**Traveller’s App**

1. **Увод**

Идеята на приложението е да предостави възможност на потребителите си да споделят преживяванията си с техните приятели, както и с други потребители на платформата. Traveller’s App позволява споделяне на мнение, снимки и поставяне на оценка за дадена дестинация, която конкретен потребител е посетил. Така потребители на приложението ще могат да потърсят отзиви за някое място, за което плануват почивка или екскурзия, което да им помогне в избора. След това те също могат да споделят емоциите и преживяванията си, помагайки на други потребители да придобият представа за конкретна дестинация. Емоциите са по-хубави, когато са споделени, което е и основната идея на платформата Traveller’s App.

1. **Проектиране и структура**

В основата на приложението е проектът “Database”, благодарение на който потребителите на Traveller’s App могат да управляват своите данни.

A screenshot of a cell phone

Description automatically generatedДруг подход, който е използван при създаването на Traveller’s App е така наречената “ORM” техника в програмирането. Има три класа-модели – **User, Destination, Excursion**, както и четири класа за работа с моделите – **UserRepository, ExcursionRepository, DestinationRepository, FriendshipRepository**. Те служат за превръщането на записи в базата от данни на обекти от някой от моделите.

A screenshot of a cell phone

Description automatically generatedUserRepository.h

FriendshipRepository.h

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedExcursionRepository.h

A screenshot of a social media post

Description automatically generatedDestinationRepository.h

Excursion.h

A screenshot of a social media post

Description automatically generatedA screenshot of a cell phone

Description automatically generatedUser.h

Destination.h

1. **Реализация и тестване**

Всяка команда, с която работят потребителите на Traveller’s App, е имплементирана чрез така нареченият “Command Pattern”. Тези команди използват четирите …Repository класа, за да управляват данните в базата от данни.

Следните тестове за коректност са проведени:

1. Достъпване на команди, които са достъпни само за потребители, които са се вписали в приложението
2. Регистрация на съществуващ потребител
3. Вход с несъществуващ потребител
4. Добавяне като приятел на потребител, който не съществува
5. Добавяне в приятели на потребител, който е вече приятел с текущия
6. Премахване на приятелство, което не съществува
7. Добавяне на екскурзия с невалиден период на престой
8. Добавяне на екскурзия със забранени думи/символи в коментарите
9. Добавяне на екскурзия без коментар(коментарът служи за обосновка на оценката и затова е задължителен)
10. Премахване на незаписана екскурзия
11. Достъпване на екскурзия на потребител, който не е приятел на достъпващия такъв.
12. **Заключение**

Началните цели са изпълнени успешно, а именно предоставянето на приложение, чрез което хората да могат да споделят своите емоции и преживявания.

За в бъдеще ще бъдат добавени покани за приятелство, тъй като в момента добавянето на приятел е моментално, без потвърждение от другата страна, както и възможност за показване на конкретни снимки от вече запазените, тъй като в момента те не са видими за никой.

Нужни са някои оптимизационни подобрения, с които командите да извършват съответните действия за по-кратко време, но те зависят от проекта “Database”.