
PROJEKT ZESPOŁOWY

Inteligentny zamek

Autorzy:

MACIEJ MARCINIAK

nr indeksu: 121996

e mail:

maciej.r.marcniak@student.put.poznan.pl

DAMIAN FILIPOWICZ

nr indeksu: 122002

e mail:

Damian.Filipowicz@student.put.poznan.pl

Spis treści

1	Aktorzy systemu	4
2	Opis składowych systemu	5
2.1	Urządzenie sterujące	5
2.2	Aplikacja mobilna	7
2.3	Aplikacja serwerowa obsługującego bazę danych	9
3	Diagram przypadków użycia	10
4	Projekt bazy danych	12
5	Zabezpieczenia systemu	15
6	Widok graficzny systemu	16
6.1	Panel logowania użytkownika	16
6.2	Panel rejestracji użytkownika	17
6.3	Panel listy zamków	17
6.4	Panel boczny	18
6.5	Panel zarządzania certyfikatami	18
6.6	Panel listy certyfikatów	19
6.7	Panel certyfikatu	19
6.8	Panel wnioskowania o certyfikat	20
6.9	Panel tworzenia certyfikatu dla gościa	20
6.10	Panel administratora	21
6.11	Panel historii użycia zamków	21
6.12	Panel generowania nowego certyfikatu (administrator)	22
6.13	Panel zarządzania certyfikatami (administrator)	23
6.14	Panel listy oczekujących użytkowników do rejestracji	23
6.15	Panel listy oczekujących certyfikatów do wygenerowania	24
6.16	Panel ustawień konta	24

Wstęp

Inteligentny zamek powinien być systemem, który ma na celu zastąpienie starego modelu zabezpieczeń różnego rodzaju drzwi i skrytek w którym używano tradycyjnych kluczy, czy szyfrów na klucze cyfrowe, którymi będzie można posługiwać się przy pomocy smartfonów z funkcją bluetooth. Celem tego rodzaju usprawnień będzie wyeliminowanie z życia codziennego sytuacji w których użytkownik musi posiadać pęk kluczy. Zamiast tego dzięki temu systemowi może wszystkie klucze przechowywać w jednym miejscu (smartfonie).

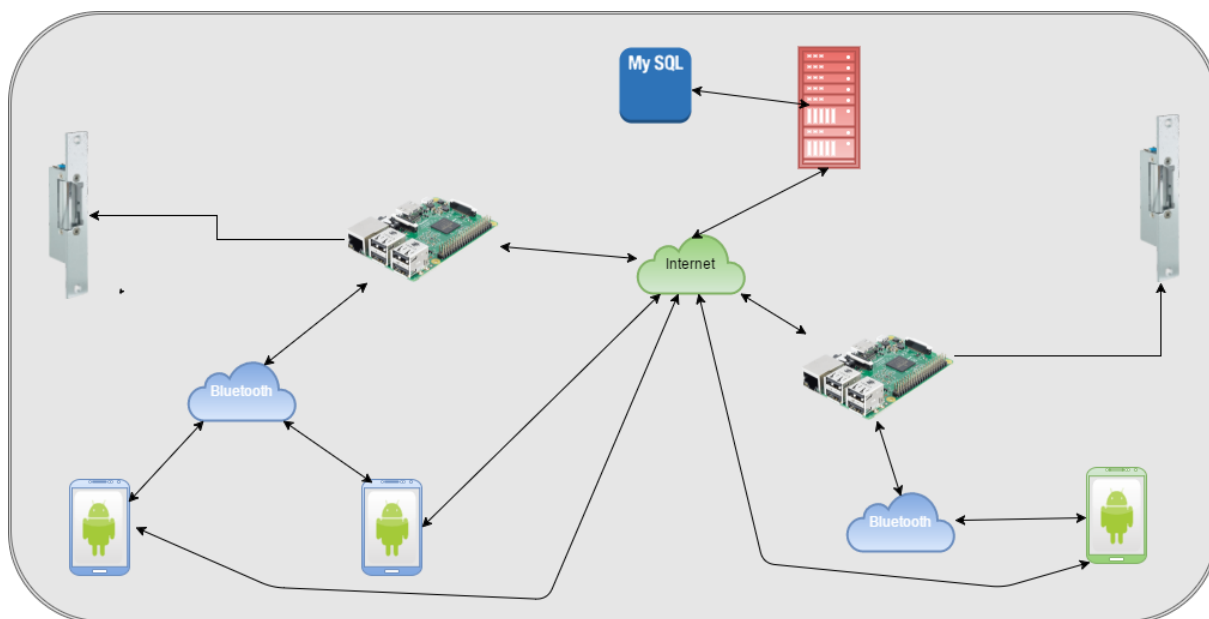
System składać się będzie z:

- urządzenia sterującego:
 - mikrokomputera Raspberry Pi 3,
 - serwomechanizmu / elektronicznego zamka,
- aplikacji mobilnej,
- aplikacji serwerowej obsługującej bazę danych.

System będzie spełniał wymagania dotyczące bezpieczeństwa poprzez zastosowanie szeregu funkcji kryptograficznych przy procesie uwierzytelniania jak i przy generowaniu kluczy takich jak np. funkcje skrótu, SSH, algorytmów szyfrowania asymetrycznego, systemu zarządzaniem kluczem publicznym (podpisu cyfrowego).

Używane klucze będą posiadały podpis cyfrowy, który jednoznacznie będzie definiował właściciela oraz stempel czasowy do określania ważności. Klucze będą mogły mieć w zależności od przeznaczenia różne okresy przedawnienia, np. właściciel mieszkający w danym domu posiadać będzie klucz o długim terminie ważności, a goście klucz jednorazowy bądź kilku godzinny bez możliwości odnowienia. Wszelkie dane dostępowe będą generowane i dystrybuowane na serwerze systemu, z możliwością zdalnej prośby o utworzenie kluczy tylko przez uprawnione przez administratora osoby.

Ogólny schemat systemu znajduje się na Rysunku 1.



Rysunek 1: Diagram wdrożeń

Rozdział 1

Aktorzy systemu

W systemie Inteligentnego zamka wyróżniamy następujących aktorów:

- **RaspberryPi** - jest to mikrokomputer Raspberry Pi 3 sterujący zamkiem,
- **serwomechanizm/elektrozamek** - jest to urządzenie służące do odblokowania/zablokowania zamka,
- **urządzenie mobilne** - jest to urządzenie posiadające system operacyjny, z funkcją bluetooth oraz posiadające możliwość instalacji aplikacji,
- **aplikacja serwerowa** - jest to program znajdujący się na serwerze z dostępem globalnym poprzez Internet,
- **użytkownik** - jest to osoba fizyczna operująca urządzeniem mobilnym, chcąc uzyskać dostęp do zamka.

Użytkowników dodatkowo dzieli się na grupy ze względu na uprawnienia w systemie:

- **gość** - posiada najniższe uprawnienia, może jedynie posiadać klucze o krótkim okresie ważności, nie może generować nowych kluczy ani udostępniać ich,
- **użytkownik zalogowany** - posiada uprawnienia gościa, dodatkowo przechowywać może klucze o stałym dostępie do zamka (np. dostęp przez cały dzień),
- **administrator** - może wykonywać wszystkie czynności związane z uprawnieniami gościa i użytkownika zalogowanego, dodatkowo posiada dostęp do statystyk historii zamka, decyduje o rejestracji użytkowników zalogowanych.

Rozdział 2

Opis składowych systemu

2.1 Urządzenie sterujące

Zadaniem urządzenia sterującego, w którego skład wchodzić będą Raspberry Pi 3 oraz serwomechanizm/zamka elektronicznego jest weryfikacja klucza cyfrowego przesyłanego przez urządzenie mobilne oraz otwieranie zamka przy pozytywnym wyniku weryfikacji.

Oprogramowanie mikrokomputera obejmuje system Linux raspbian-jessie oraz szereg podprogramów napisanych w języku Python. Skrypty programów łączą się do serwera w celu pobrania informacji o poprawności i daty ważności certyfikatu dostępu. Jeśli dane będą poprawne to zostaje wysterowany serwomechanizm, który otwiera zamek, w przeciwnym przypadku użytkownik zostanie poinformowany o odmowie dostępu, a nieudana próba dostania się do systemu zarejestrowana zostanie w bazie danych wraz z danymi właściciela klucza. Funkcjonalność urządzenia sterującego przedstawiona została w Tabeli 2.1.

