OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF

FAKULTÄT FÜR INFORMATIK

25. April 2021

TREASURETROVE: Money Control
Milestone 2 – Basic Prototype

LAKSHMI HARIKA NAKKA **(Scrum Master/Developer/ Functional Testing)**

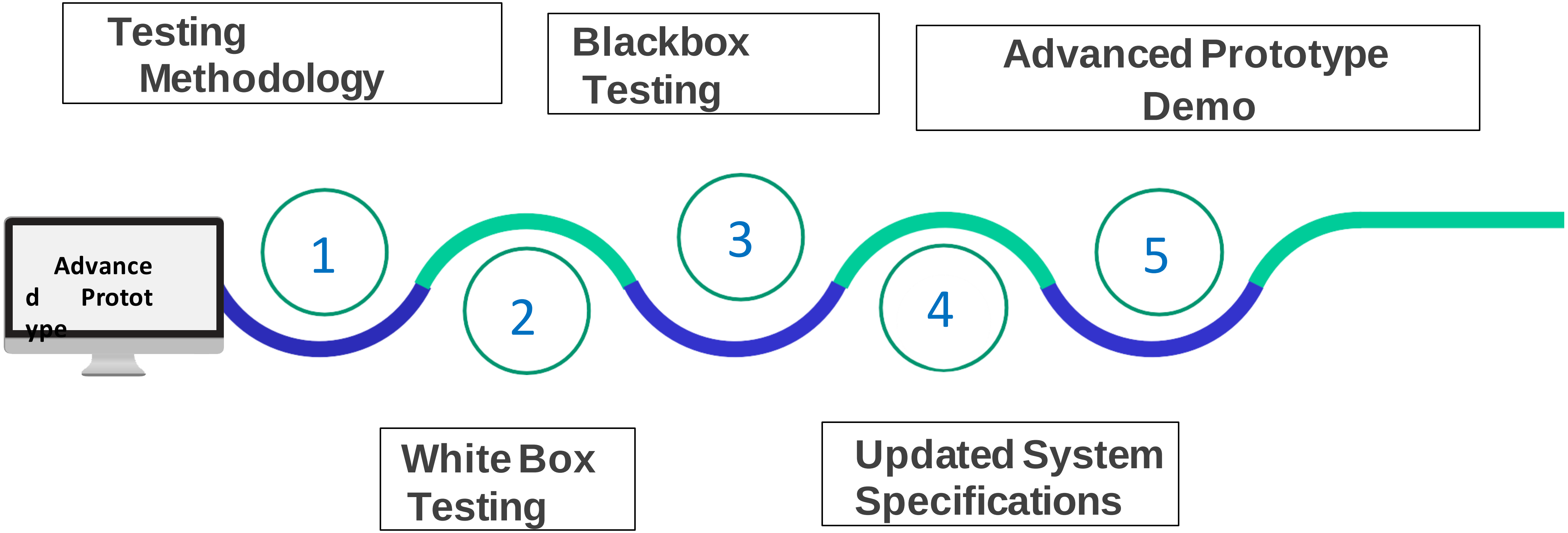
MIMRAZ HASSAN **(GIT Reports/ UI . Testing)**

