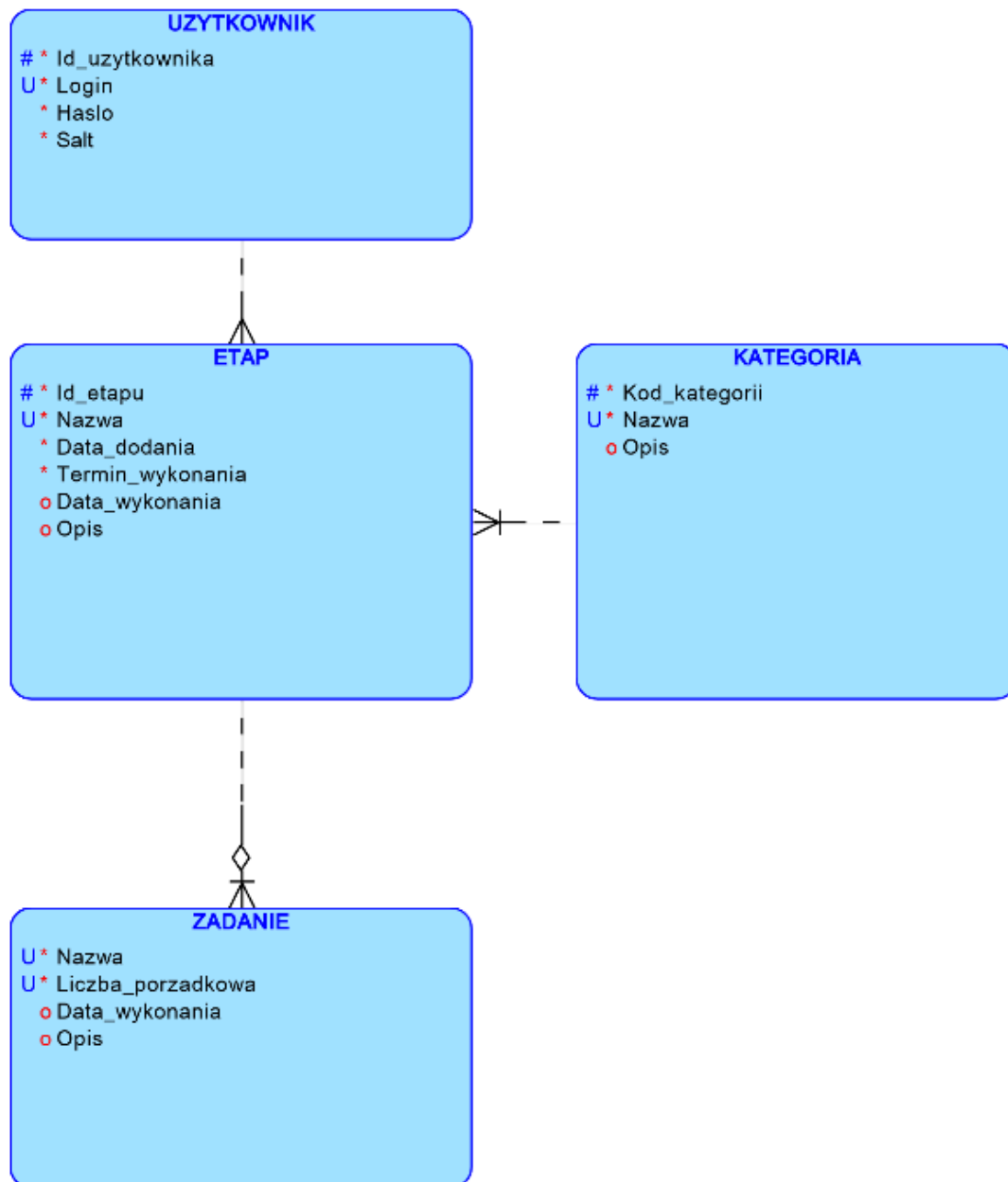


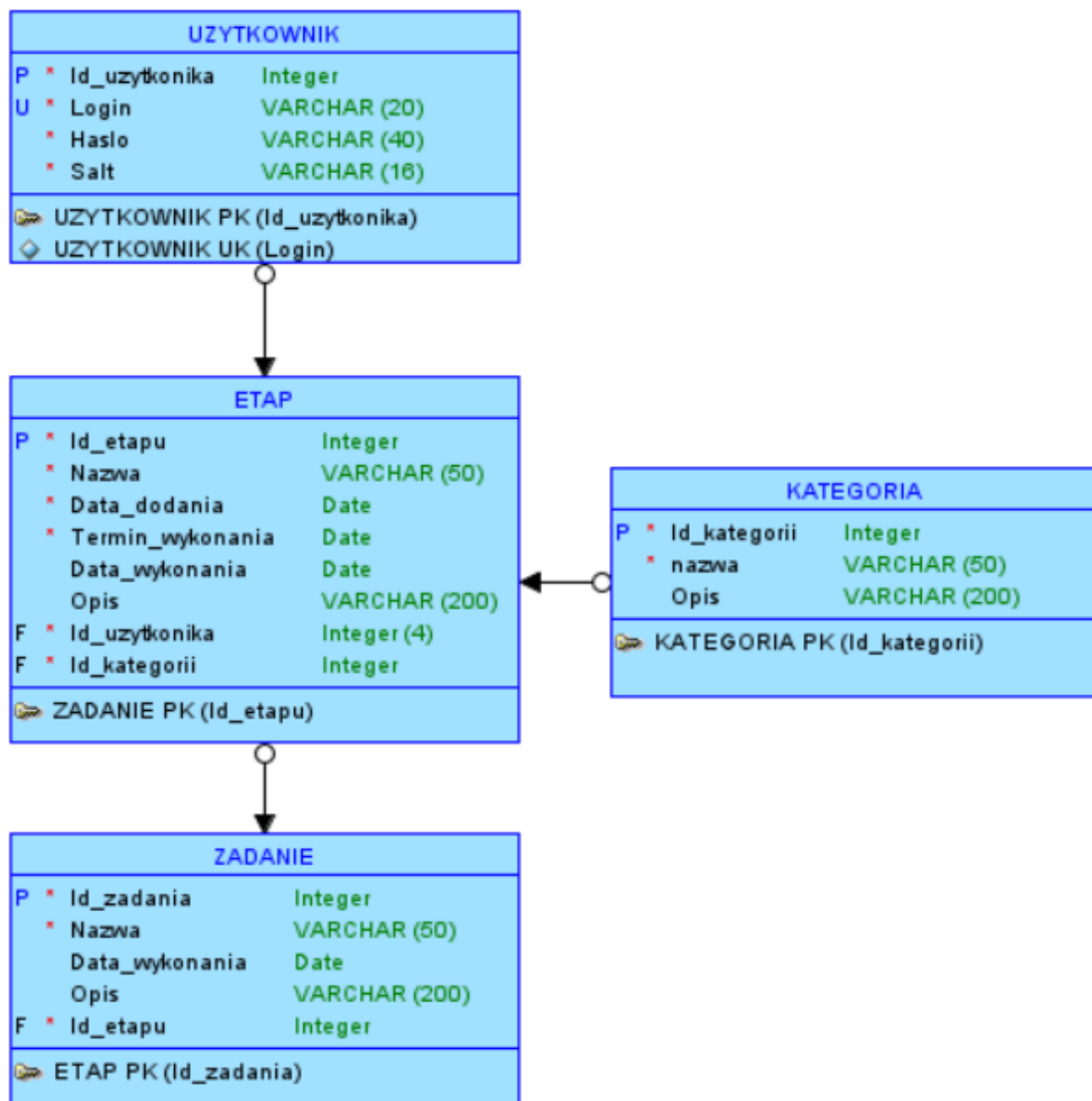
BD2 - zespół 3. - dokumentacja końcowa projektu