Tabela 2.1: Tabela wymagań funkcjonalnych urządzenia sterującego

Funkcja	Opis	Aktorzy
Parowanie urządzeń bluetooth	Parowanie bluetooth urządzenia mobilnego z Raspberry Pi	RaspberryPi, Urządzenie mobilne
Nasłuchiwanie połączenia bluetooth	Oczekuje na przychodzące połączenia bluetooth	RaspberryPi
Nawiązanie połączenia bluetooth	Każda próba nawiązania połączenia bluetooth zostanie zaakceptowana	RaspberryPi, Urządzenie mobilne
Pobranie pliku z kluczem cyfrowym przez bluetooth	Przesyłanie pliku z kluczem dostępowym z urządzenia mobilnego do Raspberry Pi poprzez bluetooth	RaspberryPi, Urządzenie mobilne, Gość
Weryfikacja poprawności klucza cyfrowego	Wysłanie zapytania w języku SQL do bazy danych	RaspberryPi, Aplikacja serwerowa
Otwarcie zamka	Otwarcie zamka poprzez wysłanie sygnału PWM do serwomechanizmu lub zezwolenie zamka elektronicznego	RaspberryPi, Serwomechanizm/ elektrozamek

Zamknięcie zamka	Zamknięcie zamka poprzez wysłanie sygnału PWM do serwomechanizmu lub zezwolenie zamka elektronicznego	RaspberryPi, Serwomechanizm/elektrozamek
Rejestracja próby dostępu	Zapis każdej pozytywnej i negatywnej próby weryfikacji klucza cyfrowego w tabeli bazy danych	RaspberryPi
Deszyfracja certyfikatu użytkownika	Deszyfracja pliku z certyfikatem dostępowym używając klucza publicznego użytkownika	RaspberryPi
Pobranie awaryjnego klucza dostępu	Umożliwia wczytanie specjalnego hasła 512-bitowego do Raspberry Pi, który otwiera zamek bez konieczności dostępu do Internetu	RaspberryPi, Aplikacja mobilna, Administrator

Wymagania pozafunkcjonalne:

- jednocześnie może być weryfikowany tylko jeden użytkownik,
- zasięg połączenia bluetooth to maksymalnie 15m,
- niezbędny dostęp do Internetu do połączenia z aplikacją serwerową przy weryfikacji kluczy,
- narzut czasowy związany z weryfikacją poprawności klucza cyfrowego zależny od parametrów serwera i sieci,
- niezbędny ciągły dostęp do zasilania 5V o prądzie co najmniej 2.5A,
- ograniczenia prądowe dla serwomechanizmu lub zamka elektronicznego,
- narzut czasowy związany z uruchomieniem urządzenia - maksymalnie 20 sekund.

2.2 Aplikacja mobilna

Aplikacja mobilna w języku Java na platformę Android ma na celu przechowywanie w pamięci smartfona klucze cyfrowe użytkownika oraz możliwość komunikacji z człowiek-zamek-serwer. Program posiadać powinien interfejs graficzny, dzięki któremu będzie można wybrać, który zamek chce się otworzyć w danej chwili. Klucz cyfrowy przesyłany będzie bezprzewodowo do komputera sterującego zamkiem za pomocą sieci bluetooth. Aplikacja powinna posiadać również funkcję generowania kluczy tymczasowych, które można udostępniać osobom postronnym z ustalonym okresem ważności (jednorazowy, godzinny, od poniedziałku do piątku w godzinach od 8 do 16 itp.). W tym celu zostaje wysłana prośba do serwera poprzez Internet o wygenerowanie klucza o określonych parametrach. Funkcjonalność aplikacji przedstawiona została w Tabeli 2.2.

Tabela 2.2: Tabela wymagań funkcjonalnych aplikacji mobilnej

Funkcja	Opis	Aktorzy
Parowanie urządzeń bluetooth	Parowanie bluetooth urządzenia mobilnego z Raspberry Pi	Urządzenie mobilne, RaspberryPi
Nawiązywanie połączenia bluetooth	Nawiązanie połączenia bluetooth z konkretnym zamkiem identyfikując go jednoznacznie adresem MAC	Urządzenie mobilne, RaspberryPi
Przesłanie pliku klucza cyfrowego	Przesłanie pliku zawierającego klucz cyfrowy do urządzenia sterującego zamkiem. Komputer sterujący odsyła wynik weryfikacji (pozytywny lub negatywny)	Urządzenie mobilne, RaspberryPi
Utworzenie klucza cyfrowego dla gości	Utworzenie specjalnego klucza cyfrowego o ograniczonym dostępie oraz krótkim terminie ważności do użytku dla gości. Każde żądanie generowania klucza wymaga wpisania klucza bezpieczeństwa	Urządzenie mobilne, Aplikacja serwerowa, Użytkownik zalogowany
Udostępnianie klucza cyfrowego dla gości	Udostępnienie specjalnego klucza cyfrowego o ograniczonym dostępie oraz krótkim terminie ważności poprzez np. wiadomość MMS, bluetooth	Urządzenie mobilne, Użytkownik zalogowany
Wczytanie klucza cyfrowego z pliku	Umożliwia wczytanie do listy dostępnych zamków pliku klucza cyfrowego	Urządzenie mobilne, Gość
Pobranie z serwera nowego klucza cyfrowego	Umożliwia pobranie z serwera klucza cyfrowego i dodanie go do listy dostępnych zamków	Urządzenie mobilne, Aplikacja serwerowa, Użytkownik zalogowany
Prośba o przedłużenie ważności klucza	W celu przedłużenia ważności certyfikatu zostaje wysłana prośba do administratora systemu	Urządzenie mobilne, Aplikacja serwerowa, Użytkownik zalogowany
Listowanie dostępnych kluczy	Wyświetlenie na ekranie telefonu listy dostępnych kluczy do danych drzwi	Urządzenie mobilne, Gość

Modyfikacja danych kluczy cyfrowych	Modyfikacja nazw użytkownika, zamków. Pozwala spersonalizować opis zamków	Urządzenie mobilne, Aplikacja serwerowa, Użytkownik zalogowany
Szyfrowanie pliku klucza cyfrowego	Szyfrowanie algorytmem RSA klucza cyfrowego z wykorzystaniem klucza prywatnego	Urządzenie mobilne
Przechowywanie kluczy cyfrowych	Przechowywanie kluczy cyfrowych (szyfrowanych) w pamięci telefonu	Urządzenie mobilne
Podgląd do historii akcji zamków	Umożliwia przeglądanie historii akcji zamka, tzn. daty otwarcia przez kogo, daty zamknięcia	Urządzenie mobilne, Aplikacja serwerowa, Administrator
Autoryzacja użytkownika do aplikacji	Logowanie użytkownika poprzez podanie hasła i loginu do odblokowania aplikacji	Urządzenie mobilne, Aplikacja serwerowa, Użytkownik zalogowany
Rejestracja użytkownika	Założenie nowego konta użytkownika w systemie	Urządzenie mobilne, Aplikacja serwerowa, Gość
Akceptacja przez administratora nowego użytkownika	Administrator systemu może zaakceptować i nadać uprawnienia użytkownika	Urządzenie mobilne, Aplikacja serwerowa, Administrator
Zarządzanie ważnością certyfikatów dostępu	Dodawanie, usuwanie ważności certyfikatów dostępowych. Usunięcie praw użytkownika nie skutkuje unieważnieniem wygenerowanych przez niego certyfikatów	Urządzenie mobilne, Aplikacja serwerowa, Administrator
Przesłanie awaryjnego klucza dostępu	Umożliwia wczytanie specjalnego hasła 512-bitowego do Raspberry Pi, który otwiera zamek bez konieczności dostępu do Internetu	Urządzenie mobilne, RaspberryPi, Administrator
Tryb otwierania zamka	Komunikacja z Raspberry może odbywać się automatycznie lub na żądanie wyzwalane przyciskiem otwierania zamka z poziomu aplikacji	Urządzenie mobilne, RaspberryPi, Użytkownik zalogowany

Wymagania pozafunkcjonalne:

- narzut czasowy związany z procesem szyfrowania kluczy cyfrowych (zależny od parametrów urządzenia mobilnego),
- zabezpieczenie transmisji danych poprzez szyfrowanie przy pomocy asymetrycznych kluczy cyfrowych,
- wymagany dostęp do Internetu do zarządzania kluczami, czy logowania,
- przyznanie uprawnień aplikacji do modułu bluetooth, wysyłania wiadomości MMS, Internetu,
- język aplikacji Polski,
- wersja androida minimalna 4.4, docelowa 5.0.

2.3 Aplikacja serwerowa obsługująca bazę danych

Rolą serwera w tym systemie będzie przechowywanie danych dostępowych w bazie danych MySQL oraz generowanie nowych kluczy cyfrowych poprzez program w języku Python. Aplikacja serwerowa oparta powinna być o technologię Python oraz serwera http Nginx. Serwer postawiony powinien być na odrębnym urządzeniu od instalacji zamka, lecz dopuszcza się ze względów ekonomicznych również postawienie serwera na wybranym (jeśli w systemie znajduje się wiele zamków) urządzeniu Raspberry Pi. Funkcjonalność aplikacji serwerowej i bazy danych przedstawiona została w Tabeli 2.3.

