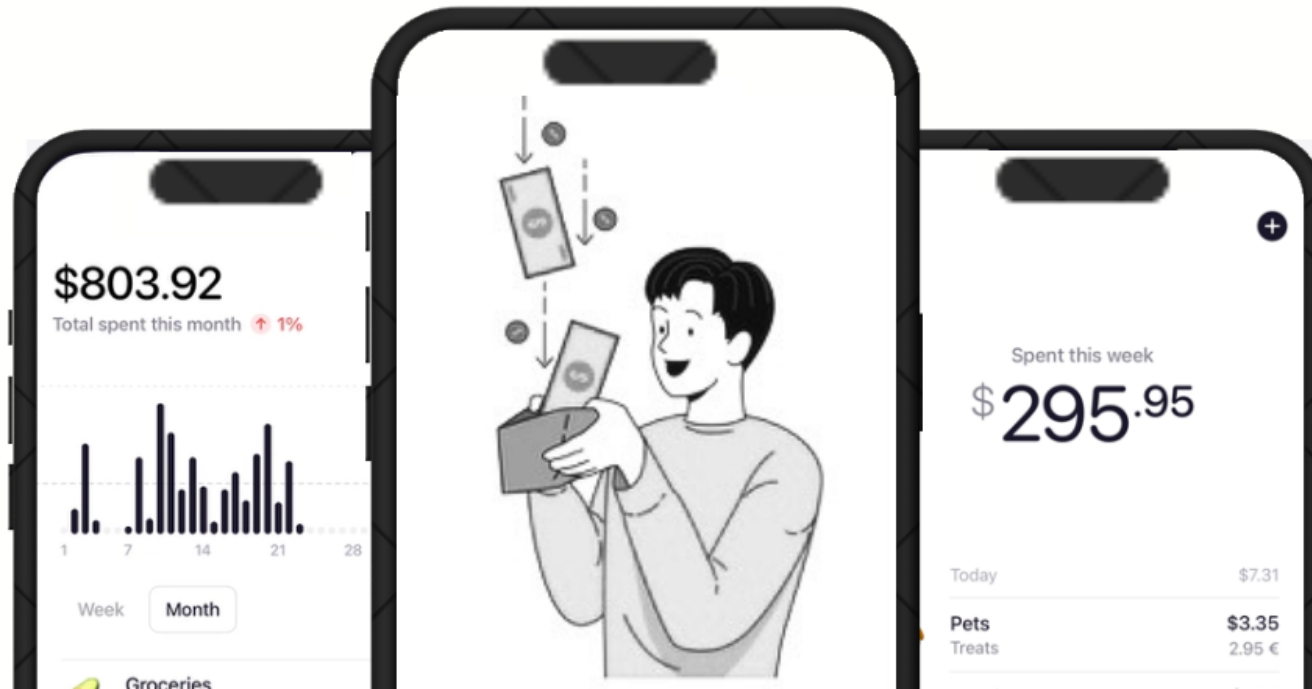




IT TALENTS
Training Camp

FINANCE TRACKER



WHO ARE WE?

Georgi Shishinyov



Kameliya Kamenova



Cadets in IT Talents Season 15

USED TECHNOLOGIES (TECH STACK)

MySQL
Workbench

Spring Web

Vonage
SMS API

Apache POI

KrasiGPT

Gson

Lombok

Spring Boot

Hibernate

Model
Mapper

Gson
Validator

Java Bean
Validation

Exchange
Rates API

Java 17

Postman

Maven

BCrypt

JUnit

Chat
GPT-
3/4

IntelliJ IDEA

IText

GIT

Spring Data
JPA

JavaMail
API

Log4j

Swagger

Mockito

GENERAL ENDPOINTS

user-controller

PUT	/users/{id}
DELETE	/users/{id}
PUT	/users/{id}/password-change
POST	/users
POST	/users/{id}/sms/confirm
POST	/users/logout
POST	/users/login
GET	/users/{id}/invalidate
GET	/email-validation