1. Modele danych

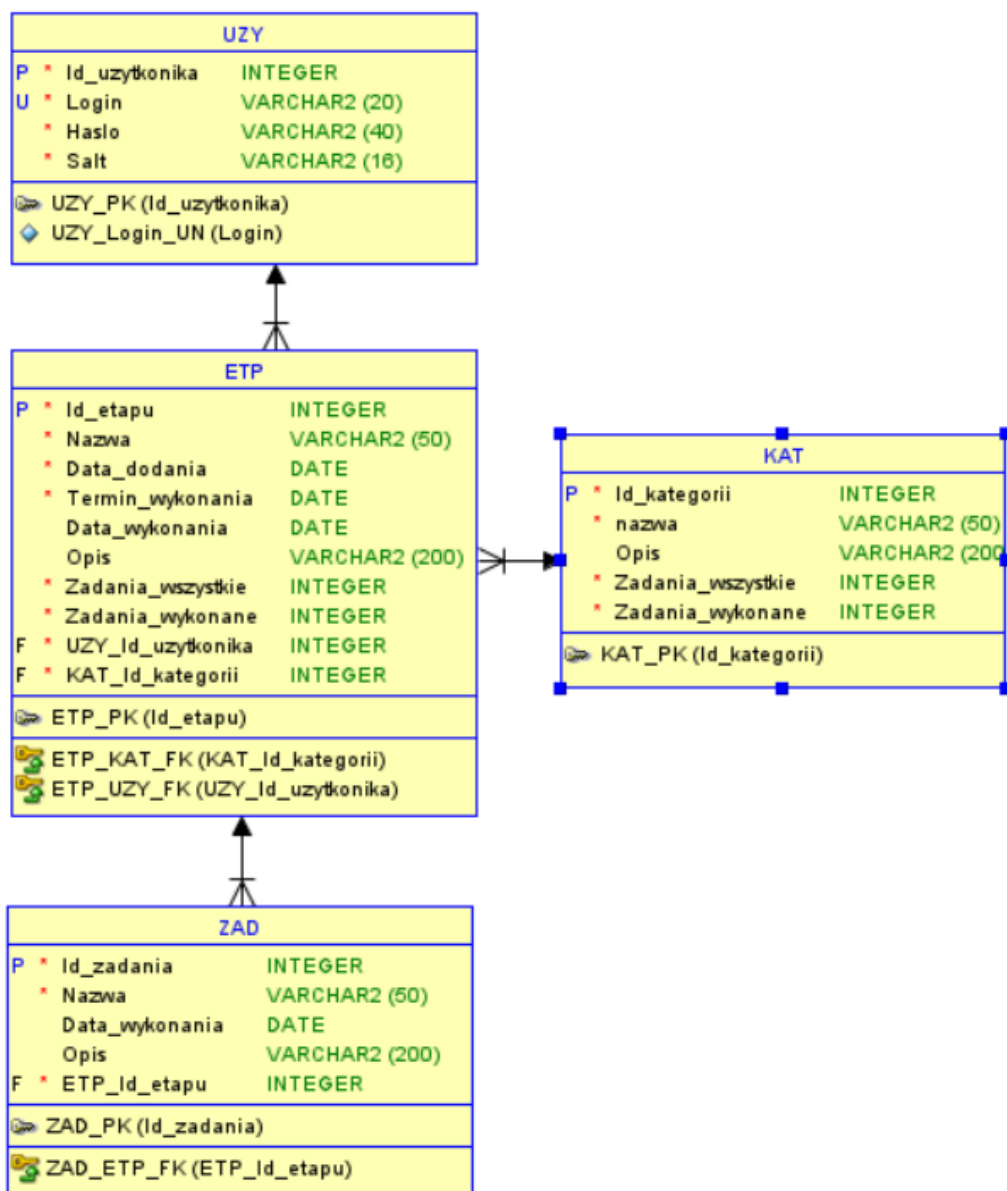
Model E-R



Model logiczny



Model fizyczny



Wykorzystanie dodatkowych mechanizmów baz danych:

- Funkcja, która po podaniu loginu zwraca sól do hasła lub informację o braku użytkownika.
- Funkcje sprawdzające liczbę zadań zaległych, do zrobienia w przeciągu tygodnia oraz w przeciągu miesiąca.
- Procedura do oznaczenia zadania za (nie)wykonane, która automatycznie wstawia datę, modyfikuje kolumny z liczbą wykonanych zadań oraz może zaktualizować stan wykonania całego etapu (zrobione, jeżeli wszystkie podpisane zadania są wykonane).
- Triggery na dodanie i usunięcie zadań, które aktualizują stan wykonania nadrzędnego etapu oraz kolumny agregujące zliczające zadania.
- Procedura do dodawania pustego etapu, która automatycznie wstawia datę dodania i domyślny termin wykonania.
- Funkcje i procedury pomocnicze przy realizacji powyższych.

2. Aplikacja

Założenia dotyczące aplikacji (z dokumentu z wymaganiami)

Aplikacja do śledzenia postępów w nauce - narzędzie dla uczniów i studentów do planowania ich nauki, śledzenia postępów w osiąganiu celów edukacyjnych oraz zarządzania zadaniami i terminami.

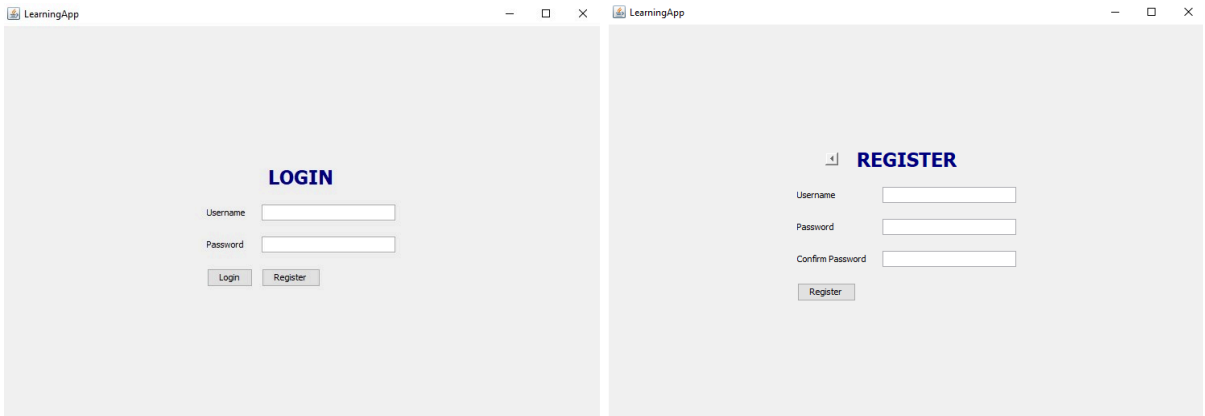
1. Panel logowania:
 - a. System rejestracji – login, hasło.
 - b. System logowania – jak wyżej.
 - c. Implementacja soli do haseł niewidoczna dla użytkownika.
2. Panel zadań:
 - a. Zliczanie zadań zaległych, do zrobienia w przeciągu tygodnia oraz w przeciągu miesiąca.
 - b. Wyświetlanie kalendarium z niezrobionymi zadaniami i ich terminami wykonania.
 - c. Po wybraniu elementu z kalendarium – okno modalne przedstawiające szczegóły dotyczące zadania i umożliwiające oznaczenie go za wykonane.
3. Panel kategorii:
 - a. Dostęp do historii realizowania zadań.
 - b. Lista kategorii, etapów i zadań w formie rozwijanych drzewek z możliwością dodawania nowych elementów (tylko podanie nazwy, szczegóły dodaje się później).
 - c. Wyświetlanie liczby wykonanych i wszystkich zadań przy każdej kategorii i etapie.
 - d. Wyświetlanie statusu wykonania przy każdym zadaniu.
 - e. Po wybraniu elementu – okno modalne, w którym można modyfikować szczegóły lub usunąć element.