Tabela 2.3: Tabela wymagań funkcjonalnych aplikacji serwerowej obsługującej bazę danych

Funkcja	Opis	Aktorzy
Utworzenie klucza cyfrowego na żądanie	Utworzenie pseudolosowego 128-bitowego klucza cyfrowego	Aplikacja serwerowa, Użytkownik zalogowany
Kontrola uprawnień użytkownika	Weryfikacja uprawnień użytkownika do wykonania danej czynności	Aplikacja serwerowa
Modyfikowanie wpisów w bazie danych	Pośredniczenie w modyfikacji danych zawartych w bazie danych	Aplikacja serwerowa
Przekazywanie wpisów z bazy danych	Pośredniczenie w przekazywaniu danych pobieranych z bazy danych	Aplikacja serwerowa
Rejestrowanie żądań dostępu	Zapisywanie danych użytkownika ubiegającego się o dostęp do serwera	Aplikacja serwerowa
Pobranie historii dostępu zamka	Pobranie statystyk związanych z historią dostępu do zamka	Aplikacja serwerowa, Administrator
Zablokowanie dostępu użytkownika	Zablokowanie certyfikatu dostępowego, np. w przypadku kradzieży telefonu	Aplikacja serwerowa, Administrator

Wymagania niefunkcjonalne:

- ograniczenie pamięci dostępnej dla bazy danych (32Gb - pamięć niezbędna dla systemu operacyjnego i oprogramowania),
- ograniczenie liczby obsługiwanych zamków zależna od wielkości dostępnej pamięci i liczby użytkowników,
- narzut czasowy związany z generowaniem nowych kluczy,
- ograniczenie liczby użytkowników wykonujących jednocześnie żądania do serwera - 9 urządzeń,
- wymagany system operacyjny Linux dedykowany pod Raspberry,
- dostęp do Internetu do połączenia z zamkami i urządzeniami mobilnymi,
- zabezpieczenie bazy danych hasłem generowanym losowo.

Rozdział 3

Diagram przypadków użycia

Diagram przypadków użycia (funkcjonalności) systemu wraz z opowiadającymi aktorami przedstawiono na Rysunku 3.1.

Rysunek 3.1: Diagram przypadków użycia

Rozdział 4

Projekt bazy danych

Baza danych przechowywać będzie składać się z pięciu tabel:

- **USERS** - przechowuje dane użytkowników oraz dane niezbędne przy weryfikacji logowania,
- **LOCKS** - zawiera informacje na temat dostępnych w systemie zamków,
- **ACCESS_TO_LOCKS** - archiwizuje próby użycia certyfikatów,
- **LOCKS_KEYS** - zawiera wszystkie klucze dostępowe użytkowników.

Wiersz tabeli USERS zawierać musi:

- **ID_USER** - unikalny identyfikator (klucz główny) użytkownika składający się z 10 cyfr,
- **LOGIN** - unikalna nazwa użytkownika niezbędna podczas logowania, zawierająca nie więcej niż 255 znaków,
- **PASSWORD** - hasło zapisane w postaci skrótu, potrzebne do autoryzacji dostępu użytkownikowi,
- **NAME** - imię użytkownika,
- **SURNAME** - nazwisko użytkownika,
- **IS_ADMIN** - pole boolowskie wskazujące czy dany użytkownik jest administratorem czy nie.

Zamek opisywany jest poprzez kolumny:

- **ID_LOCK** - unikalny identyfikator (klucz główny) zamka składający się z 10 cyfr,
- **NAME** - unikalna nazwa zamka,
- **LOCALIZATION** - nieobowiązkowe pole opisujące fizyczne położenie zamka.

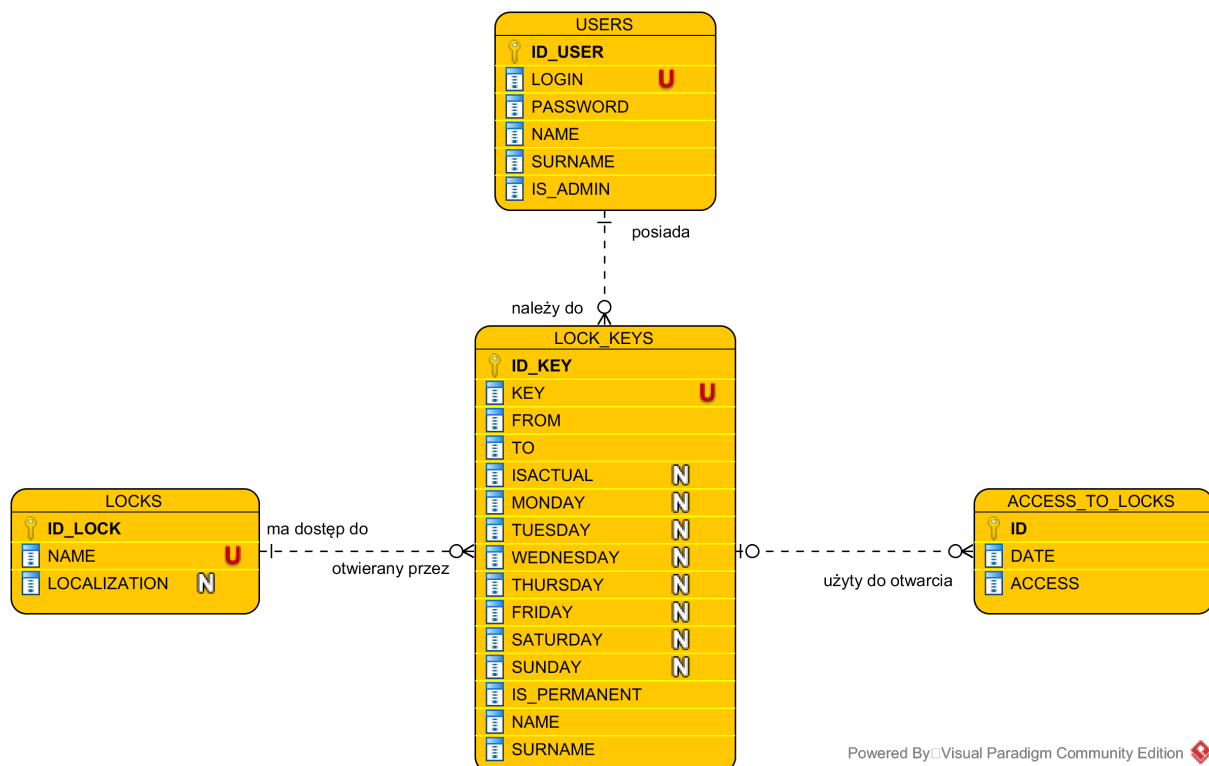
Klucz dostępowy składa się z:

- **ID_KEY** - unikalny identyfikator (klucz główny) klucza dostępowego składający się z 10 cyfr,
- **ID_LOCK** - klucz obcy do tabeli przechowującej dostępne zamki,
- **ID_USER** - klucz obcy do tabeli przechowującej dane użytkownika, jest to pole służące do określenia kto utworzył klucz dostępu,
- **KEY** - unikalna wartość certyfikatu dostępu,
- **FROM** - data od której obowiązuje klucz,
- **TO** - data do której obowiązuje klucz,
- **ISACTUAL** - data wygaśnięcia klucza, jeśli równa TO, oznacza to że klucz utracił ważność z powodu czasu, jeśli różna oznacza, to że zablokowano z innego powodu ważność,
- **MONDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w poniedziałki,
- **TUESDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp we wtorki,
- **WEDNESDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w środy,
- **THURSDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w czwartki,
- **FRIDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w piątki,
- **SATURDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w soboty,
- **SUNDAY** - słowne określenie, w których godzinach zostanie przyznany dostęp w niedziele,
- **IS_PERNAMENT** - zmienna boolowska oznaczająca czy dostęp jest zawsze,
- **NAME** - imię osoby, której dotyczy certyfikat,
- **SURNAME** - nazwisko osoby, której dotyczy certyfikat.

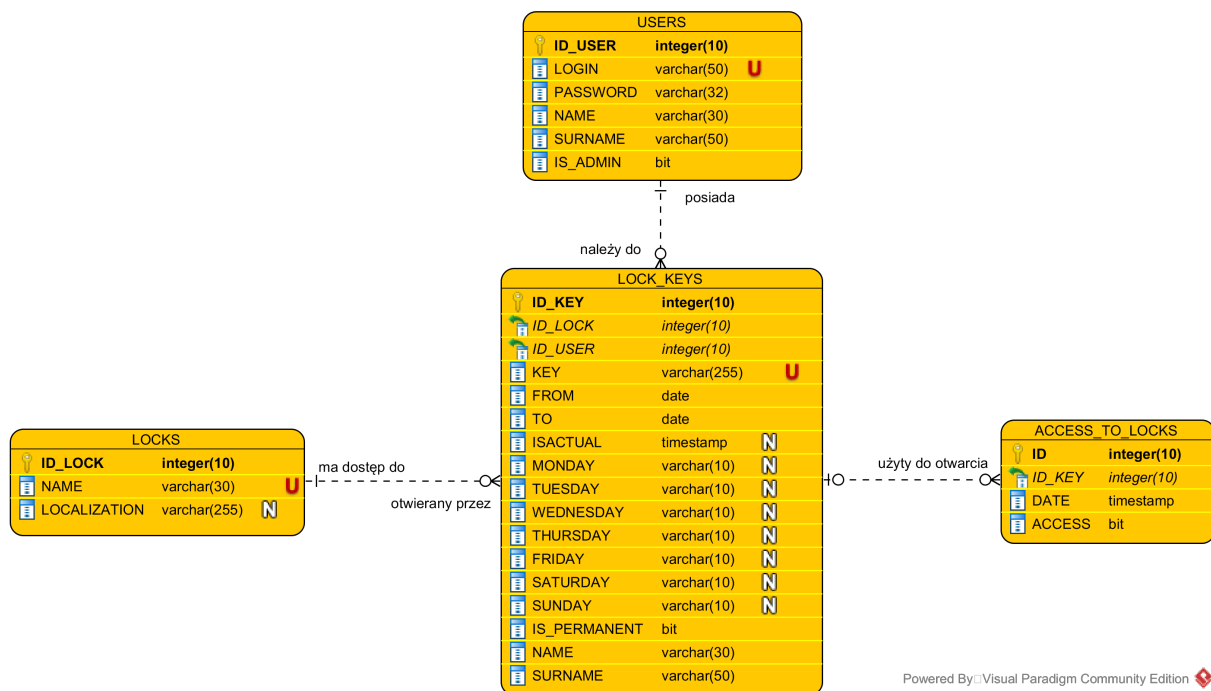
W tabeli archiwizującej akcje na zamku znajdują się takie dane jak:

- **ID** - unikalny identyfikator (klucz główny) akcji wykonanej na certyfikacie składający się z 10 cyfr,
- **ID_KEY** - klucz obcy do tabeli przechowującej klucze dostępne, dzięki tej informacji możemy uzyskać dane o zamku, który został otwierany jak również do kogo należał klucz,
- **DATE** - dokładna data z godziną użycia klucza dostępowego,
- **ACCESS** - binarna flaga informująca czy dostęp został przyznany czy odmówiony.

Diagramy bazy danych odpowiednio encji i relacji przedstawione zostały na Rysunkach 4.1 i 4.2.



Rysunek 4.1: Diagram encji bazy danych



Rysunek 4.2: Diagram relacji bazy danych

Rozdział 5

Zabezpieczenia systemu

Inteligentny zamek ma na celu zabezpieczanie wejścia do pomieszczeń, dlatego sam system powinien być również bezpieczny. Podstawowymi elementami jakie należy chronić są:

- hasła,
- dane użytkowników,
- ważniejsze funkcje systemu (tj. generowanie nowych kluczy),
- transmisję danych,
- zasilanie urządzenia,
- dostępność do instalacji.

Hasła przechowywane powinny być w bazie danych w postaci skrótu funkcji SHA-3. Każde hasło składać powinno się z przynajmniej 6 znaków i zawierać co najmniej jedną cyfrę.

Ogólne dane użytkowników znajdować się będą w bazie danych, do której jedynie bezpośredni dostęp będzie z aplikacji serwera lub z konta administratora. Hasło dostępu generowane powinny być za pomocą generatora liczb pseudolosowych i być o długości co najmniej 400bitów oraz przechodzić podstawowe testy losowości.

Zabezpieczeniem ważniejszych (wrażliwych) funkcji systemu odbywać się będzie po stronie aplikacji serwerowej. Serwer będzie zezwalał tylko na operacje dozwolone dla danego typu użytkownika. Do wrażliwych funkcji należy przede wszystkim pobieranie danych z bazy danych oraz ich modyfikacje. Inną formą zabezpieczenia operacji będzie podawanie hasła, mimo iż użytkownik będzie zalogowany. Weryfikacja hasła ma na celu ochronę przed chwilowym przejęciem telefonu i wygenerowaniem np. sobie dostępu do zamka.

Każde połączenie internetowe pomiędzy mikrokomputerem Raspberry Pi i serwerem będzie zabezpieczone protokołem IPSec. Komunikacja bluetooth nie będzie szyfrowana, lecz certyfikaty przesyłane będą podpisane cyfrowo kluczem asymetrycznym RSA. Pary kluczy (prywatny i publiczny) będą przypisane do danego certyfikatu i tracące ważność wraz z nim.

Zabezpieczeniem fizycznym systemu przed utratą zasilania z sieci energetycznej, będzie zamontowanie równoległe zasilania bateryjnego załączanego w momencie zaniku głównego. Ochroną systemu przed ingerencją w sprzęt będzie umieszczenie zabudowanego mikrokomputera po wewnętrznej stronie pomieszczenia, tak żeby dostęp miały tylko osoby które miały pozwolenie przejść przez drzwi.

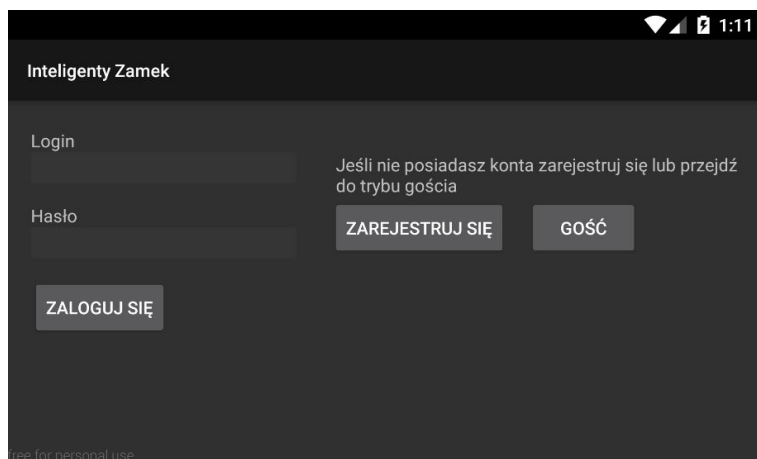
Rozdział 6

Widok graficzny systemu

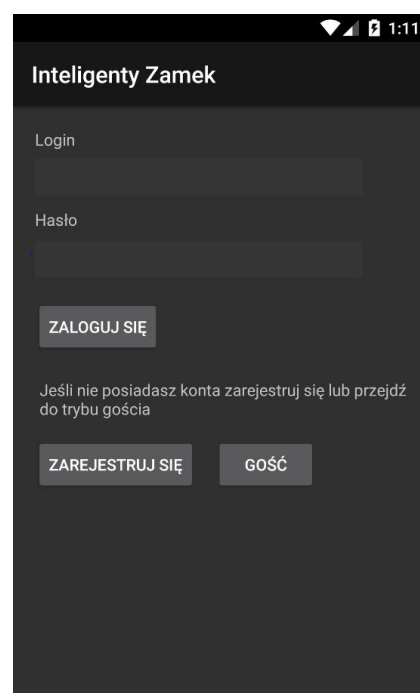
Jedynym elementem graficznym systemu Inteligentnego Zamka jest aplikacja znajdująca się na urządzeniach mobilnych. Poniżej opisano, krótko poszczególne widoki wykonane w środowisku Android Studio.

6.1 Panel logowania użytkownika

Widok umożliwia zalogowanie się użytkownika do systemu poprzez podanie loginu i hasła w odpowiednie pola, a następnie kliknięcie w przycisk “ZALOGUJ SIĘ”. Jeśli nie posiada się konta, można je utworzyć poprzez przycisk “ZAREJESTRUJ SIĘ” lub przejść do panelu Gościa przyciskiem “GOŚĆ”. (Rysunek 6.1 i 6.2)



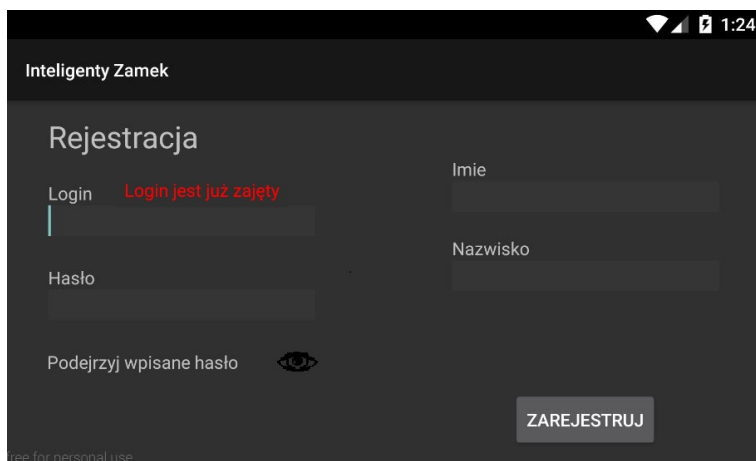
Rysunek 6.2: Panel logowania użytkownika (poziomo)



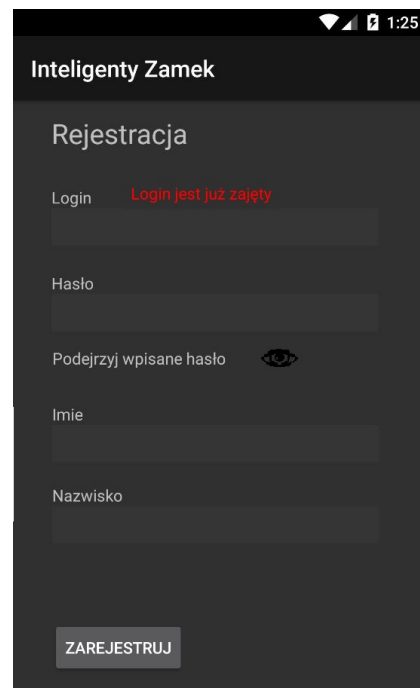
Rysunek 6.1: Panel logowania użytkownika (pionowo)