MUHAMMAD FAIZAN ASLAM **(System Design /Blog Maintanence)**

VENKATA SUBBA RAJU KANUMURI **(Developer & Implementation)**

PRESENTED BY: **LAKSHMI HARIKA NAKKA**

Overview



Advanced Prototype

1 Testing Methodology

2 White Box Testing

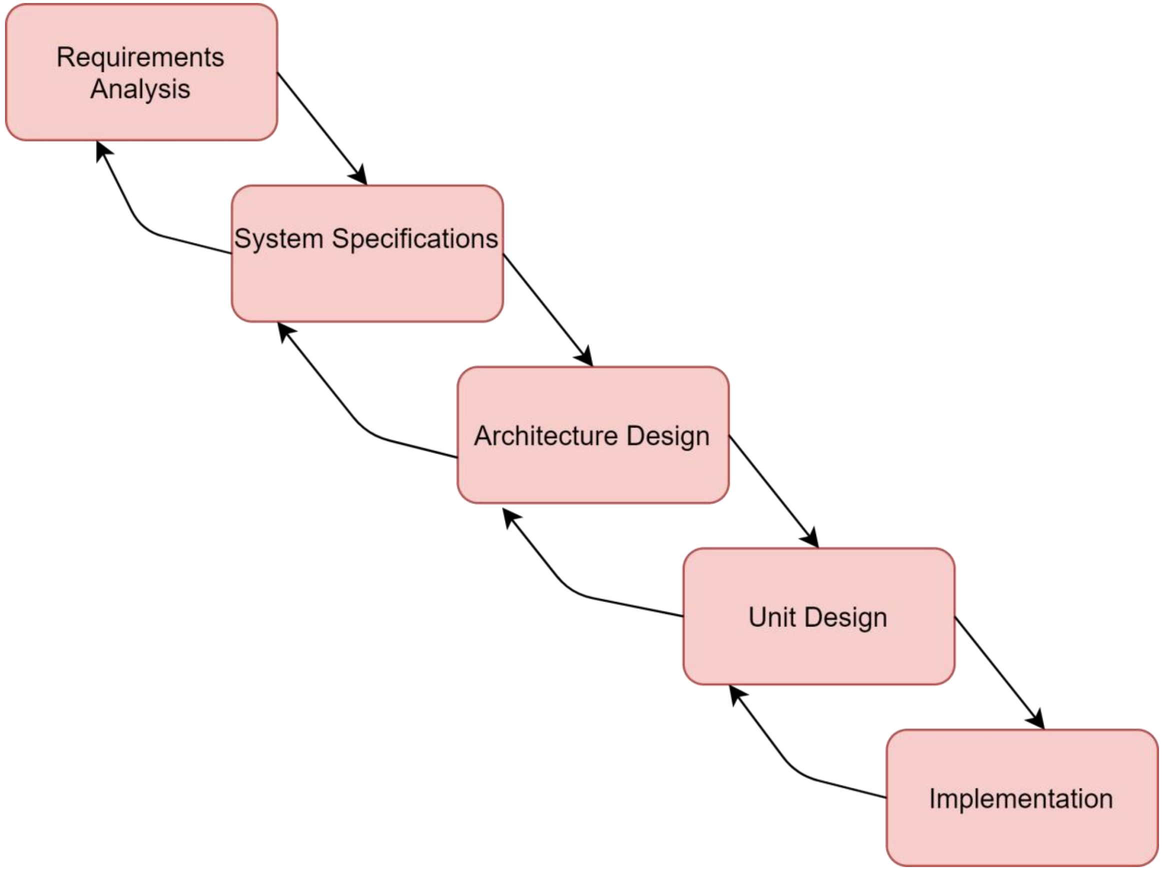
3 Blackbox Testing

4 Updated System Specifications

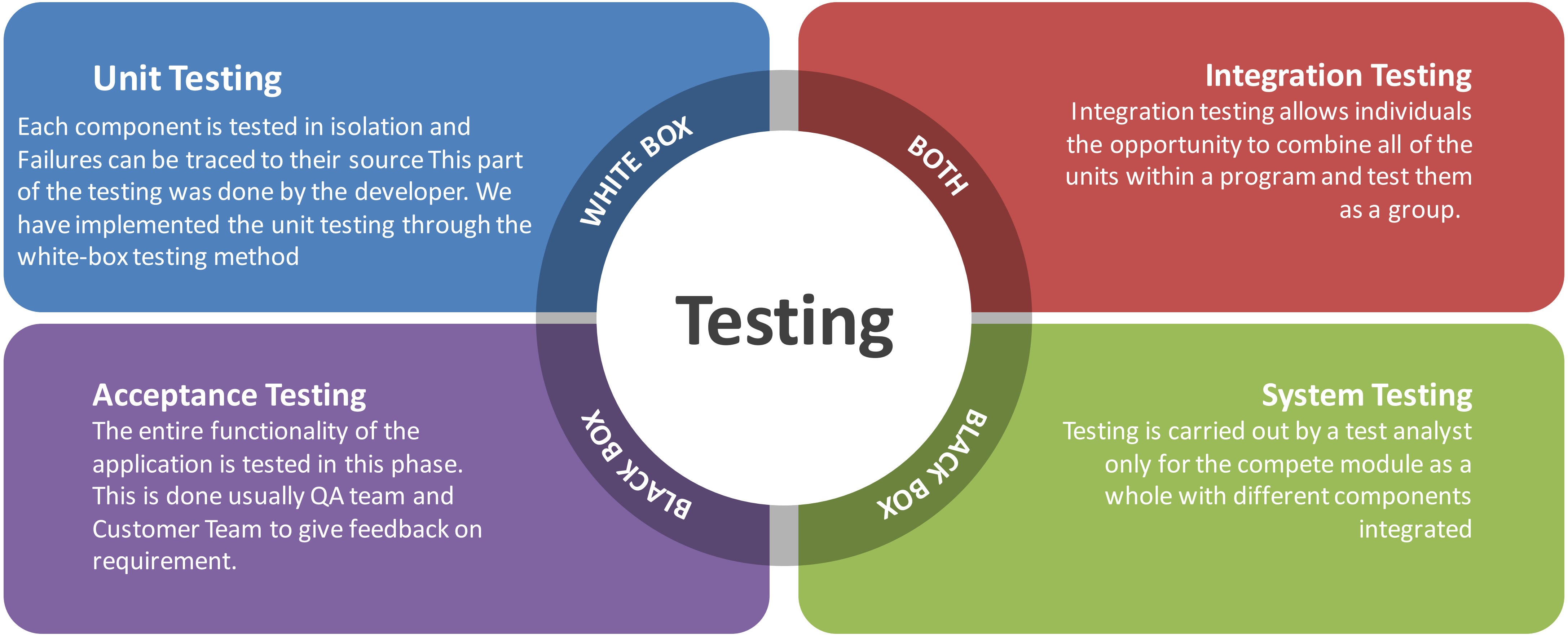
5 Advanced Prototype Demo

Static Testing

- Testing Naming Conventions
- Testing Access Modifiers
- Testing Reusability of code
- Optimizing Space and Complexity of Code
- Optimizing Runtimes



Testing Methodology



Testing

WHITE BOX

BOTH

BLACK BOX

BLACK BOX

Unit Testing

Each component is tested in isolation and Failures can be traced to their source This part of the testing was done by the developer. We have implemented the unit testing through the white-box testing method

Integration Testing

Integration testing allows individuals the opportunity to combine all of the units within a program and test them as a group.

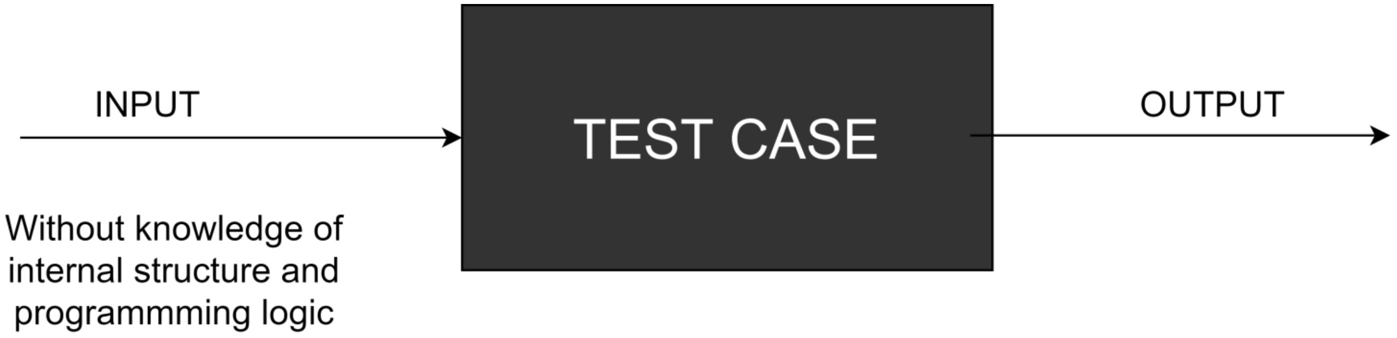
Acceptance Testing

The entire functionality of the application is tested in this phase. This is done usually QA team and Customer Team to give feedback on requirement.