transaction-controller

GET	/transactions/{id}
PUT	/transactions/{id}
DELETE	/transactions/{id}
POST	/transactions
GET	/users/{id}/transactions
GET	/transactions/filter
GET	/accounts/{id}/transactions

planned-payment-controller

GET	/planned-payments/{id}
PUT	/planned-payments/{id}
DELETE	/planned-payments/{id}
POST	/planned-payments
GET	/planned-payments/{id}/transactions
GET	/accounts/{id}/planned-payments

budget-controller

GET	/budgets/{id}
PUT	/budgets/{id}
DELETE	/budgets/{id}
GET	/budgets
POST	/budgets

account-controller

GET	/accounts/{id}
PUT	/accounts/{id}
DELETE	/accounts/{id}
GET	/accounts
POST	/accounts
GET	/accounts/{id}/export

category-controller

GET	/categories
GET	/categories/{id}
GET	/categories/{id}/image
GET	/categories/filter

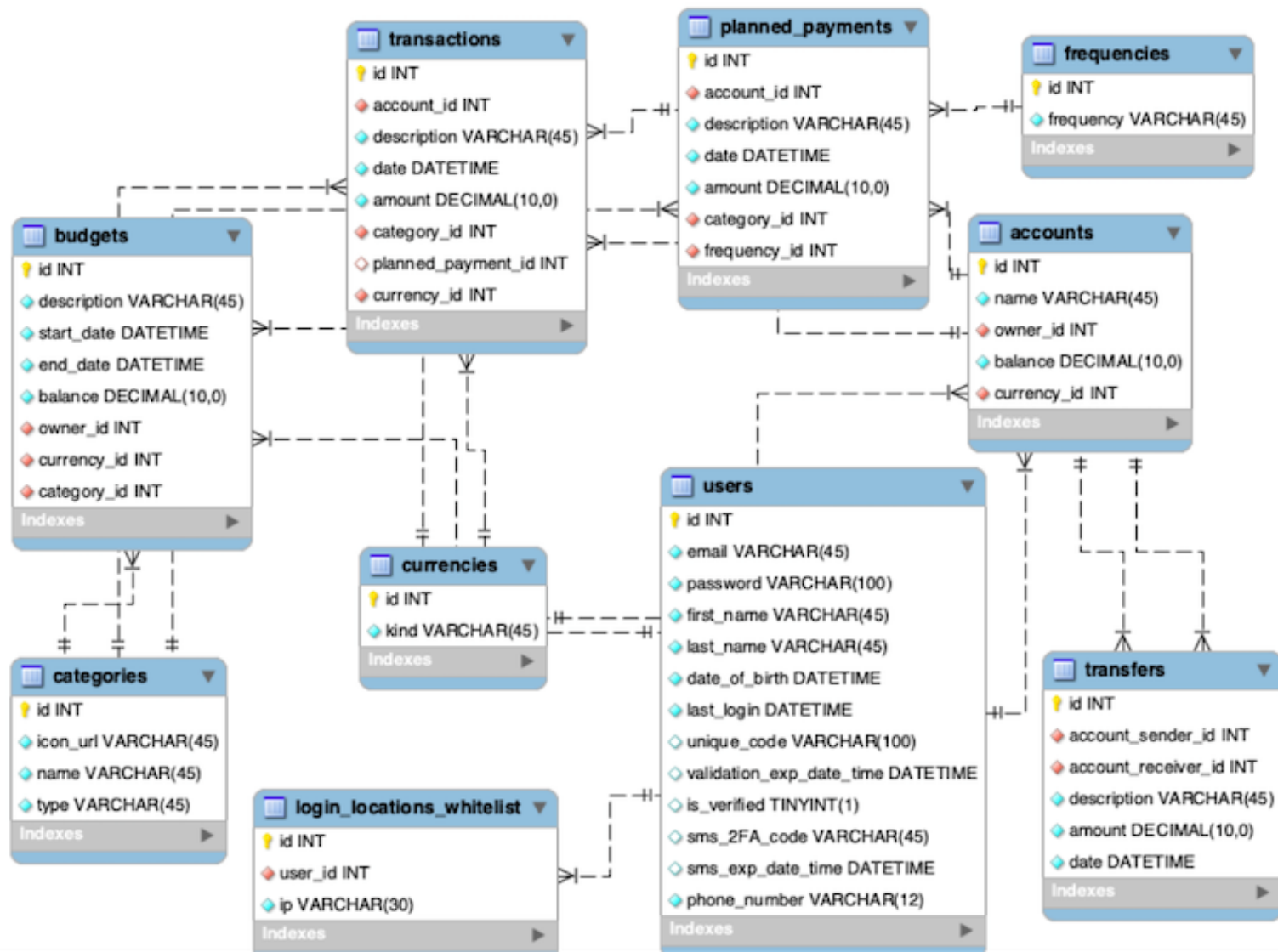
transfer-controller

GET	/transfers
POST	/transfers
GET	/transfers/{id}

currency-controller

GET	/currencies
-----	-------------

DATABASE ER DIAGRAM

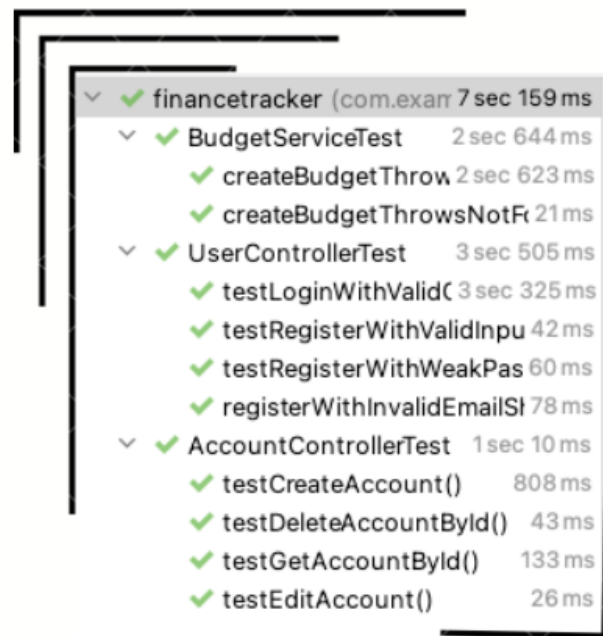


PROJECT STRUCTURE



INTERESTING FEATURES

- Unit testing
- 2FA authentication with SMS
- Swagger documentation
- Logging info with Log4j
- Exporting account data in Excel, PDF, and JSON formats
- Spring validation
- Automatic currency conversion with API
- Sending a confirmation email for account verification
