



**Uniwersytet
Kazimierza
Wielkiego
w Bydgoszczy**

Dokumentacja Zespołowego Projektu Informatycznego

Zespół nr 2: INCOGUTO

Koordynator Projektu:

Justyna Kardasz

Autorzy:

Mikołaj Białek

Kacper Banaszak

Hubert Badziąg

Dominik Kończal

Dawid Lewandowski

Michał Hempel

Spis treści

Spis treści.....	1
Opis Projektu	3
Grupy Docelowe.....	3
Podział zespołu:	3
Kluczowe Funkcje Aplikacji	5
1. Stealth Mode – Ukryta Aplikacja	5
2. Zaawansowane Szyfrowanie Wiadomości	5
3. Samoniszczące się Wiadomości	5
4. Autoryzacja i Bezpieczeństwo	6
5. Anonimowość i Prywatność	6
6. Panic Button	6
Wykorzystane Technologie.....	7
Struktura aplikacji	7
Struktura kodu	8
Kolekcje w Firestore:	8
Możliwości rozwoju aplikacji	9
Podsumowanie:	10

Opis Projektu

Incoguto to zaawansowana aplikacja do prywatnej komunikacji, która maskuje się jako zwykły kalkulator, zapewniając pełną dyskrecję. Po wprowadzeniu specjalnego kodu z autentykatora, aplikacja przełącza się w tryb czatu szyfrowanego, oferując funkcje takie jak czaty prywatne i grupowe, samoniszczące się wiadomości, blokada screenshotów oraz autoryzacja dynamicznym kodem, generowanym w dedykowanym autentykatorze. Głównym celem projektu jest stworzenie bezpiecznego środowiska komunikacji dla użytkowników, którzy cenią sobie prywatność – zarówno w kontekście osobistym, jak i biznesowym.

Grupy Docelowe

- Osoby dbające o prywatność
- Użytkownicy chcący uniknąć inwigilacji.
- Firmy wymagające bezpiecznej komunikacji wewnętrznej.
- Organizatorzy i uczestnicy gier terenowych
- Ofiary przemocy domowej
- Dziennikarze i informatorzy
- Małe i duże firmy
- Małe, zamknięte grupy współpracujące poufnie

Podział zespołu:

- **Justyna Kadrasz:** koordynator projektu
- **Mikołaj Biątek:**
 - Planowanie struktury projektu i podział zadań pomiędzy członków zespołu
 - Bieżący nadzór nad realizacją poszczególnych modułów
 - Wsparcie techniczne dla zespołu backendowego
 - Projekt i implementacja mechanizmu szyfrowania wiadomości:
 - Szyfrowanie wiadomości po stronie klienta (End-to-End Encryption)

- Zabezpieczenie danych przesyłanych i zapisywanych w bazie
- **Kacper Banaszak:**
 - implementacja interfejsu użytkownika aplikacji we Flutterze
 - Przygotowanie i implementacja widoków aplikacji
 - stworzenie strony internetowej projektu
 - zaprojektowanie i przygotowanie ulotki promocyjnej
 - dbanie o spójność wizualną projektu
- **Hubert Badziąg:**
 - Implementacja logiki czatów prywatnych
 - Implementacja czatów grupowych
 - Obsługa wysyłania i odbierania wiadomości
 - Integracja czatów z mechanizmem szyfrowania
 - Zapewnienie poprawnej synchronizacji wiadomości między użytkownikami
 - Obsługa samoniszczących się wiadomości po odczycie
- **Dominik Kończal:**
 - Integracja aplikacji z Firebase
 - Obsługa połączeń aplikacji z usługami backendowymi
 - Obsługa zaproszeń i relacji między użytkownikami
 - Konfiguracja reguł bezpieczeństwa Firebase
 - Ogólne testy aplikacji
- **Dawid Lewandowski:**
 - Implementacja dedykowanej aplikacji autentykatora
 - Generowanie dynamicznych kodów autoryzacyjnych
 - Obsługa mechanizmu zmiany kodu przez użytkownika
 - Integracja autentykatora z główną aplikacją
 - Zapewnienie niezależności kodów od danych osobowych użytkownika
- **Michał Hempel:**
 - Implementacja systemu logowania do aplikacji
 - Obsługa tworzenia kont użytkowników opartych na losowym identyfikatorze
 - Zarządzanie listą znajomych