System Testing

Testing is carried out by a test analyst only for the compete module as a whole with different components integrated

Black Box Testing


INPUT
TEST CASE
OUTPUT
Without knowledge of internal structure and programmming logic

The test **scenarios** and test **cases** were derived from the use case and tabulated

The **conditions** which will cause the execution were tabulated

Data **Values** were stored

Actual and **Expected** Result were listed

TEST SCENARIOS

SCENARIO_ID	RELEVANT USER STORY	TEST SCENARIO
TREASURE_TROVE_1	As a user I must be able to regiester and login to the application with a username and password	Validate that the user is abe to register and login with any email address successfully and invalid data is notified
TREASURE_TROVE_2	As a user I must be able to add different categories of transactions and add datae, amount and payments method.	Validate that the user is able to add all the entries successfully and invalid data in each of the fields resut in a warning motification
TREASURE_TROVE_3	As a user I should be able to filter and search the transaction according to my choice.	Validate that when transactions are filteed according to selection correct results are displayed
TREASURE_TROVE_4	As a user I should be able to export data from the application	Validate that all the relevant data is stored and exported from the application
TREASURE_TROVE_5	As a user I should be able to add the amount in multiple currencies	Validate that for each entry the user is able to adda custom currency in which he wants to store the data
TREASURE_TROVE_6	As a user I should be able to see a Contact form for support and queries	Validate that with the help form the user can contact the product owner. Validate that the email/notification for the contact form message is received my the product owner

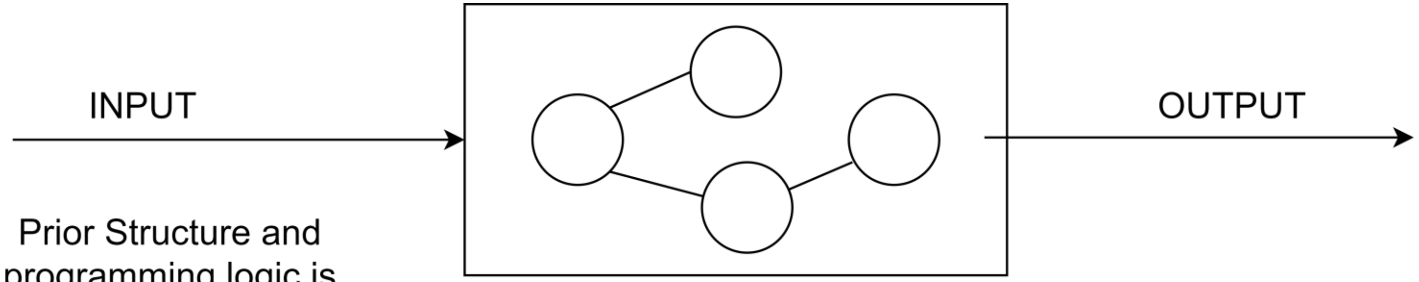
Test Cases(FUNCTIONAL & NON FUNCTIONAL

TESTCASE_ID	DESCRIPTION	CONDITIONS	DATA VALUES	EXPECTED RESULT	ACTUAL
TESTCASE_1	Validate the registration and login data	1. Open the application 2. Click on sign up 3. Enter full name 4. Enter email 5. Enter Password 6. Confirm the password	FullName. email, password, confirm password	Successfully register and sign in	Successful in most emails but invalid for certain data inputs of email
TESTCASE_2	Add all entries in order to successfully add the transaction	1. Open the application 2. Select the navigation drawer 3. Click on Add Expense 4. Add a category 5. choose a category 6. Enter the amount 7. Select Currency 8. Add title 9. Select date 10. Click on Add	1. Category = Appointment 2. Amount = 250 3. Currency=EUR. 4. Title=Rent 5. Date = 21/04/2021 6. Payment = Cash	Successfully add the transaction. Data entered message pops up	Successful
TESTCASE_3	Update/Delete Transaction	1. Open the application 2. Go to Transaction History 3. Select the transaction to edit 4. Edit the details 5. Choose Update/Delete option	1. Select a transaction 2.Change the category from Appointment to Food 3.Click on Update expense	Expense Updated and saved	Successful
TESTCASE_4	Export data	1. Open the application 2. Go to Transaction History 3. Click on the Export icon 4. Choose the required option (email, drive, device, etc.)	Select the download button on transaction Activity	Downloaded as an xl file locally	Successful
TESTCASE_14	Data in charts is displayed according to the logic and not overwritten	1. Open the application 2. Go to Transaction History 3. Click on the Chart Icon 4. Click on the Pie Chart	N/A	Data retrieved and represented logically and aesthetically	Overwriting of Data and Income savings to be seperated
TESTCASE_15	Data is loaded smoothly within the Activities	Scroll on Different Activities to check the loading and retrieval times of data	N/A	Data retrieved successfully	Successful
TESTCASE_16	Check UI elements structure on different screen sizes	Install app on different devices or use different custom emulator with different screens	N/A	UI Elements on all screens are balanced and consistent to the layout	Elements on Login and Addtransaction Activity are inconsistent

Failures, Issues and Bugs

TESTCASE	FAILURE	FAULT	RPOPOSED SOLUTION
TESTCASE_2	App crashes when invalid data such as 20 digit amount is added to the amount int the new category	Data is stored successfully but on retrieval from firebase it tends to crash	Adding a constraint of 20 digit in the amount to be added
TEST_CASE_14	The keyboard does not collapse when the transaction is added	The Keyboard hiding and displaying was not explicitly coded	Adding Keyboard hide Activity and Fragment
TEST_CASE_3	The data is not displayed when user is adding from a paused state	The data was displayed in the dashboard using Oncreate thus not resumed	Using OnResume Method to view data
TEST_CASE_11	Empty Screen when no transaction is added	Aesthetically not well defined user interface	A message is explicitly coded to show no transactions
TESTCASE_5	Filter View not showing accurate results	while using "==" to compare strings the logic does not work accurately	using ".equals" method
TEST CASE_7	Array Index out of bound error when displaying Bar chart for month January	An exception is being displayed	Add exception for the month of January
TEST_CASE_6	In the support form the contact number validation did not exist	Adding more than 11 or 2-3 digits did not invalidation warning	Explicit constraints in amount of digits in contact number were added

