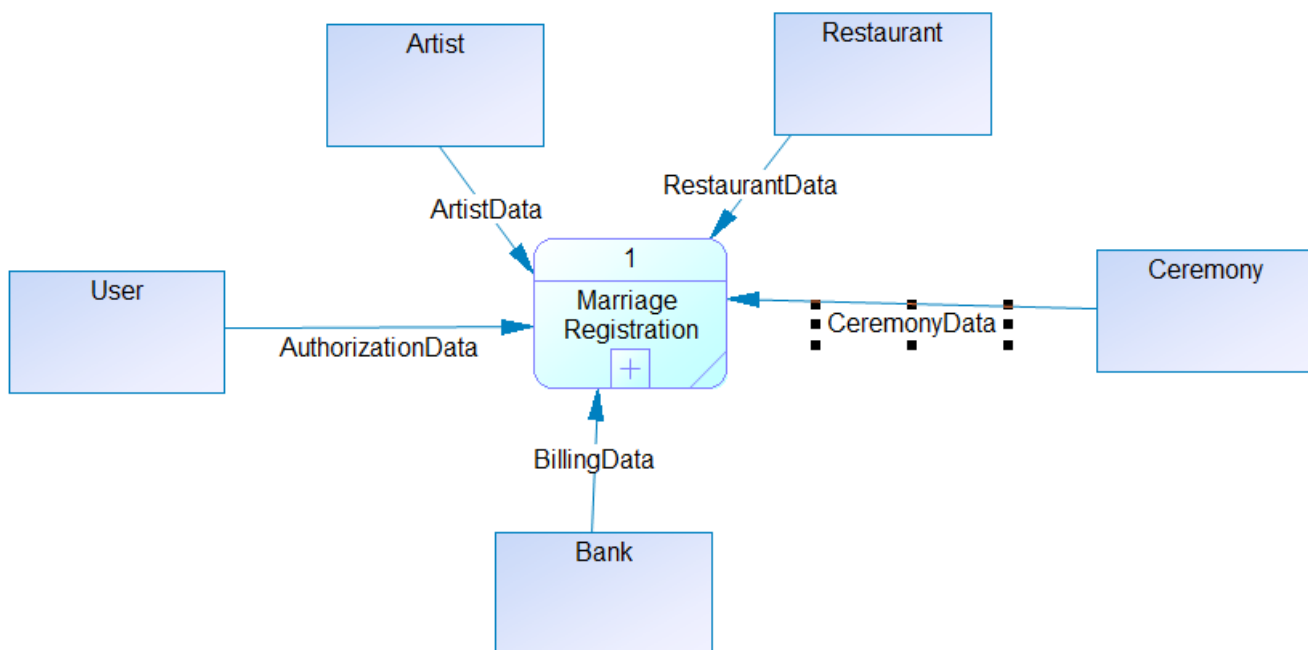


Рисунок 1.1 - Діаграма потоків даних нульового рівня

Головний процес інформаційної системи Marriage Registration. Цей процес отримує: від зовнішньої сутності User дані у структурі AuthorizationData, від зовнішньої сутності Artist дані у структурі ArtistData, від зовнішньої сутності Restaurant дані у структурі RestaurantData, від зовнішньої сутності Ceremony дані у структурі CeremonyData, від зовнішньої сутності Bank дані у структурі BillingData.

1.2 Діаграма потоків даних першого рівня

Діаграму потоків даних першого рівня зображено на рис.1.2.

