

INTRODUCTION TO PROJECT

“Engineering” is a extremely common course took up by youth, by passion, influence, force and many more reasons. “B²E” which stands for Back Bench Engineers, is that the real struggle of facing exams, preparing for the exams(which students usually do it on the night before the exam) is known only for a backbencher alone.

Now, he/she has to crack the exam just the day before the exam, as the consequences are very devastating(backlogs, etc.) Also, some subjects or exams does not need a student to prepare for the whole syllabus, or all the modules and chapters. So we came with this application and our main motive of coming up with our application, is to completely serve the purpose of students last minute preparation of students. That doesn't mean, our content is not applicable to everyone, it obviously covers all the materials required for an excellent, average, and below average or slow learners. Proper study material is the most essential thing, with proper guidance to study the most important parts which would bring prosperous results for the students is the most important thing. For example, a person starting for an exam within a weeks span, cannot go around collecting all the materials required, which he might or might not get at the end, and even if he/she successfully accomplishes in compiling all the materials by then he would have already lost a lot of time in just acquiring, which instead they could invest for preparation, as “TIME” plays a major and vital role in these crucial times. One more thing to be noted is that, even after getting all the materials, we can observe people failing in examinations, mainly due to “lack of guidance on what to study”, because of which many extremely capable students end up having backlogs.

So, here after proper evaluation, and observation we have come up with question banks and revision sets which are derived after keenly going through the examination patterns, and the things which we ourselves implemented in our college days, and have been successful. We also bundled, links to the courses and certifications which are open at that particular time, and people enrolled would receive daily updates and notifications regarding the same, so that they are focused and enroll for their own benefit. What makes our idea standout against other counterparts is that, we have clearly observed and ourselves have experienced that, there is not a proper portal or application with most updated study materials according to the changes in syllabus and according to University norms and updating the batches and schemes content after every batch passout.

CHAPTER 2

REQUIREMENT ANALYSIS

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications. Requirements analysis is an important aspect of project management.

Flashcards exercise the mental process of **active recall**: given a prompt (the question), one produces the answer.

1. The application is used as alternative for physical flashcards which children can easily use to memorize easily.
2. In this app we have a set of questions and 2 options for them which can be selected.
3. Depending on the option selected a prompt message is displayed.
4. There is also a next button to navigate across the questions.
5. The answer with respect to every question is stored in the code. Hence if an incorrect answer is selected then an incorrect message alert is shown or else the correct message alert is displayed on choosing of correct option.
6. It helps the active recall process therefore it makes it easier to learn new concepts and revise them therefore it helps the users recall faster the concepts studied.

CHAPTER 3

SOFTWARE REQUIREMENT AND SPECIFICATION

Hardware Specifications

Computer Processor	Intel® Core™ i3 processor
Processor Speed	1.70 GHz Processor
Hard Disk	64GB
RAM	4.00 GB

Software Specifications

Operating System	Windows 10
Language used	Kotlin and Java
Development IDE	Android Studio

CHAPTER 4

IMPLEMENTATION

Java code

MainActivityController.java

```
package com.example.flashcards;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivityController extends AppCompatActivity {
    //create Buttons and TextView for UI communication
    private Button mTrueButton;
    private Button mFalseButton;
    private Button mNextButton;
    private TextView mQuestionTextView;

    //create the question bank
    private QuestionsModel[] mQuestionBank = new QuestionsModel[]{
        new QuestionsModel(R.string.question_java, true),
        new QuestionsModel(R.string.question_c, false),
        new QuestionsModel(R.string.question_python, true),
        new QuestionsModel(R.string.question_java2, false)
    };
    private int mCurrentIndex = 0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_controller);

        //calls to the UI (TextView)
        //displays the text
        mQuestionTextView = (TextView) findViewById(R.id.questions);
        //calls to the UI (Button)
        mTrueButton = (Button) findViewById(R.id.true_button);
        mTrueButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //will define method in a moment
                checkAnswer(true);
            }
        });
        mFalseButton = (Button) findViewById(R.id.false_button);
        mFalseButton.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
public void onClick(View v) {  
    checkAnswer(false);  
}
```

```

    }
    });

    mNextButton = (Button) findViewById(R.id.next_button);
    mNextButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mCurrentIndex = (mCurrentIndex + 1) % mQuestionBank.length;
            updateQuestion();
        }
    });
    //method declaration
    updateQuestion();
}
//method updateQuestion
//void so no return
private void updateQuestion() {
    int question = mQuestionBank[mCurrentIndex].getTextId();
    mQuestionTextView.setText(question);
}
private void checkAnswer(boolean userPress) {
    boolean answerTrue = mQuestionBank[mCurrentIndex].isAnswer();
    int messageResId = 0;
    if (userPress == answerTrue) {
        messageResId = R.string.correct_toast;
    } else {
        messageResId = R.string.incorrect_toast;
    }
    Toast.makeText(this, messageResId, Toast.LENGTH_SHORT).show();
}
}