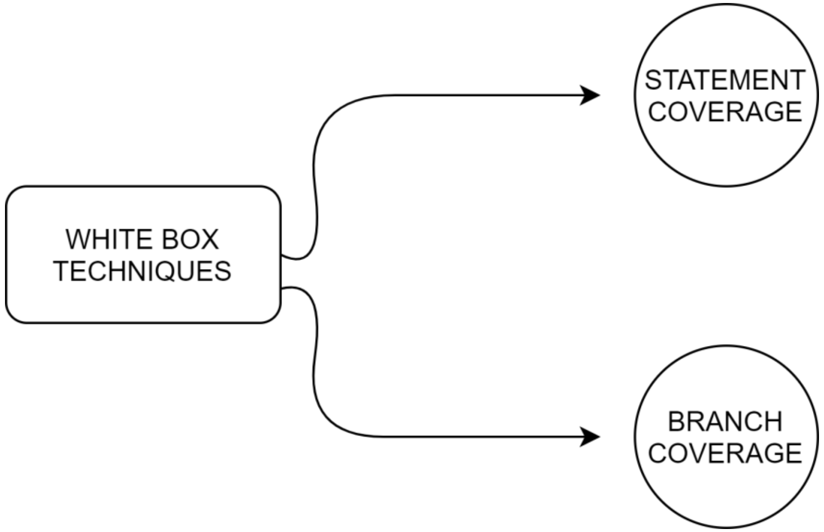
White Box Testing



INPUT

OUTPUT

Prior Structure and programming logic is to be known

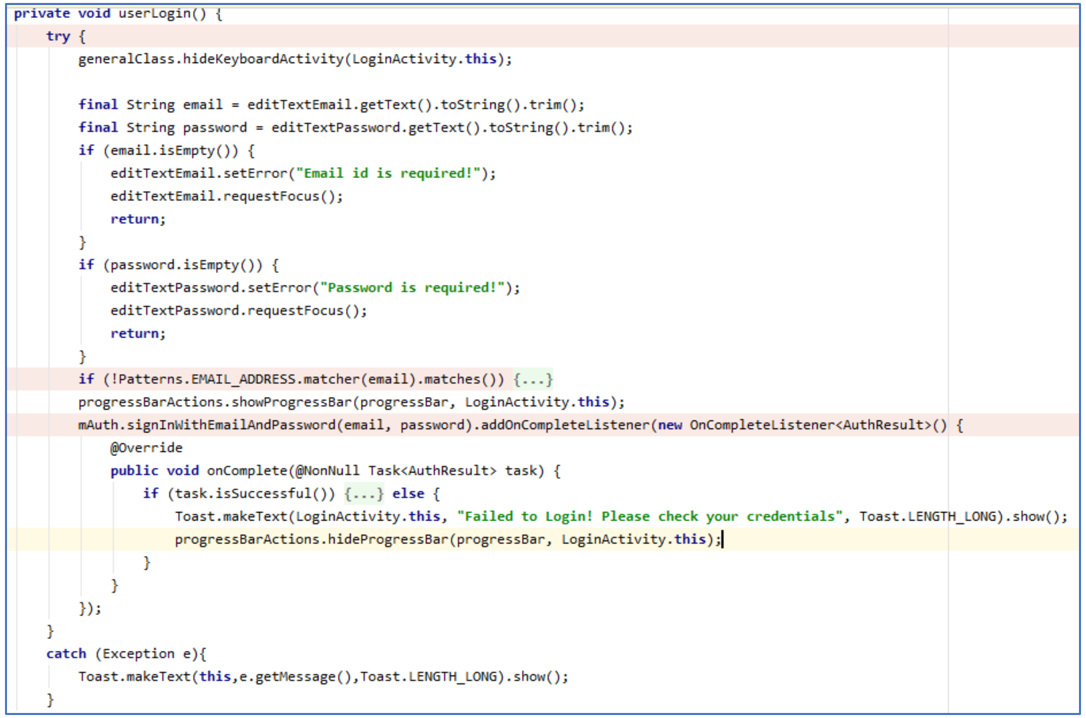


WHITE BOX TECHNIQUES

STATEMENT COVERAGE

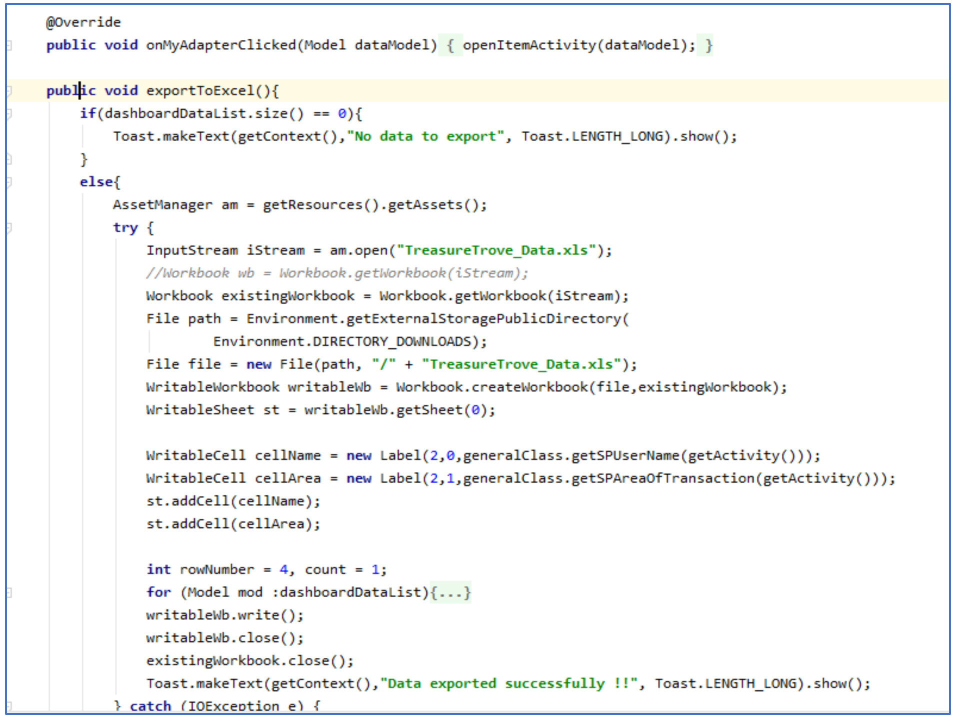
BRANCH COVERAGE

White Box Testing



```java
private void userLogin() {
    try {
        generalClass.hideKeyboardActivity(LoginActivity.this);

        final String email = editTextEmail.getText().toString().trim();
        final String password = editTextPassword.getText().toString().trim();
        if (email.isEmpty()) {
            editTextEmail.setError("Email id is required!");
            editTextEmail.requestFocus();
            return;
        }
        if (password.isEmpty()) {
            editTextPassword.setError("Password is required!");
            editTextPassword.requestFocus();
            return;
        }
        if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()) {...}
        progressBarActions.showProgressBar(progressBar, LoginActivity.this);
        mAuth.signInWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if (task.isSuccessful()) {...} else {
                    Toast.makeText(LoginActivity.this, "Failed to Login! Please check your credentials", Toast.LENGTH_LONG).show();
                    progressBarActions.hideProgressBar(progressBar, LoginActivity.this);
                }
            }
        });
    }
    catch (Exception e){
        Toast.makeText(this,e.getMessage(),Toast.LENGTH_LONG).show();
    }
```