6.2 Panel rejestracji użytkownika

Panel rejestracji służy do utworzenia nowego użytkownika poprzez podanie loginu, hasła, imienia i nazwiska. Po upewnieniu się, że wszystkie dane są poprawne, aby zakończyć proces rejestracji, klikamy przycisk “ZAREJESTRUJ”. (Rysunek 6.3 i 6.4)



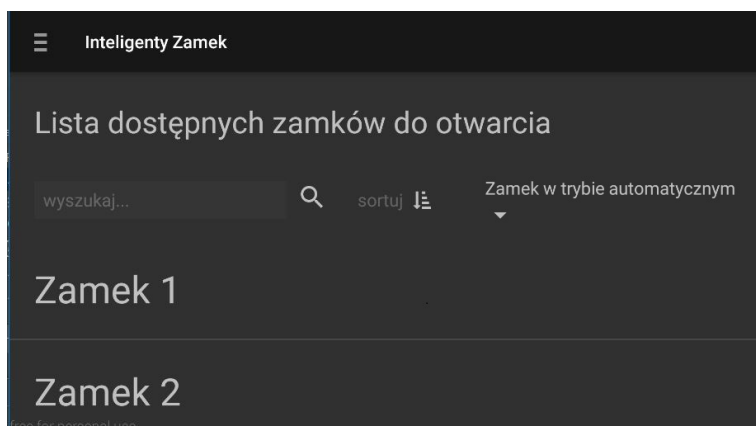
Rysunek 6.4: Panel logowania użytkownika (poziomo)



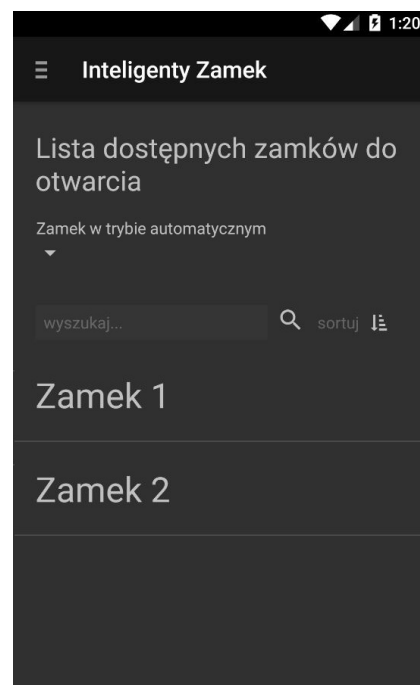
Rysunek 6.3: Panel logowania użytkownika (pionowo)

6.3 Panel listy zamków

Widok listy dostępnych zamków przedstawia listę nazw zamków do jakich dany użytkownik ma dostęp. Ułatwieniem jest możliwość sortowania wyników i wyszukiwanie po nazwach. Kliknięcie w nazwę zamka powoduje otwarcie zamka. Ustawić można również zamek, który ma być otwierany automatycznie gdy jest się w pobliżu zamka. (Rysunek 6.5 i 6.6)



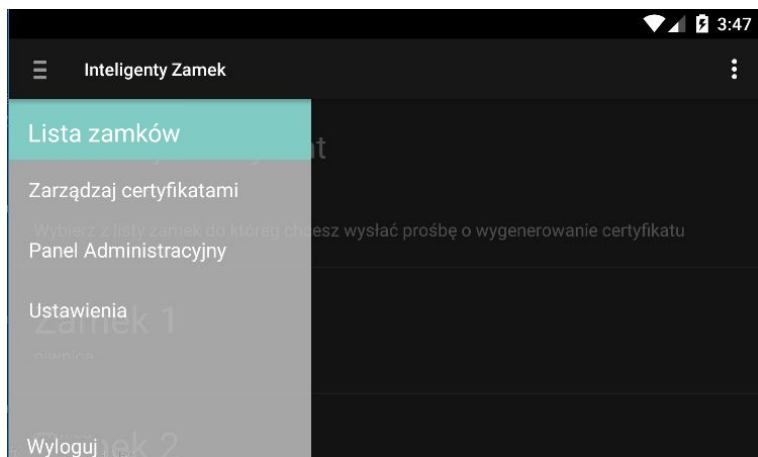
Rysunek 6.6: Lista dostępnych zamków (poziomo)



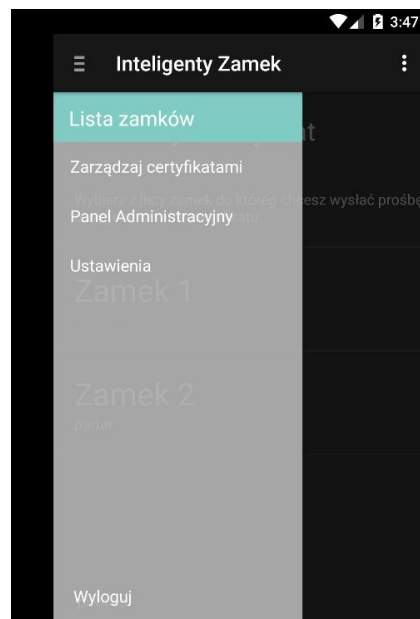
Rysunek 6.5: Lista dostępnych zamków (pionowo)

6.4 Panel boczny

Panel boczny pozwala na szybkie przełączanie pomiędzy widokami. Chowany jest po lewej stronie ekranu. Umożliwia przechodzenie odpowiednio do listy zamków, zarządzania certyfikatami, panelu administracyjnego oraz ustawień. (Rysunek 6.7 i 6.8)



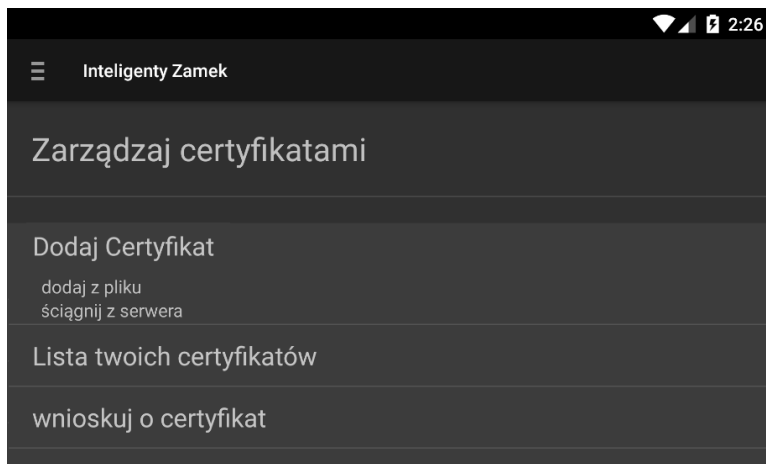
Rysunek 6.8: Panel boczny (poziomo)



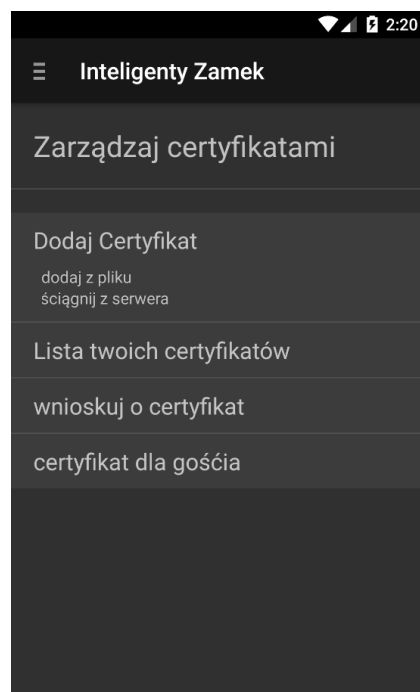
Rysunek 6.7: Panel boczny (pionowo)

6.5 Panel zarządzania certyfikatami

Panel zarządzania certyfikatami umożliwia wybór funkcji dodania certyfikatu w dwóch wariantach (rozwijana lista) - dodania z pliku lub ściągnięcia z serwera. Kolejne pozycje to lista posiadanych certyfikatów, wysłanie wniosku o utworzenie nowego certyfikatu oraz ostatnia opcja to wygenerowanie certyfikatu dla gościa. (Rysunek 6.9 i 6.10)



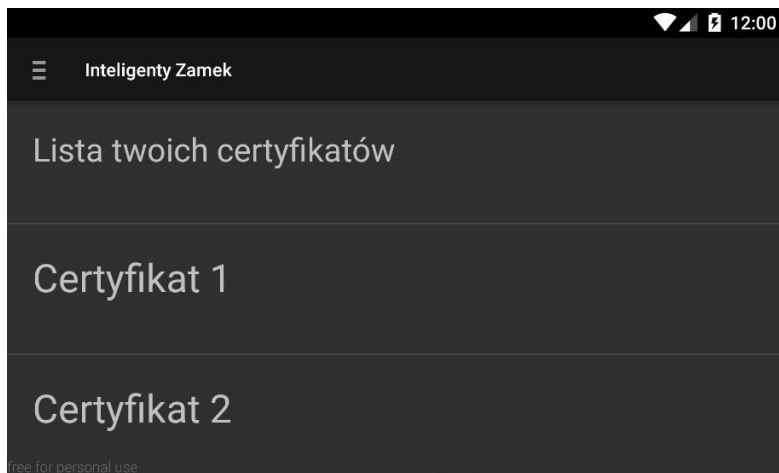
Rysunek 6.10: Panel zarządzania certyfikatami (poziomo)