- Sformułowanie treści ulotki projektu

Kluczowe Funkcje Aplikacji

1. Stealth Mode – Ukryta Aplikacja

Stealth Mode to fundament bezpieczeństwa aplikacji, którego celem jest całkowite zamaskowanie jej prawdziwego przeznaczenia.

- Maskowanie interfejsu
Aplikacja z zewnątrz wygląda jak standardowy kalkulator. Interfejs, ikona oraz nazwa aplikacji nie dają po sobie poznać, że jest to ukryta aplikacja
- Ukryty dostęp do właściwego czatu
Prawdziwy interfejs komunikatora jest dostępny dopiero po wpisaniu określonej sekwencji znaków w kalkulatorze, a następnie zalogowanie się na swoje konto przy użyciu loginu i kodu wygenerowanego w autentykatorze, który jest osobną aplikacją

2. Zaawansowane Szyfrowanie Wiadomości

Bezpieczeństwo komunikacji oparte jest na nowoczesnych standardach kryptograficznych.

- End-to-End Encryption (E2EE)
Wszystkie wiadomości są szyfrowane na urządzeniu nadawcy i odszyfrowywane wyłącznie na urządzeniu odbiorcy.
- Zapewnienie najwyższej możliwej prywatności
Nikt poza osobami uprawnionymi nie ma możliwości wglądu do cudzych danych, grup oraz wysłanych i odebranych wiadomości.

3. Samoniszczące się Wiadomości

System wiadomości tymczasowych zwiększa kontrolę użytkownika nad swoimi danymi.

- Blokada zrzutów ekranu
Aplikacja blokuje próby tworzenia zrzutów ekranu, zapewniając pewną prywatność rozmów dla wszystkich uczestników.

- Jednorazowy odczyt Wiadomości są niszczone po odczytaniu i opuszczeniu chatu przez obie strony lub uczestników grupy.

4. Autoryzacja i Bezpieczeństwo

Wielowarstwowy system zabezpieczeń chroni konto użytkownika.

- Dynamiczny kod autoryzujący logowanie
Kod autentykatora użytkownika jest zmieniany automatycznie co minutę lub ręcznie na żądanie użytkownika w dowolnym momencie, utrudnia śledzenie i identyfikację, a także nie jest powiązany z danymi osobowymi.

5. Anonimowość i Prywatność

Aplikacja została zaprojektowana z myślą o maksymalnej ochronie tożsamości.

- Brak danych osobowych. Rejestracja nie wymaga: numeru telefonu, adresu e-mail, imienia ani nazwiska. Konto tworzone jest na podstawie losowego identyfikatora.
- Brak logów rozmów. Serwer nie przechowuje historii czatów po ich usunięciu ani nie archiwizuje treści wiadomości.

6. Panic Button

Funkcje awaryjne pozwalają natychmiast zabezpieczyć dane. Panic Button: Jedno działanie powoduje:

- natychmiastowe zamknięcie czatu,
- przejście do fałszywego interfejsu (kalkulatora).

Wykorzystane Technologie

- **Frontend:** Flutter

Flutter to framework UI firmy Google służący do tworzenia aplikacji mobilnych, webowych i desktopowych z jednego kodu źródłowego. Umożliwia szybkie tworzenie wydajnych i estetycznych interfejsów użytkownika.

- **Backend:** Firebase

Firebase to platforma Backend-as-a-Service (BaaS) oferowana przez Google, która dostarcza gotowe usługi backendowe, takie jak autoryzacja użytkowników, hosting, funkcje chmurowe oraz integrację z aplikacjami mobilnymi i webowymi.