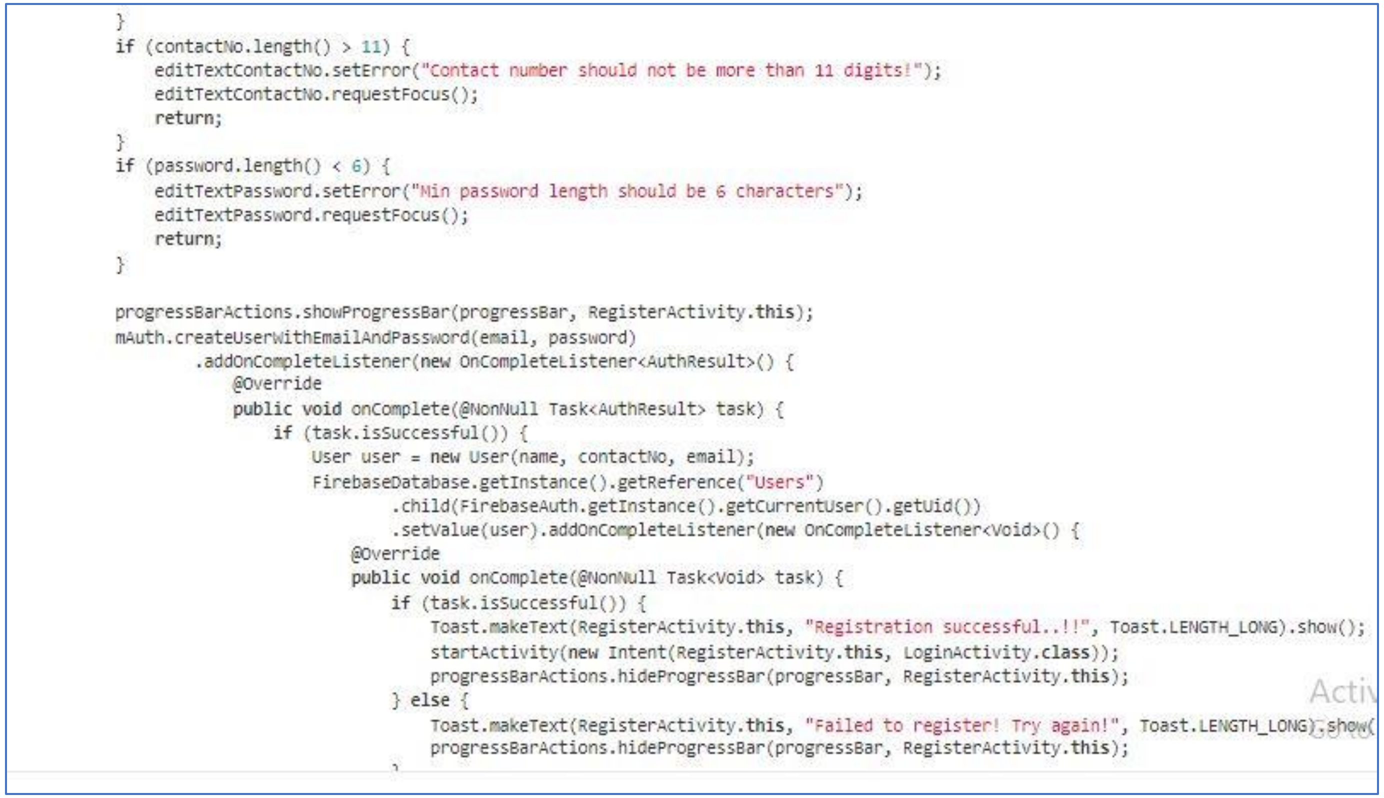
Login Validation



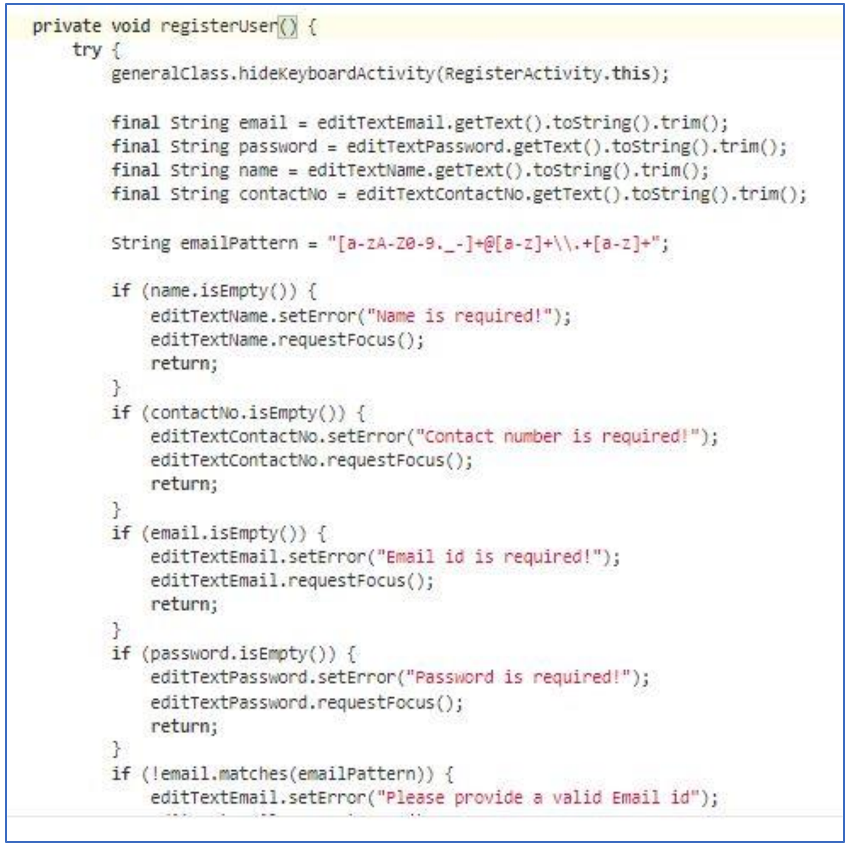
```java
@Override
public void onMyAdapterClicked(Model dataModel) { openItemActivity(dataModel); }

public void exportToExcel(){
    if(dashboardDataList.size() == 0){
        Toast.makeText(getContext(),"No data to export", Toast.LENGTH_LONG).show();
    }
    else{
        AssetManager am = getResources().getAssets();
        try {
            InputStream iStream = am.open("TreasureTrove_Data.xls");
            //Workbook wb = Workbook.getWorkbook(iStream);
            Workbook existingWorkbook = Workbook.getWorkbook(iStream);
            File path = Environment.getExternalStoragePublicDirectory(
                    Environment.DIRECTORY_DOWNLOADS);
            File file = new File(path, "/" + "TreasureTrove_Data.xls");
            WritableWorkbook writableWb = Workbook.createWorkbook(file,existingWorkbook);
            WritableSheet st = writableWb.getSheet(0);

            WritableCell cellName = new Label(2,0,generalClass.getSPUserName(getActivity()));
            WritableCell cellArea = new Label(2,1,generalClass.getSPAreaOfTransaction(getActivity()));
            st.addCell(cellName);
            st.addCell(cellArea);

            int rowNumber = 4, count = 1;
            for (Model mod :dashboardDataList){...}
            writableWb.write();
            writableWb.close();
            existingWorkbook.close();
            Toast.makeText(getContext(),"Data exported successfully !!", Toast.LENGTH_LONG).show();
        } catch (IOException e) {
```

Export Data Validation

White Box Testing



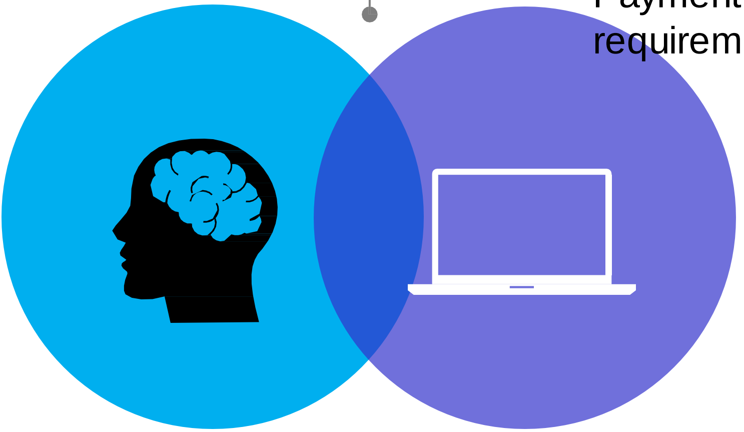
```java
        }
        if (contactNo.length() > 11) {
            editTextContactNo.setError("Contact number should not be more than 11 digits!");
            editTextContactNo.requestFocus();
            return;
        }
        if (password.length() < 6) {
            editTextPassword.setError("Min password length should be 6 characters");
            editTextPassword.requestFocus();
            return;
        }

        progressBarActions.showProgressBar(progressBar, RegisterActivity.this);
        mAuth.createUserWithEmailAndPassword(email, password)
                .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {
                            User user = new User(name, contactNo, email);
                            FirebaseDatabase.getInstance().getReference("Users")