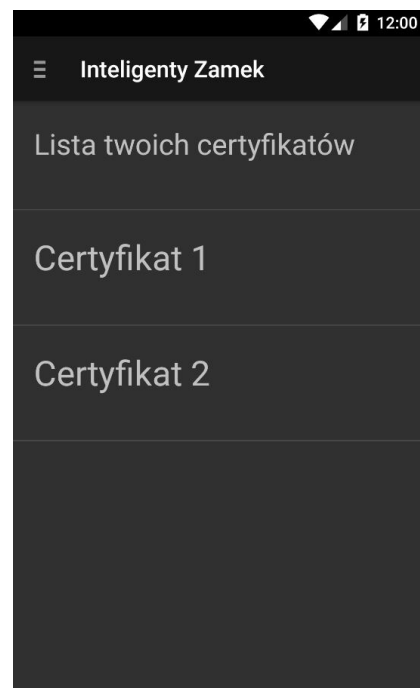
Rysunek 6.9: Panel zarządzania certyfikatami (pionowo)

6.6 Panel listy certyfikatów

Panel listy certyfikatów, jest listą aktualnych certyfikatów należących do użytkownika. Kliknięcie w dany certyfikat przenosi do widoku szczegółowego związanego z operacjami na certyfikacie. (Rysunek 6.11 i 6.12)



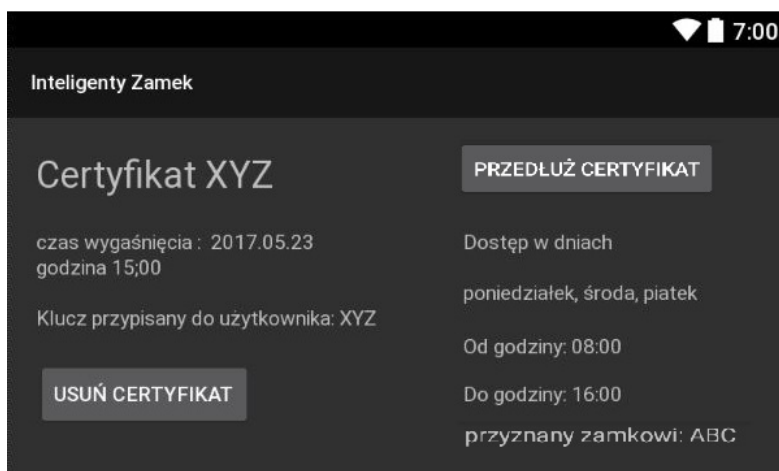
Rysunek 6.12: Panel listy certyfikatów (poziomo)



Rysunek 6.11: Panel listy certyfikatów (pionowo)

6.7 Panel certyfikatu

Panel certyfikatu u góry zawiera nazwę certyfikatu, poniżej informacje o dacie wygaśnięcia, którego zamku dotyczy oraz w jakim czasie przyznaje dostęp. Na dole dostępne są dwa przyciski pozwalające usunąć certyfikat lub wysłać prośbę o przedłużenie ważności. (Rysunek 6.13 i 6.14)



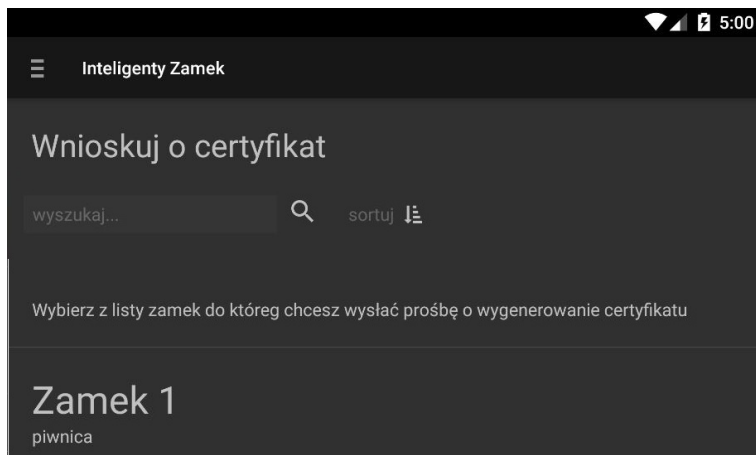
Rysunek 6.14: Panel certyfikatu (poziomo)



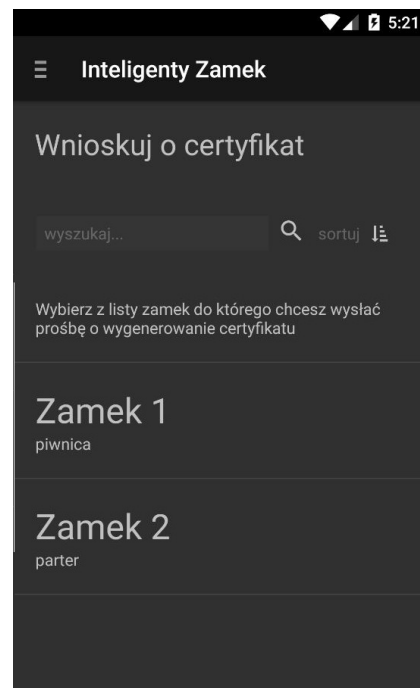
Rysunek 6.13: Panel certyfikatu (pionowo)

6.8 Panel wnioskowania o certyfikat

Panel wnioskowania o certyfikat polega na wybraniu z listy wszystkich zamków, konkretnego do którego chcemy uzyskać dostęp i wysłać wniosek o przydzielenie dostępu. (Rysunek 6.15 i 6.16)



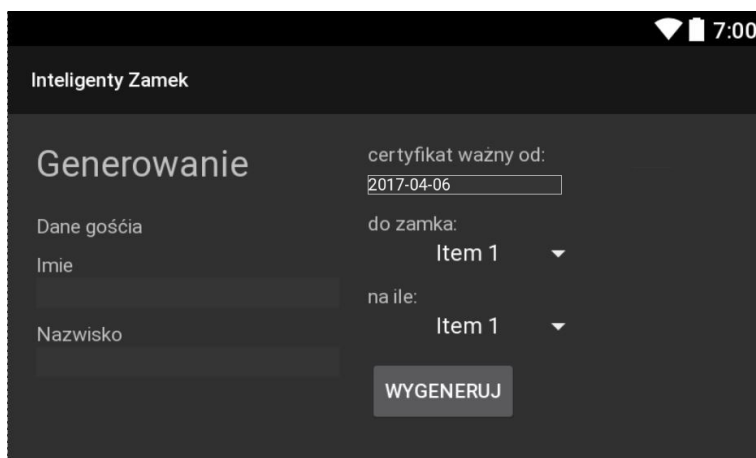
Rysunek 6.16: Panel wnioskowania o certyfikat (poziomo)



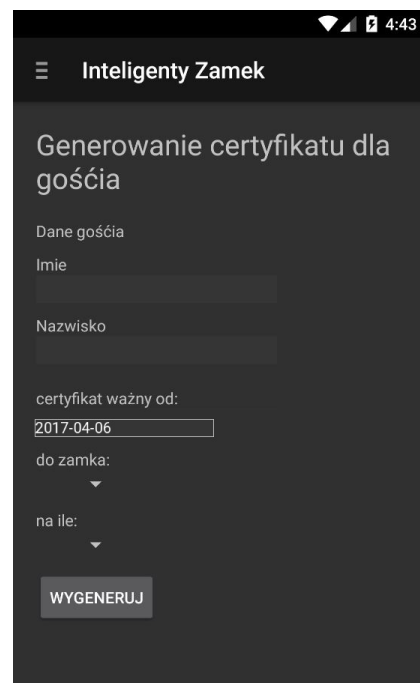
Rysunek 6.15: Panel wnioskowania o certyfikat (pionowo)

6.9 Panel tworzenia certyfikatu dla gościa

Panel tworzenia certyfikatu gościa składa się z danych gościa (imię i nazwisko), daty, od której obowiązuje certyfikat, zamka którego dotyczy oraz jak długo pozostanie ważny (wybór z rozwijanej listy). Aby zakończyć proces generowania należy kliknąć przycisk "WYGENERUJ". (Rysunek 6.17 i 6.18)



Rysunek 6.18: Panel tworzenia certyfikatu dla gościa (poziomo)



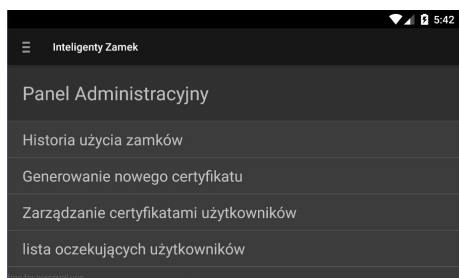
Rysunek 6.17: Panel tworzenia certyfikatu dla gościa (pionowo)

6.10 Panel administratora

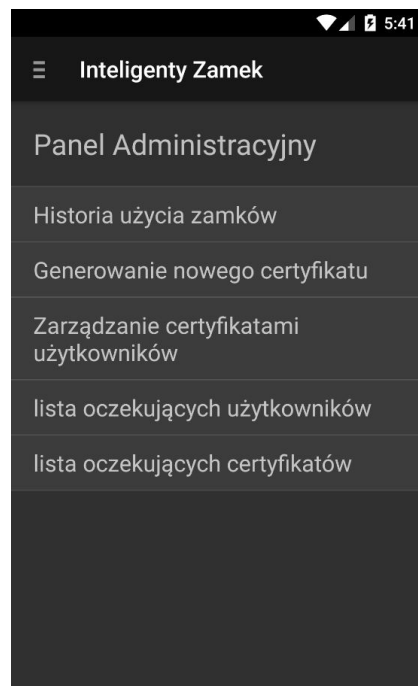
W panelu administratora znajdują się 4 przyciski do administrowania systemem zamków:

- „Historia użycia Zamków”
- „Generowanie nowego klucza”,
- „Zarządzanie kluczami użytkowników”,
- „Lista oczekujących użytkowników na rejestrację”,
- „Lista oczekujących certyfikatów na akceptację”.