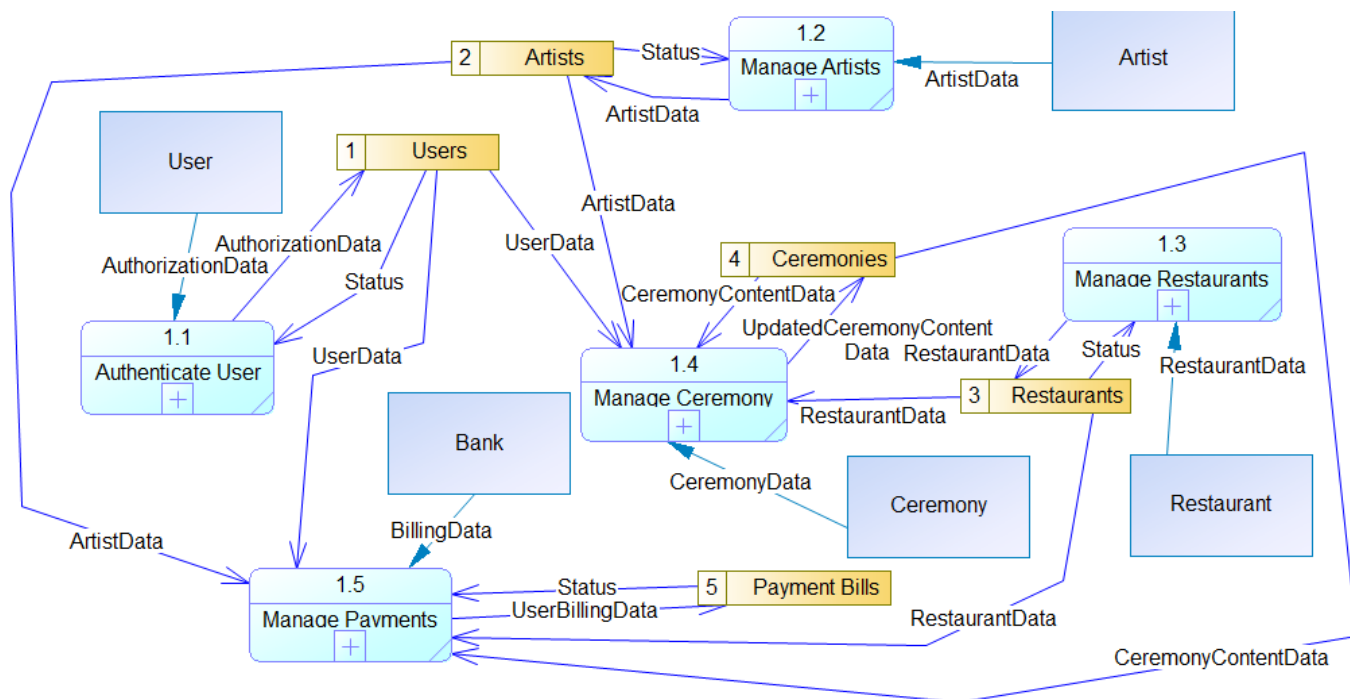


Рисунок 1.2 - Діаграма потоків даних першого рівня

Процес *Authenticate User* отримує дані від сутності *User* у структурі *AuthorizationData*. Далі цей процес передає дані у структурі *AuthorizationData* внутрішньому накопичувачу *Users* та отримує у відповідь статус виконання операції.

Процес *Manage Artists* отримує дані від сутності *Artist* у структурі *ArtistData*. Далі цей процес передає дані у структурі *ArtistData* внутрішньому накопичувачу *Artists* та отримує у відповідь статус виконання операції.

Процес *Manage Restaurants* отримує дані від сутності *Restaurant* у структурі *RestaurantData*. Далі цей процес передає дані у структурі *RestaurantData* внутрішньому накопичувачу *Restaurants* та отримує у відповідь статус виконання операції.

Процес *Manage Ceremony* отримує дані від сутності *Ceremony* у структурі *CeremonyData*, від внутрішнього накопичувача *Users* дані у структурі *UserData*, від

внутрішнього накопичувача Artists дані у структурі ArtistData, від внутрішнього накопичувача Restaurants дані у структурі RestaurantData. Далі цей процес передає дані у структурі UpdatedCeremonyContentData внутрішньому накопичувачу Ceremonies та отримує у відповідь дані у структурі CeremonyContentData.

Процес Manage Payments отримує дані від сутності Bank у структурі BillingData, від внутрішнього накопичувача Users дані у структурі UserData, від внутрішнього накопичувача Artists дані у структурі ArtistData, від внутрішнього накопичувача Restaurants дані у структурі RestaurantData. Далі цей процес передає дані у структурі UserBillingData внутрішньому накопичувачу Payment Bills та отримує у відповідь статус виконання операції.

1.3 Діаграма потоків даних другого рівня для процесу Authenticate User

Діаграму потоків даних другого рівня для процесу Authenticate User зображено на рис.1.3.