                                    .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                                    .setValue(user).addOnCompleteListener(new OnCompleteListener<Void>() {
                                @Override
                                public void onComplete(@NonNull Task<Void> task) {
                                    if (task.isSuccessful()) {
                                        Toast.makeText(RegisterActivity.this, "Registration successful..!!", Toast.LENGTH_LONG).show();
                                        startActivity(new Intent(RegisterActivity.this, LoginActivity.class));
                                        progressBarActions.hideProgressBar(progressBar, RegisterActivity.this);
                                    } else {
                                        Toast.makeText(RegisterActivity.this, "Failed to register! Try again!", Toast.LENGTH_LONG).show(
                                        progressBarActions.hideProgressBar(progressBar, RegisterActivity.this);
```



```java
private void registerUser() {
    try {
        generalClass.hideKeyboardActivity(RegisterActivity.this);

        final String email = editTextEmail.getText().toString().trim();
        final String password = editTextPassword.getText().toString().trim();
        final String name = editTextName.getText().toString().trim();
        final String contactNo = editTextContactNo.getText().toString().trim();

        String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";

        if (name.isEmpty()) {
            editTextName.setError("Name is required!");
            editTextName.requestFocus();
            return;
        }
        if (contactNo.isEmpty()) {
            editTextContactNo.setError("Contact number is required!");
            editTextContactNo.requestFocus();
            return;
        }
        if (email.isEmpty()) {
            editTextEmail.setError("Email id is required!");
            editTextEmail.requestFocus();
            return;
        }
        if (password.isEmpty()) {
            editTextPassword.setError("Password is required!");
            editTextPassword.requestFocus();
            return;
        }
        if (!email.matches(emailPattern)) {
            editTextEmail.setError("Please provide a valid Email id");
```

Registration Validation

Updated Requirements



Necessary Requirements

- Improve Icons and Charts
- Improve UI Elements of Recurring Activity
- Payment Method now a necessary requirement