- **Baza danych:** Firestore

Cloud Firestore to nierelacyjna, dokumentowa baza danych w chmurze, będąca częścią Firebase. Umożliwia przechowywanie i synchronizację danych w czasie rzeczywistym oraz skalowanie aplikacji bez konieczności zarządzania infrastrukturą.

Struktura aplikacji

Aplikacja została podzielona na widoki, żeby zachować porządek w strukturze logicznej aplikacji i kodu.

Autentykator:

admin_panel_screen.dart - widok administratora, z którego poza generowaniem kodu autentykującego logowanie możliwe jest tworzenie nowych kont użytkowników w systemie.

auth_code_screen.dart - tutaj widoczny jest kod autentykujący, który automatycznie odświeża się co minutę. Użytkownik ma możliwość wymuszenia resetu kodu.

auth_login_screen.dart - logowanie do autentykatora za pomocą loginu i hasła.

Aplikacja Chatu:

friend_screen.dart - lista znajomych, umożliwia dodawanie i usuwanie znajomych.

calculator_screen.dart - widok kalkulatora, domyślnie widoczny przy uruchomieniu aplikacji. Przejście do chatu następuje po wpisaniu odpowiedniej kombinacji znaków.

friend_request_screen.dart- widok zaproszeń do znajomych. Tutaj wyświetlane są przychodzące zaproszenia od innych użytkowników, można je zaakceptować lub odrzucić.

chat_list_screen.dart - jest to lista wszystkich dostępnych w danej chwili chatów indywidualnych oraz grupowych, które są odpowiednio oznaczone w celu ich odróżnienia. Obsługiwane są również powiadomienia informujące o nowych wiadomościach.

group_chat_screen.dart- widok pojedynczego chatu grupowego, wyświetlane są wiadomości wraz z nazwą autora oraz czasem wysłania. Moderatorzy i administrator grupy mogą modyfikować jej nazwę, dodawać lub usuwać członków, administrator ma dodatkową możliwość usunięcia całej grupy.

group_create_screen.dart- widok tworzenia chatu grupowego, na którym można dodać użytkowników, twórca grupy jest automatycznie jej administratorem. Wiadomości są usuwane po odczytaniu ich przez wszystkich członków grupy.

private_chat_screen.dart- widok chatu prywatnego, tak jak w chatach grupowych, widoczne są wiadomości i ich czas wysłania. Wiadomości są usuwane po ich odczytaniu przez obu użytkowników

messenger_login_screen.dart- widok logowania do aplikacji chatowej. Tutaj wykorzystywany jest login konta oraz kod z autentykatora do uzyskania dostępu do chatów.

Struktura kodu

Kolekcje w Firestore:

- chats

🏠 > chats > 1tQr6XYQrXOz9...		
(default)	chats	1tQr6XYQrXOz9UHgyOwO
+ Start collection	+ Add document	+ Start collection
chats >	1tQr6XYQrXOz9UHgyOwO >	+ Add field
friendRequests	2P6uGBSI+4D1j9Hx50Sg	createdAt: December 9, 2025 at 10:52:42 AM UTC+1
users	3wKxMRpKSP1TLnhpB39	members
	⋮ 6Q64LDyZqeCYma1J0xT	0 "4TSrDHBtsY7yBTiZgi3V"
	8oALrUXzWnv0072GLfJs	1 "BOWItaOYvhFAJhAmaFp8"
	EBZRFgwFO1muPtYYGtTM	2 "b9Z5OKD2Ng7x7Uhoch2z"
	HPOTfmvYRnGct6fWeH6r	name: "nazwa1"
	K14ztH7cUNb0bCjYxwRM	ownerId: "b9Z5OKD2Ng7x7Uhoch2z"
	XrD2dQKLnGIUsang7DGH	roles
	YA75DWSRLY4LSeIYhUG8	4TSrDHBtsY7yBTiZgi3V: "moderator"
	bmLIPd9ZjT6LUOb99Fq	BOWItaOYvhFAJhAmaFp8: "user"
	f138xLqfmIR8SCF9p57F	b9Z5OKD2Ng7x7Uhoch2z: "admin"
	lkna4BYVtH1i139snxQI	type: "group"
	msdVVZvjLQPg06pK2IYd	
	p9ShxnI3l6LJL7o8pWvF	
	p_4TSrDHBtsY7yBTiZgi3V_BOWItaOYvhFAJhAm...	