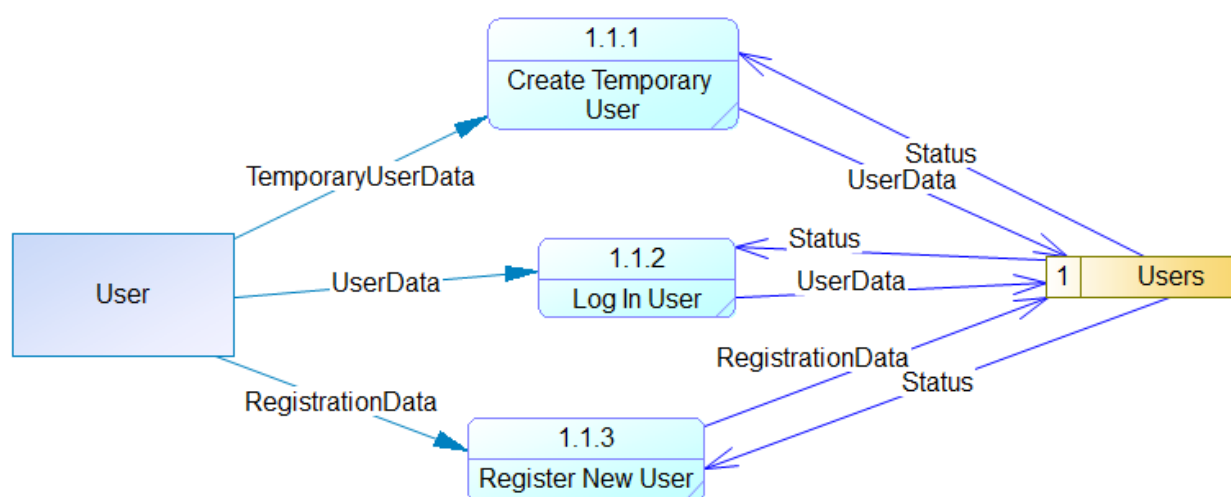


Рисунок 1.3 - Діаграма потоків даних другого рівня для процесу Authenticate User

Процес Create Temporary User отримує від зовнішньої сутності User дані у структурі TemporaryUserData. Далі процес передає дані у структурі UserData до

внутрішнього накопичувача User, де вони зберігаються та користувач отримує роль тимчасового користувача та може продовжити використовувати інформаційну систему у межах його ролі. Процес Create Temporary User отримує у відповідь статус виконання операції.

Процес Log In User отримує від зовнішньої сутності User дані у структурі UserData. Далі процес передає дані у структурі UserData до внутрішнього накопичувача User, у якому перевіряється наявність такого користувача та, у разі наявності, відповідність його пароля. У разі успішної авторизації користувач може продовжувати використовувати інформаційну систему у межах його ролі. Процес Log In User отримує у відповідь статус виконання операції.

Процес Register User отримує від зовнішньої сутності User дані у структурі RegistrationData. Далі процес передає дані у структурі RegistrationData до внутрішнього накопичувача User, у якому перевіряється наявність такого користувача та, у разі не наявності, створюється новий обліковий запис. Процес Register User отримує у відповідь статус виконання операції.

1.4 Діаграма потоків даних другого рівня для процесу Manage Artists

Діаграму потоків даних другого рівня для процесу Manage Artists зображено на рис.1.4.

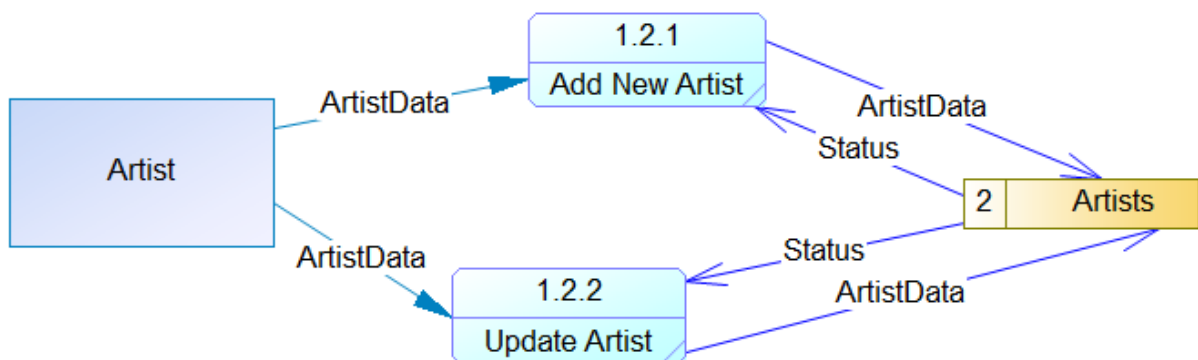


Рисунок 1.4 - Діаграма потоків даних другого рівня для процесу Manage Artists

Процес Add New Artist отримує від зовнішньої сутності Artist дані у структурі ArtistData. Далі процес передає дані у структурі ArtistData до внутрішнього накопичувача Artists, де вони зберігаються для можливої подальшої співпраці у організації церемоній оформлення шлюбу. Процес Add New Artist отримує у відповідь статус виконання операції.