- Logging out from all devices in case of security threats
- Pagination for responses with large amounts of data
- Cron job for cleaning unverified profiles
- Cron job for automatic execution of planned payments



A screenshot of a JUnit test runner interface showing a list of tests and their execution times. The tests are grouped under the package 'com.example.financetracker'. Each test is preceded by a green checkmark, indicating successful execution. The tests are: BudgetServiceTest (2 sec 644 ms), UserControllerTest (3 sec 505 ms), and AccountControllerTest (1 sec 10 ms). Each test has several sub-tests, all of which also passed.

✓ financetracker (com.example)	7 sec 159 ms
✓ BudgetServiceTest	2 sec 644 ms
✓ createBudgetThrow	2 sec 623 ms
✓ createBudgetThrowsNotFr	21 ms
✓ UserControllerTest	3 sec 505 ms
✓ testLoginWithValidC	3 sec 325 ms
✓ testRegisterWithValidInpu	42 ms
✓ testRegisterWithWeakPas	60 ms
✓ registerWithInvalidEmailSt	78 ms
✓ AccountControllerTest	1 sec 10 ms
✓ testCreateAccount()	808 ms
✓ testDeleteAccountByld()	43 ms
✓ testGetAccountByld()	133 ms
✓ testEditAccount()	26 ms



PROJECT STATISTICS

3 days writing the core

4 days implementing additional features

More than 3900 rows of code

41 endpoints

Unit tests coverage:

Class - 33%, Method - 16%



DEMO TIME?



THANK YOU