Po kliknięciu każdego przycisku przechodzi się do nowego odpowiadającego widoku. (Rysunek 6.19 i 6.20)



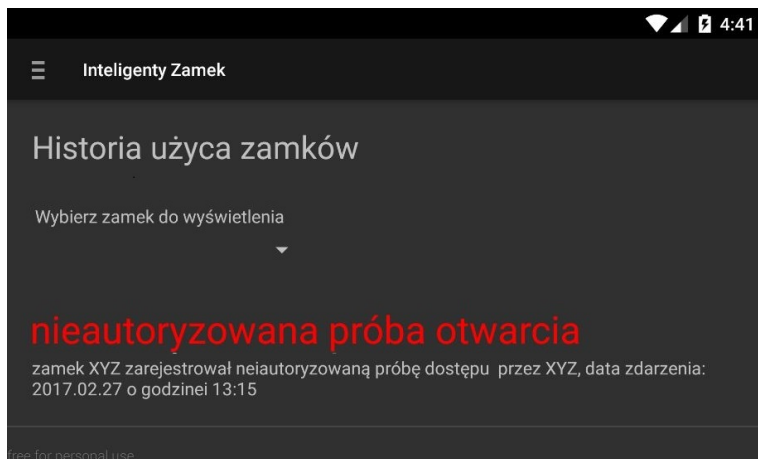
Rysunek 6.20: Panel administratora (poziomo)



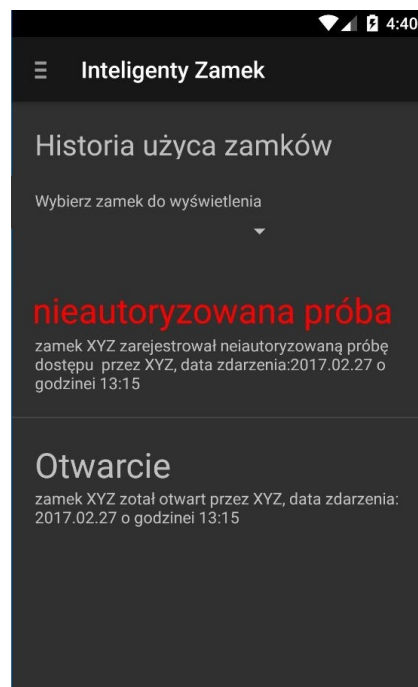
Rysunek 6.19: Panel administratora (pionowo)

6.11 Panel historii użycia zamków

Panel historii użycia zamków składa się z rozwijanej listy wszystkich zamków znajdujących się w systemie, służącej do filtrowania wyników. Dodatkowo wybrać można okres z jakiego wyniki są prezentowane. Pozycje opisane kolorem czerwonym dotyczą operacji odrzucanych, białe przyznających dostęp do zamka. (Rysunek 6.21 i 6.22)



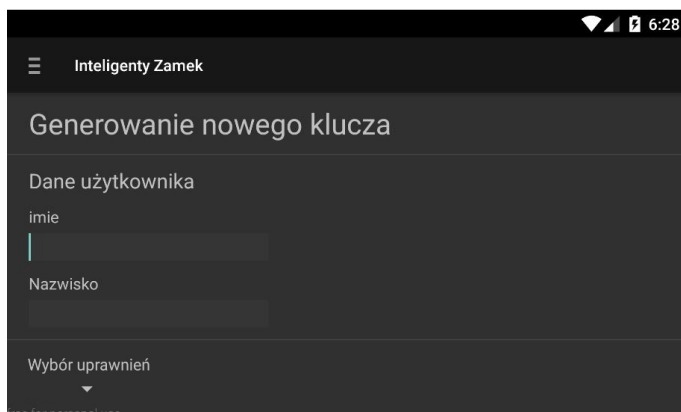
Rysunek 6.22: Panel historii użycia zamków (poziomo)



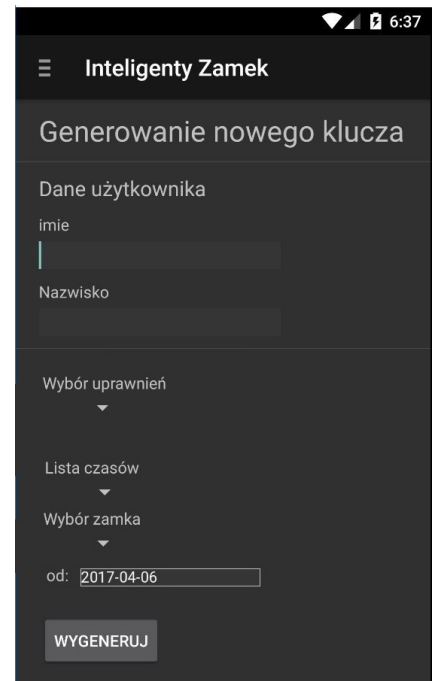
Rysunek 6.21: Panel historii użycia zamków (pionowo)

6.12 Panel generowania nowego certyfikatu (administrator)

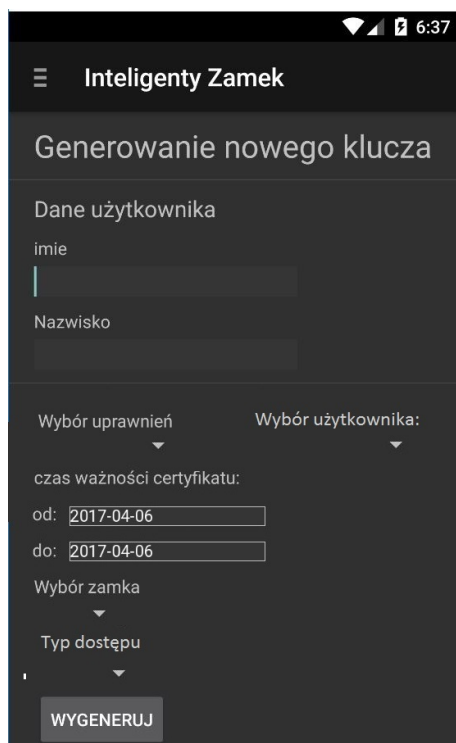
Panel generowanie nowego certyfikatu (administrator) służy do tworzenia nowych certyfikatów przez administratora. W pierwszych polach podaje się imię i nazwisko kogo dotyczy certyfikat. W następnych administrator wybiera typ certyfikatu (gość / użytkownik zalogowany). W zależności od wyboru dostępne są określone pola. Wspólne to od kiedy jest dostęp i jakiego zamka dotyczy. Dla gościa należy wybrać na jak długo jest dostęp, a dla zalogowanego użytkownika do kiedy jest ważność i w jakich godzinach danych dni tygodni ma dostęp. (Rysunek 6.23, 6.24 i 6.25)



Rysunek 6.24: Panel generowania nowego klucza (administrator) (poziomo)



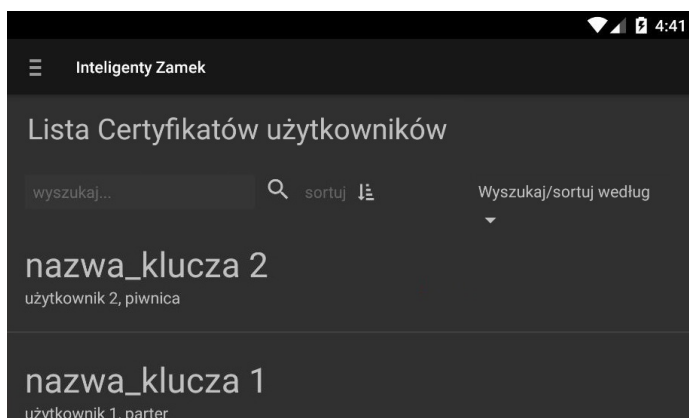
Rysunek 6.23: Panel generowania nowego klucza dla gościa (administrator) (pionowo)