Процес Update Artist отримує від зовнішньої сутності Artist дані у структурі ArtistData. Далі процес передає оновлені дані у структурі ArtistData до внутрішнього накопичувача Artists, де вони зберігаються для можливої подальшої співпраці у організації церемоній оформлення шлюбу або для вказання на тимчасове припинення співпраці із даним артистом. Процес Update Artist отримує у відповідь статус виконання операції.

1.5 Діаграма потоків даних другого рівня для процесу Manage Restaurants

Діаграму потоків даних другого рівня для процесу Manage Restaurants зображено на рис.1.5.

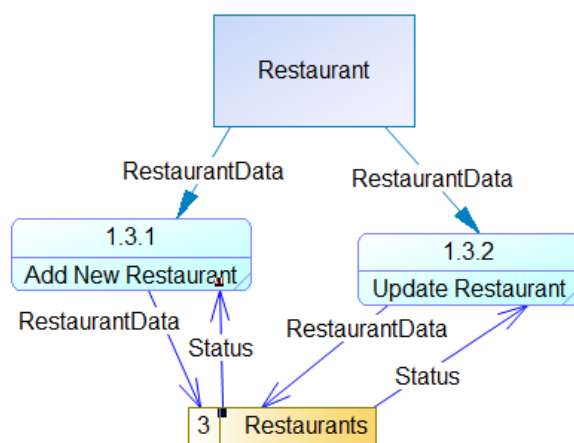


Рисунок 1.5 - Діаграма потоків даних другого рівня для процесу Manage Restaurants

Процес Add New Restaurant отримує від зовнішньої сутності Restaurant дані у структурі RestaurantData. Далі процес передає дані у структурі ArtistData до

внутрішнього накопичувача Restaurants, де вони зберігаються для можливої подальшої співпраці у організації церемоній оформлення шлюбу. Процес Add New Restaurant отримує у відповідь статус виконання операції.

Процес Update New Restaurants отримує від зовнішньої сутності Restaurants дані у структурі RestaurantData. Далі процес передає оновлені дані у структурі RestaurantData до внутрішнього накопичувача Restaurants, де вони зберігаються для можливої подальшої співпраці у організації церемоній оформлення шлюбу або для вказання на тимчасове припинення співпраці із даним артистом. Процес Update New Restaurants отримує у відповідь статус виконання операції.

1.6 Діаграма потоків даних другого рівня для процесу Manage Ceremony

Діаграму потоків даних другого рівня для процесу Manage Ceremony зображено на рис.1.6.