Realizacja wymagań

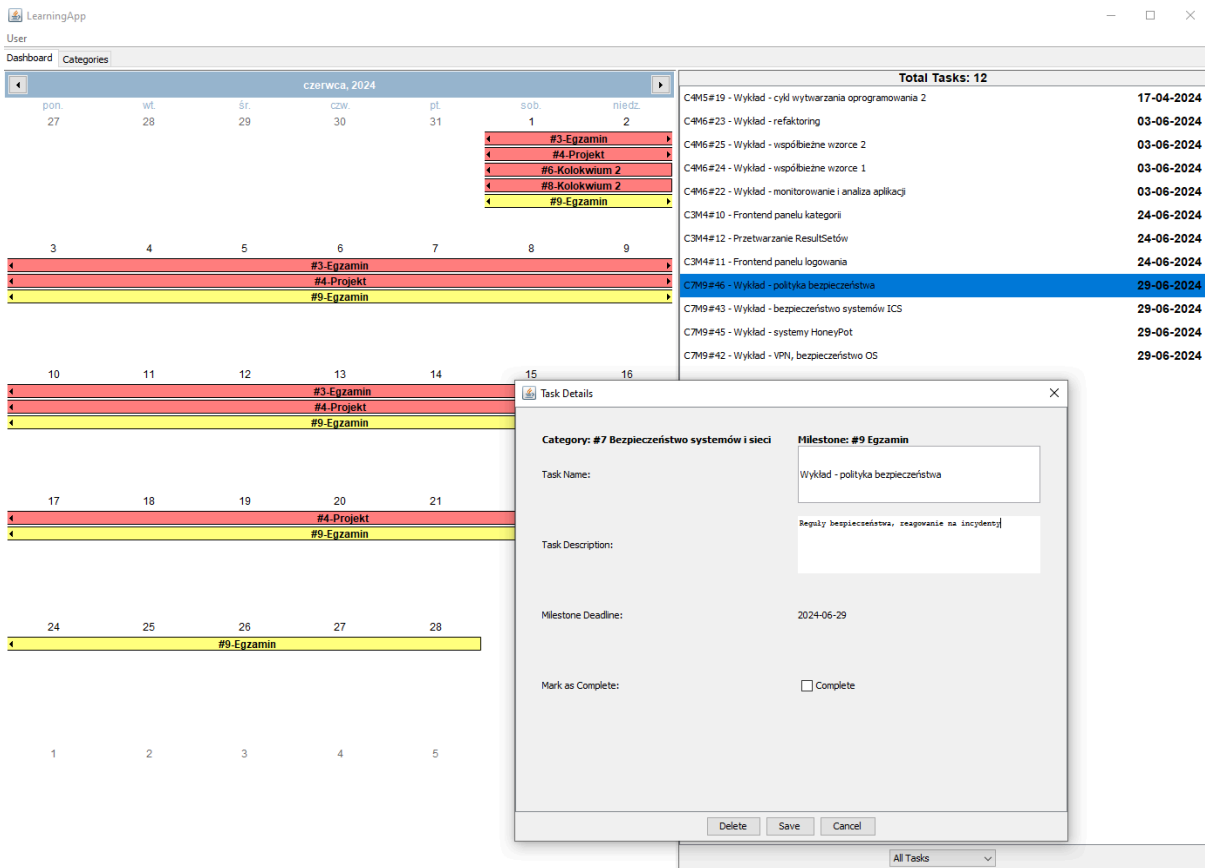
Aplikacja została wykonana w Javie i Swingu z wykorzystaniem Mavena, JDBC i JUnit. Wybrana baza danych to Oracle.

Do eksplorowania aplikacji można skorzystać z przygotowanych danych testowych - w pliku `config.properties` należy ustawić wartość `databaseName=learning_app_db_test`, a do logowania użyć loginu `user3` i hasła `user3`.