```

QuestionsModel.java

```

package com.example.flashcards;

public class QuestionsModel {
    private int mTextId;
    private boolean mAnswer;

    //create a constructor
    public QuestionsModel(int textId, boolean answer) {
        mTextId = textId;
        mAnswer = answer;
    }
    //create the getter and setters for the constructor
    //these can be autogenerate as well
    public void setTextId(int textId) {
        mTextId = textId;
    }

    public void setAnswer(boolean answer) {
        mAnswer = answer;
    }

    public int getTextId() {
        return mTextId;
    }
}

```

```

    }

    public boolean isAnswer() {
        return mAnswer;
    }
}

```

XML FILES

Activity_main_controller.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="24dp"
        android:id="@+id/questions"/>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/true_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/true_button"/>
        <Button
            android:id="@+id/false_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/false_button"/>

    </LinearLayout>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/next_button"
        android:text="@string/next_button"/>

</LinearLayout>

```

String.xml

```
<resources>
    <string name="app_name">FlashCards</string>
    <string name="true_button">True</string>
    <string name="false_button">False</string>
    <string name="correct_toast">Correct</string>
    <string name="incorrect_toast">Incorrect</string>
    <string name="next_button">Next</string>
    <string name="question_java">Java is an Object Oriented language?</string>
    <string name="question_c">C is an interpreted language? </string>
    <string name="question_python">Python is an interpreted language?</string>
    <string name="question_java2">Java uses a malloc function?</string>

</resources>
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.flashcards">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.FlashCards">
        <activity android:name=".MainActivityController">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```


CHAPTER 6

TESTING

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. Testing is the process of executing a program with the explicit intention of finding errors that is making the program fail. The tester may analysts, programmer or a specialist trained for software testing, is actually trying to make the program fail. Analysts know that an effective testing program does not guarantee system reliability. Therefore, reliability must be designed into the system.

SL No	Description	Input	Expected output	Actual output	Remarks
1.	Given question	True or False option	If the correct option pressed then get correct prompt	Alert Observed as correct answer	Correct
2.	Given question	True or False option	If the incorrect option pressed then get incorrect prompt	Alert Observed as wrong answer	Incorrect

CHAPTER 7

SCREENSHOTS

Module 2 – INTRODUCTION TO COMPILERS

Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compiler, Applications of compiler technology.

Lexical Analysis: The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens.

Text book 2: Chapter 1 1.1-1.5, Chapter 3: 3.1-3.4

RBT: L1, L2, L3

2.1: Language Processors

Language Processors. An integrated software development environment includes many different kinds of language processors such as compilers, interpreters, assemblers, linkers, loaders, debuggers, profilers.

A *Compiler* is a program that can read a program in one language – the *source* language – and translate it to another language - the *target* language (Fig. 2.1).

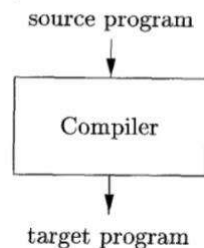


Fig. 2.1 A compiler

An important role of the compiler is to **report any errors** in the source program that it detects during the translation process.

If the target program is an executable machine-language program, it can be called by the user to process inputs and produce outputs. (Fig. 2.2)

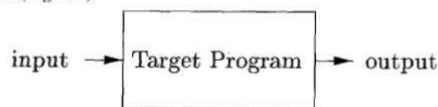


Fig. 2.2 Running the target program

2.1.1 Interpreter

An interpreter is another common kind of language processor. An interpreter appears to directly execute the operations specified in the source program on inputs supplied by the user (Fig. 2.3).

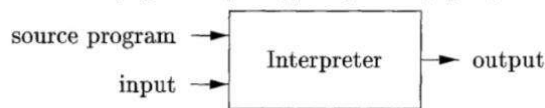


Fig. 2.3 An interpreter

2.1.2 Differences

System Software and Compiler - 18CS61 – Module 2 – Introduction (Compilers) - Notes prepared by Dr. Josephine Prem Kumar

1

- The machine-language target program produced by a compiler is usually much faster than an interpreter at mapping inputs to outputs.
- An interpreter, however, can usually give better error diagnostics than a compiler, because it executes the source program statement by statement.

2.1.3 Hybrid Compiler

Java language processors combine compilation and interpretation (Fig. 2.4).