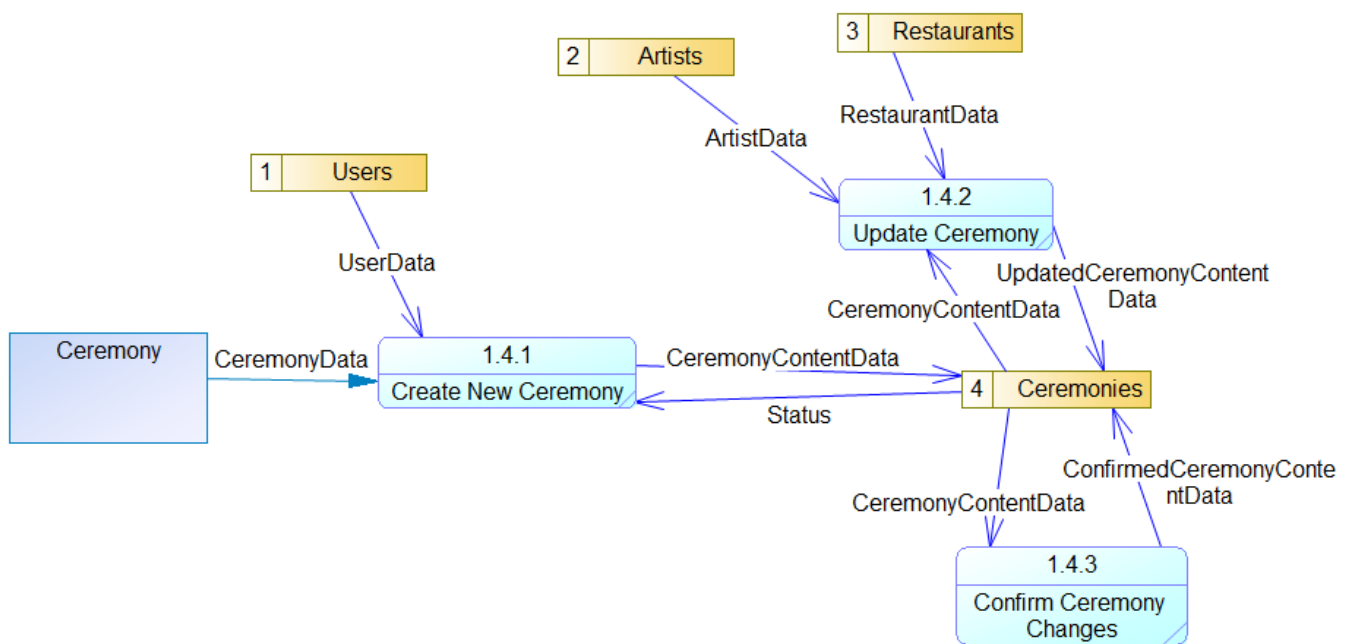


Рисунок 1.6 - Діаграма потоків даних другого рівня для процесу Manage Ceremony

Процес Create New Ceremony отримує від зовнішньої сутності Ceremony дані у структурі CeremonyData, від внутрішнього накопичувача Users дані у структурі UserData. Далі процес передає дані у структурі CeremonyContentData до внутрішнього накопичувача Ceremonies, де вони зберігаються для можливого подальшого редагування церемоній оформлення шлюбу. Процес Create New Ceremony отримує у відповідь статус виконання операції.

Процес Update Ceremony отримує від внутрішнього накопичувача Ceremonies дані у структурі CeremonyContentData, від внутрішнього накопичувача Artists дані у структурі ArtistData, від внутрішнього накопичувача Restaurants дані у структурі RestaurantData. У даному процесі відбувається наповнення церемонії бажаними артистами та місцем проведення або редагування минулих побажань. Процес Update Ceremony повертає накопичувачу Ceremonies дані у структурі UpdatedCeremonyContentData.

Процес Confirm Ceremony отримує від внутрішнього накопичувача Ceremonies дані у структурі CeremonyContentData. У даному процесі відбувається підтвердження та узгодження внесених користувачем змін. Процес Confirm Ceremony повертає накопичувачу Ceremonies дані у структурі ConfirmedCeremonyContentData.

1.7 Діаграма потоків даних другого рівня для процесу Manage Payments

Діаграму потоків даних другого рівня для процесу Manage Payments зображено на рис.1.7.

