**IMPLEMENTATION, TESTING AND MAINTENANCE**

**Introduction to Programming Languages, IDE’S, Tools and Technologies**

The following programming Languages, IDE’S, Tools and Technologies used for this android project ‘BashaBari’ Implementation.

**Programming Languages**

**Java:** As the project is developing an Android Application, the default programming language is Java. All Android applications are built using Java in Android Studio. Java is a popular and widely used language throughout the world. As mentioned in, Java is one of the powerful programming languages like C, C++. developed by Sun Microsystems which has many powerful features as described below. After the development of C, C++, Java has come into evolution by addressing their drawbacks. It is one of the open-source projects that could be easily installed in our machine. The language is also easy to learn, understand and implement. Java is used in various kinds of applications like Web, Desktop, Mobile, and Big Data. Many powerful features are supported by Java including various libraries, application services, graphics library for 2D/3D applications. The language is flexible enough to maintain code complexity, test, implementation, integration and support. Apart from these, there are other key features which make Java more special. It is object-oriented programming language, one of the important hierarchies in the programming languages which is used to implement real time applications, it provides for code reusability, it has a platform independence feature including any virtual machines, as in no need to write the many code for different OS as the Java Compliers convert the java source files to bytecode and this could be interpreted by any machine and the actual code is compiled irrespective of any machine, OS. It is more secured as the compilers are designed efficiently to figure out any kind of errors.

**IDE’S, Tools and Technologies**

**Android Studio**: Android Studio is exclusively designed for developing Android applications. It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app. The IDE is designed very efficiently which makes the developer’s job easy. It also supports the IntelliJ IDE, the main idea behind this IDE is that it automatically senses the variables, methods, classes, built-in functions or it could be anything else when we press the first letter of it. Say, suppose we declared few variables or methods that starts with an ‘S’, it automatically senses everything that starts with an ‘S’ and makes suggestions. It also supports Git as a version control system to maintain the app changes and push them into github. All java files, layout files (for design) are integrated into a single project easily. After the completion of project, the whole application could be put as an .APK (Android Package) file, in which we can run that APK file in any device and use the application. Other main tools include Android SDK, ADB, and Gradle Build.

**Android Software Development Kit (SDK):** One of the main tools used in developing android applications, as it packages many core features into one SDK and it can be used in the application easily. This helps us to avoid writing lot of code, and building applications faster.

**Gradle Build:** Gradle Scripts are the recent feature that is added to Android Studio. It is basically an automated build system which is used to automate the various phases involved in designing an application that includes design, development, test, debug, and publish. We need to configure the project and modules by mentioning all the supported jar files, SDK’s, version name, level, compiled SDK version, build tools version. to ensure that the developed app is compatible with the testing device/emulator. Gradle is also similar to Ant and Maven which helps in maintaining java projects (repositories).

**SDK Manager:** It is one of the main tools to maintain the updates of all the installed components required to run the project. It also notifies us when the project is not compatible with device or any other compatibility issues and to download any component that is required.

**AVD Manager:** It is used to create virtual devices of any desired API level to support higher level SDK’s incase our device does not support. Using emulators to test the application is difficult as it might be little slower when compared to real device.

**FireBase:** We have used a Firebase database which is an open-source database, easy to find in Google. In Firebase, we use a realtime database. We add a JSON file in our android apps which we get from our Firebase to create the database. The Firebase Realtime Database is a cloud-hosted NoSQL database that lets us store and sync data between our users in realtime. The Firebase database server could be registered by providing our Gmail account.

**Security and Permissions in Android**

Security notions in Android are quite high. Whenever a new Android Application is created, a unique user and group ID. This makes the maintenance of the application in an easier way to avoid any security or privacy issues. As the application is created uniquely, it becomes private and no one can access other’s applications. Permissions are another important concept which is included in AndroidManifest.XML configuration file. This is required if the application wants to access the external features.

If the application wants to access the Internet, Camera or it could be any feature, it requires permissions. It is included within the tags as it is an XML file. Permissions are automatically created for the basic applications at the time when we create the application. If the app uses higher level API or SDK we must explicitly mention the permissions inside uses-permissions tag to access the features or components.

**Testing Plan**

Test plan is necessary for any project to plan the testing phase and decide the project's scope. Test plan involves collecting design specifications about the project, adding test cases, executing them manually or automatically using automated testing tools. Testing any application is highly important. The test plan is a method of documenting the test cases, specification plans, and other basic level details about how the application works.

**1. Black Box testing:** In this project BashaBari, sample test cases manual testing is done to check the functionality of the application and focus on ***what is the output?***.

**2. White Box testing:** Once the application meets the user requirements and functionalities according to the test cases, its internal logic are completely tested to ensure that the application does not have any logical errors or issues. Basically, here we focus on the internal mechanism ***how the output is achieved?***.

**3. Unit Testing:** We have tested all the modules of the application individually by running a test program.

**4. Integration testing:** After testing the modules individually, tested them by integrating all the submodules, modules into one application.

**5. System Testing:** It refers to checking whether the system in which the application is built meets the necessary requirements like software support. For example: In this project, I have checked whether the device in which the application developed is compatible with the software (Android Studio) 29. Here, we also have security testing, recovery testing, performance testing.

**6. End to End Testing:** Tested the complete environment of the application by connecting the device with different machines, installing it as an APK file, with the database, and in the local network.

**7. Usability Testing:** Finally, usability testing is performed by testing the application’s flow, UI design, and how flexible and easy the application is easy to use.

**Test cases**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|
| 1 | Navigate to the Apps | | Apps should open | | As Expected | | | Pass | | |
| 2 | Enter email & Password for Login | | The credential can be entered | | As Expected | | | Pass | | |
| 3 | Click Login | | The user is logged in | | As Expected | | | Pass | | |
| 4 | Select menu bar and press any button | | All working | | As Expected | | | Pass | | |
| 5 | Click see more notice/request button | | Go to the more notice/ request page | | As Expected | | | Pass | | |
| 6 | Select payment method and confirm | | Go to the payment methods | | As Expected | | | Pass | | |
| 7 | Click pay bill | | Go to the payment page | | As Expected | | | Pass | | |

**Application Maintenance**

Apart from designing and developing the application, maintaining the application is one of the important characteristics. The developer/owner of the application should be concerned about the maintenance of the application by fixing the issues. The solution for fixing such type of issues when the application crashes by using any of the phone number that are available for mobile applications.