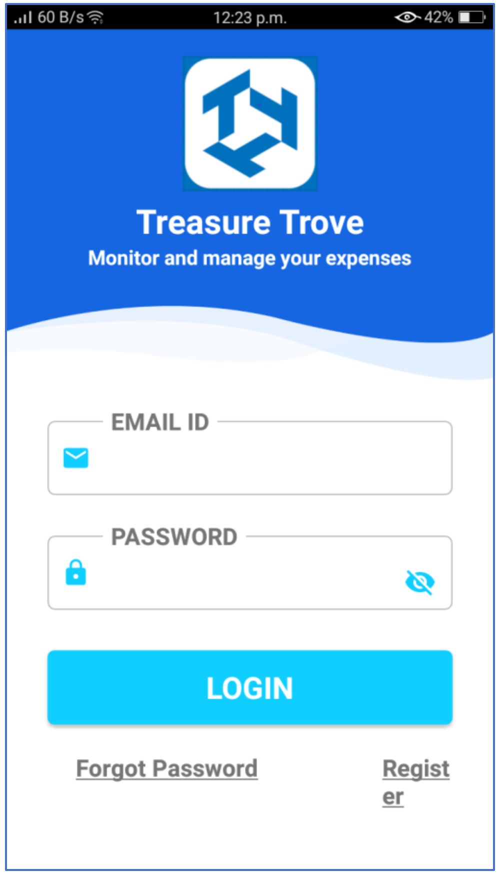
Essential Requirements

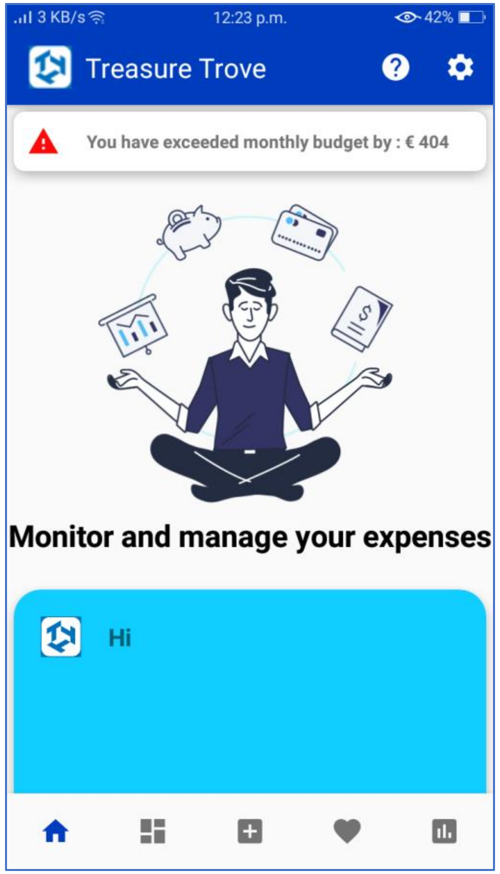
- Export of data locally
- Contact form
- Availability of Multiple Environments

Nice to have Requirements

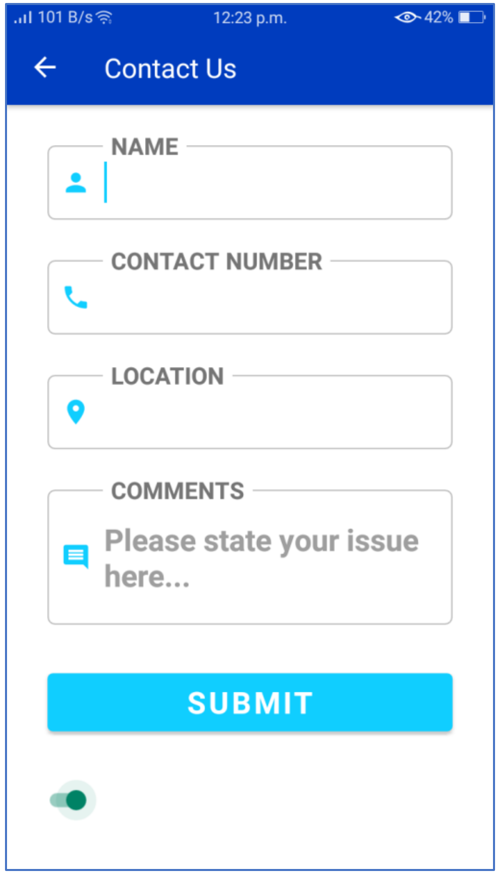
- Import data into .xl file formats
- Showing Multiple Currency and Conversions

Updated Design

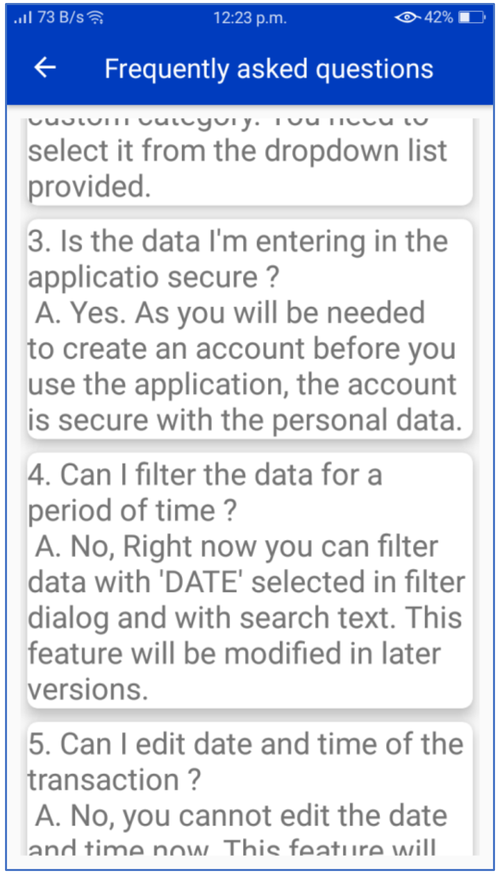




You have exceeded monthly budget by : € 404



Please state your issue here...



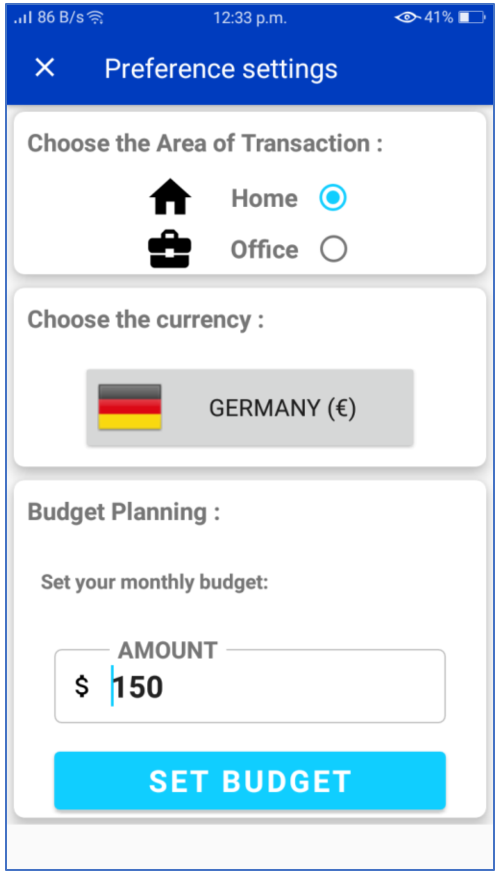
custom category. You need to select it from the dropdown list provided.