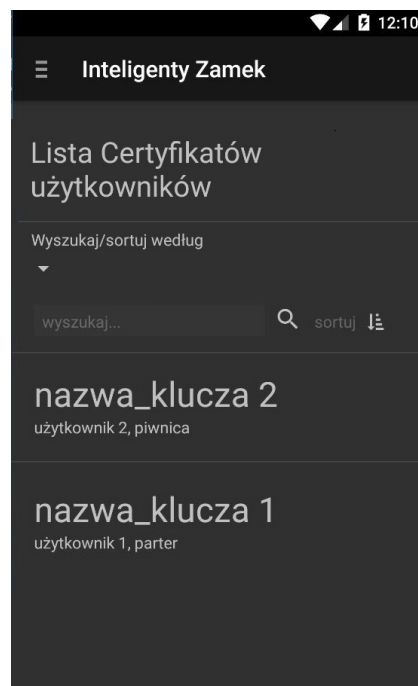
Rysunek 6.25: Panel generowania nowego klucza dla użytkownika zalogowanego (administrator) (poziomo)

6.13 Panel zarządzania certyfikatami (administrator)

Panel zarządzania certyfikatami użytkowników (administrator) jest widokiem tylko wszystkich aktywnych certyfikatów w systemie. Administrator klikając na pozycję przechodzi do panelu certyfikatu opisanego wyżej. Tam może usunąć dostęp lub go przedłużyć. Ułatwieniem jest możliwość wyboru typu sortowania. (Rysunek 6.26 i 6.27)



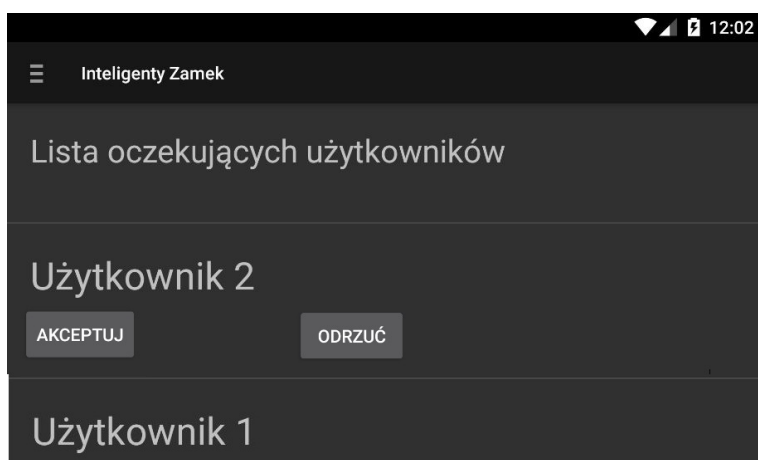
Rysunek 6.27: Panel zarządzania certyfikatami (administrator) (poziomo)



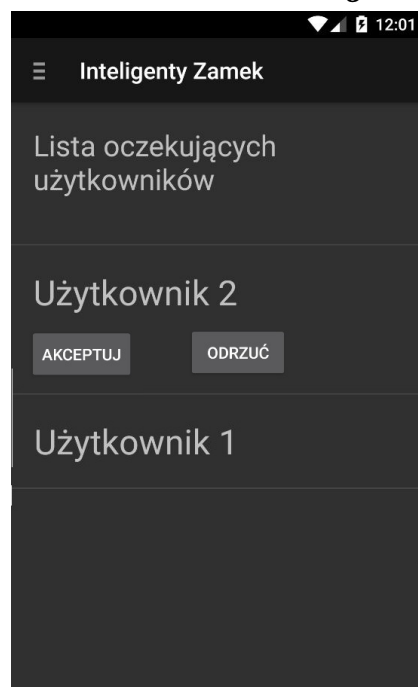
Rysunek 6.26: Panel zarządzania certyfikatami (administrator) (pionowo)

6.14 Panel listy oczekujących użytkowników do rejestracji

Panel listy oczekujących użytkowników jest listą wszystkich gości, którzy ubiegają się o zarejestrowanie. PO kliknięciu w odpowiednią pozycję pojawiają się dwie opcje: “AKCEPTUJ” lub “ODRZUĆ”. (Rysunek 6.28 i 6.29)



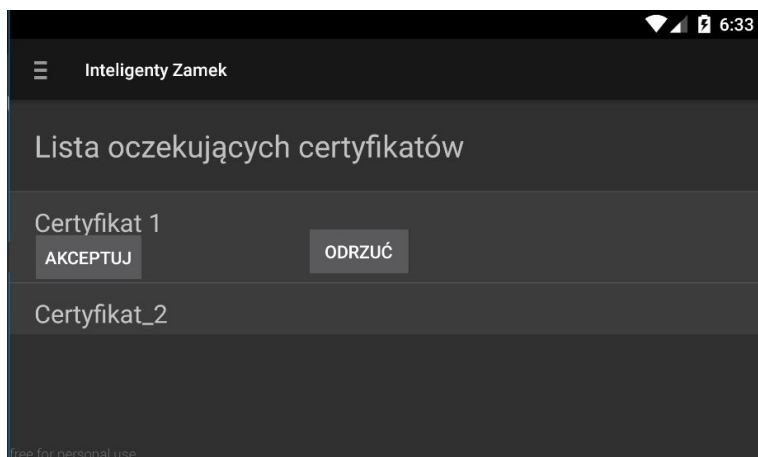
Rysunek 6.29: Panel listy oczekujących użytkowników (poziomo)



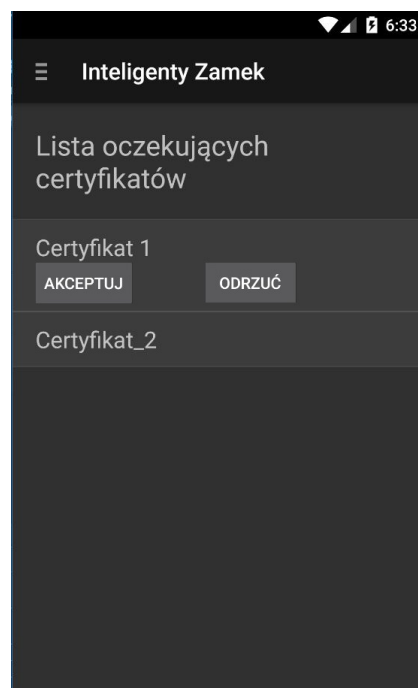
Rysunek 6.28: Panel listy oczekujących użytkowników (pionowo)

6.15 Panel listy oczekujących certyfikatów do wygenerowania

Panel listy oczekujących certyfikatów jest listą wszystkich certyfikatów, które ubiegają się o akceptację administratora. Po kliknięciu w odpowiednią pozycję pojawiają się dwie opcje: “AKCEPTUJ” lub “ODRZUĆ”. (Rysunek 6.30 i 6.31)



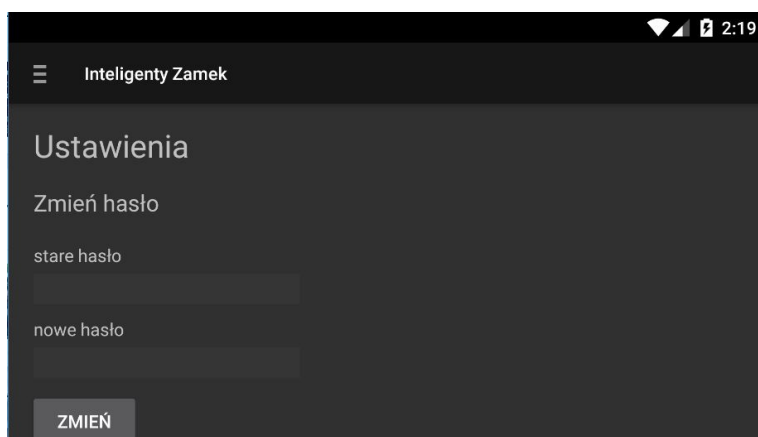
Rysunek 6.31: Panel listy oczekujących certyfikatów (poziomo)



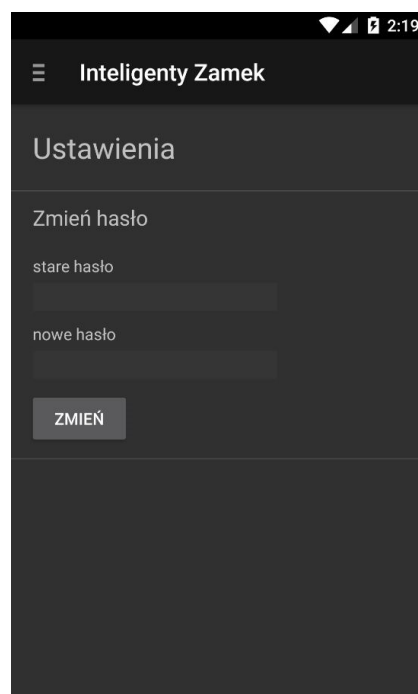
Rysunek 6.30: Panel listy oczekujących certyfikatów (pionowo)

6.16 Panel ustawień konta

W panelu ustawień użytkownik może zmienić hasło do swojego konta. Wymagane jest podanie starego hasła, a następnie nowego. Aby zakończyć czynność klikamy na “ZMIEN”. (Rysunek 6.32 i 6.33)



Rysunek 6.33: Panel ustawień konta (poziomo)



Rysunek 6.32: Panel ustawień konta (pionowo)