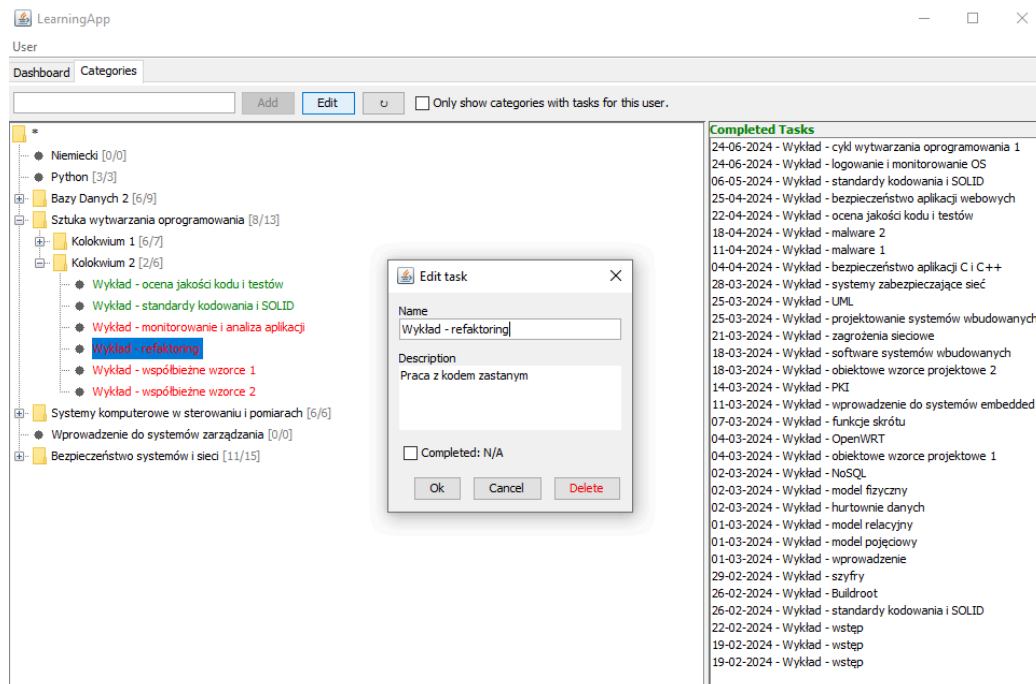
Po otwarciu aplikacji użytkownik widzi ekran logowania - może zalogować się lub utworzyć nowe konto.



Po zalogowaniu użytkownik widzi dashboard, na którym znajduje się kalendarz z zaznaczonymi etapami oraz lista niewykonanych zadań z ich terminami. Jest kilka filtrów do wyboru. Nad listą widoczna jest liczba zadań do zrobienia. Po wybraniu zadania z listy pojawia się okno modalne, gdzie można edytować dane o zadaniu.



Użytkownik ma również dostęp do panelu kategorii - dostępna jest historia wykonanych zadań oraz drzewka kategorii-etapów-zadań. Stan każdego zadania jest oznaczony kolorem, a kategorie i etapy zliczają wszystkie i ukończone zadania. Każdy element można edytować i usuwać lub dodawać nowe.



3. Instalacja - Windows

Uwaga: nazwy baz danych oraz loginu i hasła do nich można dostosować - w takim przypadku należy podmienić wartości w pliku `config.properties`. Hasło należy zacząć literą i unikać znaku `%`.

Pobrać i zainstalować Javę w wersji co najmniej 21:
<https://www.oracle.com/java/technologies/downloads/>

Pobrać i zainstalować bazę danych Oracle (należy zapamiętać główne hasło przy wyborze):
<https://www.oracle.com/database/technologies/oracle-database-software-downloads.html>
Instalacja była testowana na Oracle Database 21c XE i EE. W przypadku instalacji na EE należy pozostawić domyślne opcje.

Uruchomić plik `setup.bat` - skrypt poprosi o hasło. Baza danych powinna być gotowa do pracy. Jeżeli zajdzie potrzeba jej usunięcia, można skorzystać ze skryptu `drop.bat`.

Aby skompilować aplikację, można skorzystać z IDE (np. IntelliJ IDEA Community Edition: <https://www.jetbrains.com/idea/download/?fromIDE=§ion=windows>) i znaleźć opcję "mvn package". Można też samodzielnie zainstalować Mavena i w folderze z `pom.xml` wykonać `mvn package`. W folderze `target` pojawi się plik z aplikacją gotową do uruchomienia: `LearningApp-1.0-jar-with-dependencies.jar`.