• friendRequests

🏠 > friendRequests > 4cb4CpN6LDbV...		
(default)	friendRequests	4cb4CpN6LDbVwVm2KZt
+ Start collection	+ Add document	+ Start collection
chats	4cb4CpN6LDbVwVm2KZt >	+ Add field
friendRequests >	FWar2H6XnA3Shc8DPE47	createdAt: December 1, 2025 at 9:25:32 PM UTC+1
users	PoJihwk1Bdis1sRzWIY	fromUid: "BOWItaOYvhFAJhAmaFp8"
	dH0JaAKUVRm5o2p4JVqh	respondedAt: December 1, 2025 at 9:25:34 PM UTC+1
	fCZsFzZ9pHvvH9k5idyC	status: "accepted"
	1755vr4WZP4PD1gX00Nnb	toUid: "b9Z5OKD2Ng7x7Uhoch2z"
	unsfWU1ufgSa0Ryqyz0	

• users

🏠 > users > bPJdy6t2NpC...		
(default)	users	bPJdy6t2NpCBH21sEV
+ Start collection	+ Add document	+ Start collection
chats	2SIqT8LUNwKz8u3pMFgr	friends
friendRequests	4TSrDHBtsY7yBTiZgi3V	userChats
users >	BOWItaOYvhFAJhAmaFp8	
	JKR8zuLqVvkh3BPbHLjn	+ Add field
	b9Z5OKD2Ng7x7Uhoch2z	authCode: "A9KT9C"
	⋮ bPJdy6t2NpCBH21sEV >	authCodeUpdatedAt: January 13, 2026 at 4:07:28 PM UTC+1
	fTcqrcWwKp9S0aajbCt	createdAt: December 16, 2025 at 1:59:19 PM UTC+1
	iwyuOWLqLJWYtcaD91YL	login: "kacper"
	p6nTKuNtPBRCSGov9zsg	nickname: "Kapi"
	rc3xSIWajkWsXVDFo0Ee	password: "kacper"
		role: "user"

Możliwości rozwoju aplikacji

- **Rozszerzenie możliwości chatu** (wiadomości głosowe, wysyłanie obrazów, odpowiedzi cytowane, zdjęcie otwierane za pomocą kodu QR)
- **Ukryte czaty w innych aplikacjach** (np. Kalendarz lub kontakty.)
- **Zaawansowane wiadomości tymczasowe** (możliwość ustawienia czasu życia wiadomości)
- **Dodatkowe metody uwierzytelniania** (biometria)
- **Geolokalizacja z autousuwaniem** (Możliwość sprawdzenia i udostępnienia swojej lokacji znajomym, z zachowaniem automatycznego usuwania, tak jak w chatach.)
- **Integracja z VPN** (Zapewnienie maksymalnego bezpieczeństwa przesyłanych danych.)
- **Wydanie aplikacji na Google Apps i AppStore** (Możliwość dotarcia do szerokiej grupy użytkowników).

Podsumowanie:

Incoguto stanowi nowoczesne podejście do prywatnej komunikacji między użytkownikami, zachowując tym samym bezpieczeństwo. Aplikacja oferuje osobie korzystającej z aplikacji anonimowość, co jest ważne dla osób, którzy cenią sobie prywatność korzystając z takich funkcji jak na przykład samoniszczące się wiadomości.

Projekt osiągnął wszystkie założone cele oraz otworzył duże możliwości rozwoju aplikacji, jak na przykład wyżej wymienione rozszerzone możliwości chatu. Realizacja projektu umożliwiła zespołowi uzyskanie nowego doświadczenia w bezpieczeństwie danych oraz pracy w zespole.