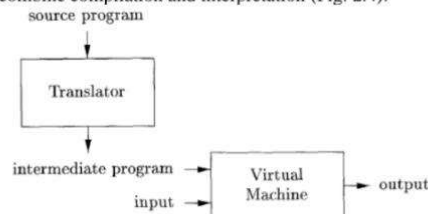
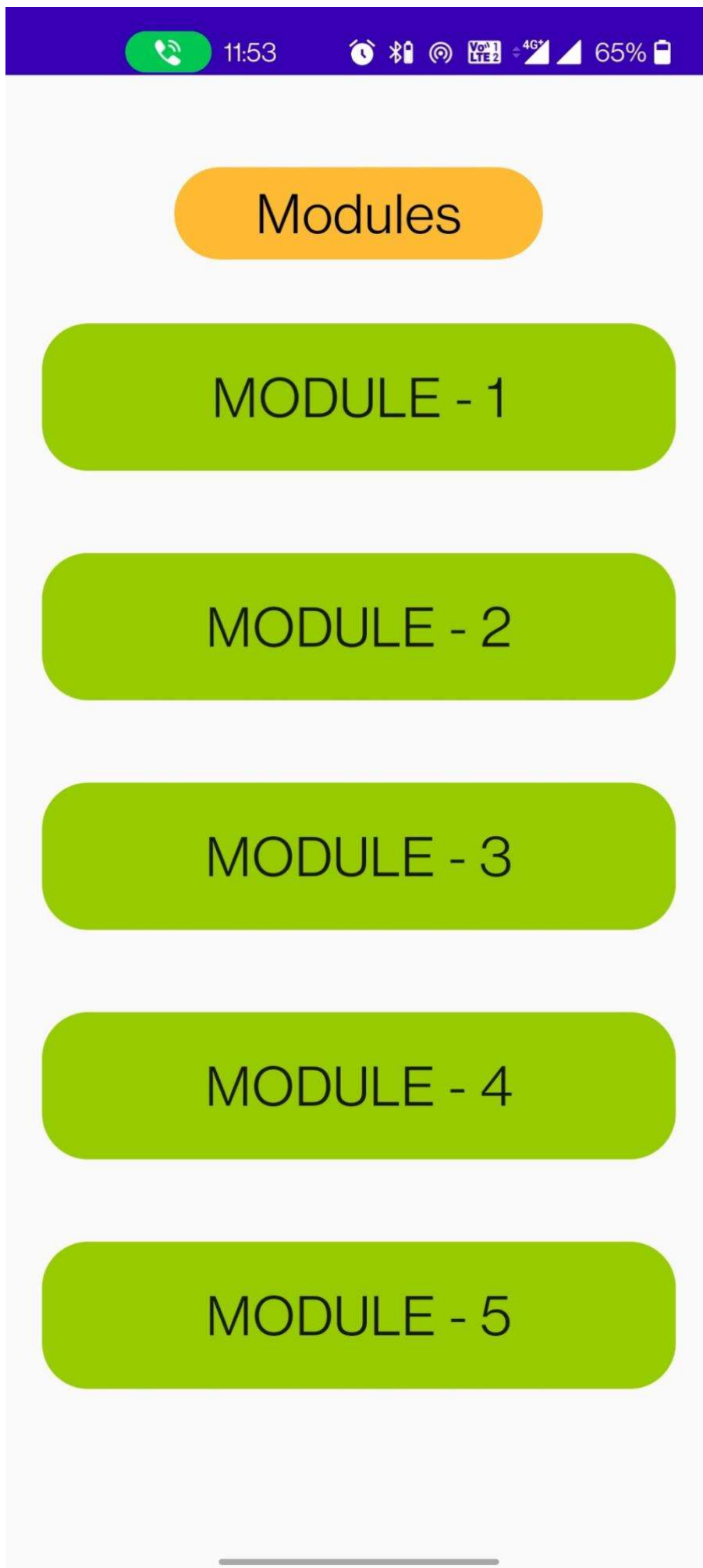


Fig. 2.4 A hybrid compiler

Java language processors combine compilation and interpretation. A Java source program may first be compiled into an intermediate form called *bytecodes*. The bytecodes are then interpreted by a virtual machine. The benefit is that bytecodes compiled on one machine can be interpreted on another machine.

In order to achieve faster processing of inputs to outputs, some Java compilers, called *just-in-time* compilers, translate the bytecodes into machine language immediately before they run the intermediate program to process the input. □





Branch

P-CYCLE

C-CYCLE

CSE

ISE

ECE

EEE

CIV

MECH



Branch



Circular



Stuff of the day



Trending Courses



Rate Us



About App



CONCLUSION

“Engineering” is an extremely common course taken up by youth, by passion, influence, force and many more reasons. “B²E” which stands for Back Bench Engineers, is that the real struggle of facing exams, preparing for the exams (which students usually do it on the night before the exam) is known only for a backbencher alone. Now, he/she has to crack the exam just the day before the exam, as the consequences are very devastating (backlogs, etc.) Also, some subjects or exams do not need a student to prepare for the whole syllabus, or all the modules and chapters.

So we came with this application and our main motive of coming up with our application, is to completely serve the purpose of students' last minute preparation of students. That doesn't mean, our content is not applicable to everyone, it obviously covers all the materials required for an excellent, average, and below average or slow learners. Proper study material is the most essential thing, with proper guidance to study the most important parts which would bring prosperous results for the students is the most important thing. For example, a person starting for an exam within a week's span, cannot go around collecting all the materials required, which he might or might not get at the end, and even if he/she successfully accomplishes in compiling all the materials by then he would have already lost a lot of time in just acquiring, which instead they could invest for preparation, as “TIME” plays a major and vital role in these crucial times. One more thing to be noted is that, even after getting all the materials, we can observe people failing in examinations, mainly due to “lack of guidance on what to study”, because of which many extremely capable students end up having backlogs.