3. Is the data I'm entering in the applicatio secure ?
A. Yes. As you will be needed to create an account before you use the application, the account is secure with the personal data.

4. Can I filter the data for a period of time ?
A. No, Right now you can filter data with 'DATE' selected in filter dialog and with search text. This feature will be modified in later versions.

5. Can I edit date and time of the transaction ?
A. No, you cannot edit the date and time now. This feature will

Updated Design



Preference settings

Choose the Area of Transaction :

Home ●

Office ○

Choose the currency :

GERMANY (€)

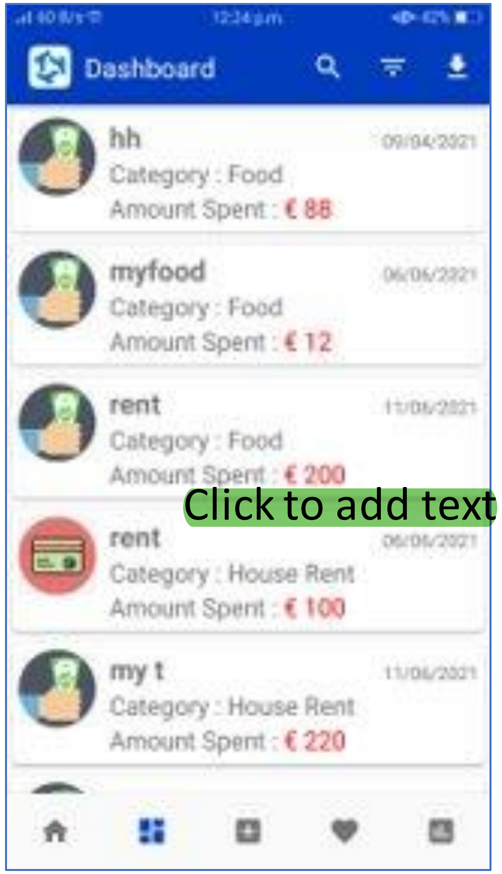
Budget Planning :

Set your monthly budget:

AMOUNT

$ 150

SET BUDGET



Dashboard

hh 09/04/2021
Category : Food
Amount Spent : € 88

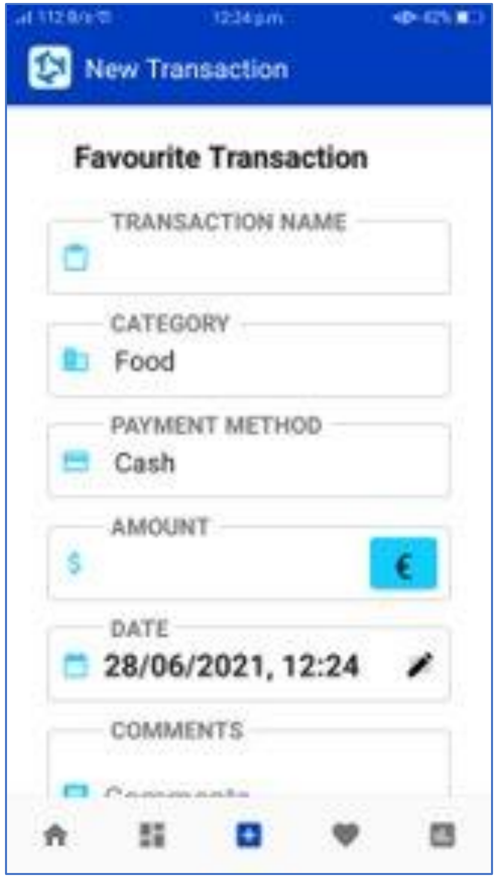
myfood 06/06/2021
Category : Food
Amount Spent : € 12

rent 11/06/2021
Category : Food
Amount Spent : € 200

rent 06/06/2021
Category : House Rent
Amount Spent : € 100

my t 11/06/2021
Category : House Rent
Amount Spent : € 220

Click to add text



New Transaction

Favourite Transaction

TRANSACTION NAME

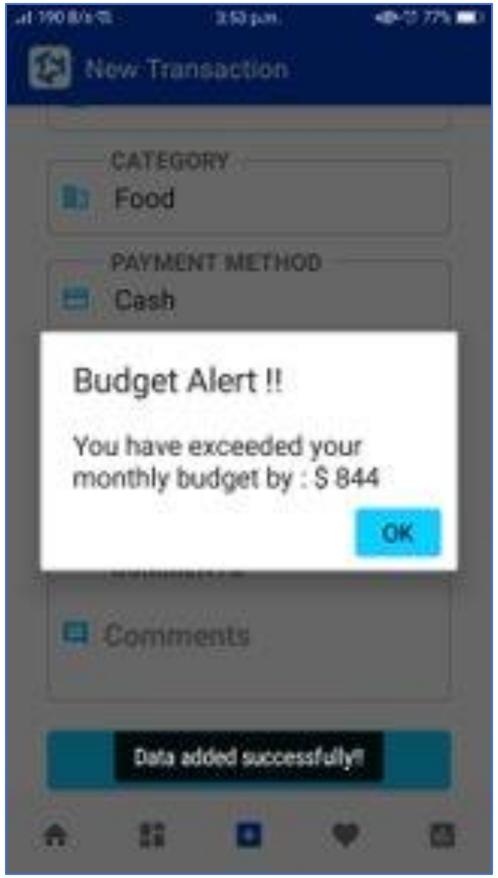
CATEGORY
Food

PAYMENT METHOD
Cash

AMOUNT
$
€

DATE
28/06/2021, 12:24

COMMENTS
Comments



New Transaction

CATEGORY
Food

PAYMENT METHOD
Cash

Budget Alert !!

You have exceeded your monthly budget by : $ 844

OK

Comments

Data added successfully!!

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF FAKULTÄT FÜR INFORMATIK

APPLICATION DEMONSTRATION…….

BY : LAKSHMI HARIKA NAKKA