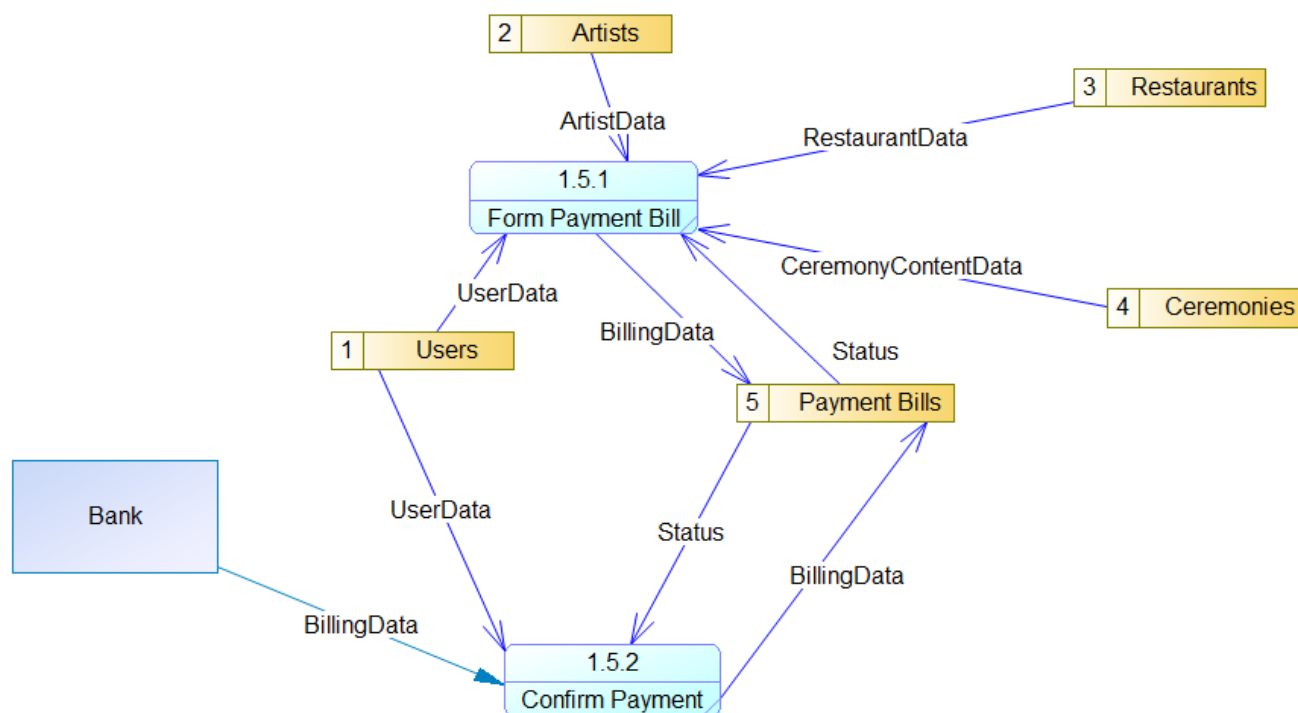


Рисунок 1.7 - Діаграма потоків даних другого рівня для процесу Manage Payments

Процес Form Payment Bill отримує від внутрішнього накопичувача Restaurant дані у структурі RestaurantData, від внутрішнього накопичувача Users дані у структурі UserData, від внутрішнього накопичувача Artists дані у структурі ArtistData, від внутрішнього накопичувача Ceremonies дані у структурі CeremonyContentData. На основі поточного стану церемонії, артистів, що приймають у, ресторану, де відбуватиметься церемонія, формується рахунок. Цей рахунок у структурі BillingData передається до внутрішнього накопичувача Payment Bills. Процес Form Payment Bill отримує у відповідь статус виконання операції.

Процес Confirm Payment отримує від зовнішньої сутності Bank дані у структурі BillingData, від внутрішнього накопичувача Users дані у структурі UserData. У процесі перевіряється сплата рахунку користувачем та ця інформація заносить до накопичувача Payment Bills. Процес Confirm Payment отримує у відповідь статус виконання операції.

1.8 Опис структур даних

Структура `UserData` містить поля `login`, що містить логін користувача; `password`, що містить пароль користувача.

Структура `TemporaryUserData` містить поля `ipAddress`, що містить IP-адресу користувача; `timeEntered`, що містить час входу користувача до інформаційної системи.

Структура `RegistrationData` містить усі поля структури `UserData`; `email`, що містить електронну адресу користувача.

Структура `ArtistData` містить поля `name`, що містить повне ім'я артиста; `genre`, що містить жанр виступів артиста; `email`, що містить електронну адресу артиста; `pricePerHour`, що містить ціну за годину виступу артиста.

Структура `RestaurantData` містить поля `name`, що містить повну назву ресторану; `address`, що містить адресу ресторану; `email`, що містить електронну адресу ресторану; `pricePerHour`, що містить ціну за годину оренди ресторану.

Структура `CeremonyData` містить поля `groomName`, що містить повне ім'я нареченого; `brideName`, що містить повне ім'я нареченої.

Структури `CeremonyContentData`, `UpdatedCeremonyContentData`, `ConfirmedCeremonyContentData` містять поля структури `UserData`, та масиви структур `ArtistData` та `RestaurantData`.

Структура `BillingData` містить поля `billNumber`, що містить номер квитанції; `amount`, що містить суму для оплати.

ВИСНОВКИ

При виконанні даного етапу курсової роботи було створено діаграми потоків даних за стандартом DFD для нульового, першого та другого рівня.

На нульовому та першому рівні наявні п'ять зовнішніх сутностей — User, Artist? Restaurant, Bank та Ceremony, які об'єднані процесом Marriage Registration.

На першому рівні було виділено 5 підпроцесів, що поєднують вказані вище сутності. Також були вказані внутрішні накопичувачі даних. Кожен підпроцес було декомпоновано до другого рівня. Для другого рівня було додано опис логіки роботи процесів.

Також було описано усі структури, у яких передаються дані в інформаційній системі.