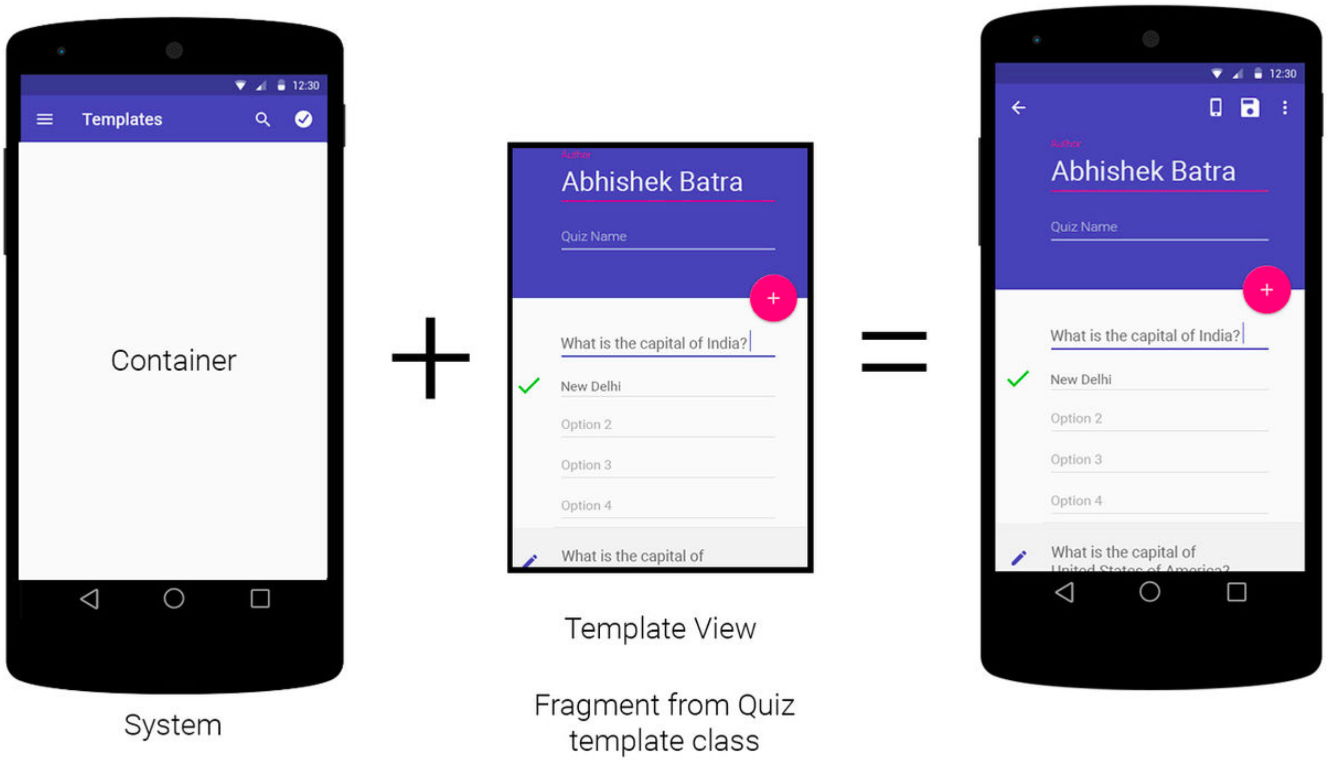
BuildmLearn
Toolkit

Architecture Design:

BuildmLearn toolkit for android will be based on plugin architecture. Each template will be a plugin, and toolkit app will be the system. Plugins will fit into the system. This will allow us to add more plugin (template) in the future without changing the functionality of the system. Moreover since BuildmLearn Toolkit is an open source software, thus it will be easy for contributors to write plugins rather than modifying the source code of the toolkit. Once the plugin is completed by a contributor, it can be tested by Toolkit moderator and if it works as expected it can be included into the toolkit with minimal lines of code by the moderator.



Hence from design point of view, toolkit will be divided into two modules:

1. System : A system will provide all the functionalities that are independent of Template plugin. Some of the functionalities are listed here.
    1. Save Project: Save an ongoing project into the device or server.
    2. Open Project: Open any saved project from the local storage or server.
    3. Publish APK: Publish apk generated by the template to BuildmLearn store.
    4. Sign APK: Sign the apk generated based on existing key provided by the user or create a new key, so that user can directly publish the app from his Google Play account.
    5. Zip Align APK: Zip align the signed apk.
    6. Navigate to BuildmLearn store: This will open the BuildmLearn store app if installed on user device or it will redirect the user to install BuildmLearn store app.
2. Base Plugin: Base plugin will be an interface containing the functionalities/modules to be provided by each template. The common functionalities/modules that must be provided by each template plugin will include.
    1. APK de-compile: Each template will be shipped with an apk of that template. The task of apk de-compile module will be to open the apk package and get the contents of assets folder.
    2. Validate: Validate the contents that are entered by the user. For example, if the template is Quiz, then Validate module will check if all the questions have an answer with four choices.
    3. APK compile: APK compile module converts all the user provided data into the structure required for assets file. Like, for flash card template, this will involve converting the images to Base64 and writing it to assets file. After assets file is successfully modified, next step will be packaging the apk.
    4. Simulator Fragment: BuildmLearn toolkit for android will have an inbuilt Simulator for testing the app. Each template must have an inbuilt Simulator logic. Simulator will just provide a container in which the app will be simulated. The container will allow to run a fragment inside it. This simulator logic should return an instance of the fragment.
    5. Plugin View: Each Template must provide a View (ie. fragment). This View will be displayed in ViewHolder provided by the System. The logic behind the View must be handled by the template itself. For example in quiz template, the view will be a fragment allowing the user to add, modify, delete and review questions.
    6. Snapshot: Create a snapshot of user workspace. This will be used by the System to save the file.
    7. Restore from snapshot: This will restore the state of user workspace back from a snapshot. This will be used to open a saved project.

These modules will be triggered by the System. The interface will be similar to this :

```
public interface BaseTemplate {

    public void initialize();

    public void apkDeCompile();
```

```
        public boolean apkRecompile(String data);

        public Fragment getFragmentForSimulator(Context context);

        public void setIcon(Bitmap icon);

        public void setAuthor(String author);

        public String getAuthor();

        public boolean saveApk(String path);

        public boolean validate();

        public Snapshot getSnapshot();

        public void restoreFromSnapshot(Snapshot snap);

    }
```

Keys (for understanding)

The following keys/words will be frequently used in this template.

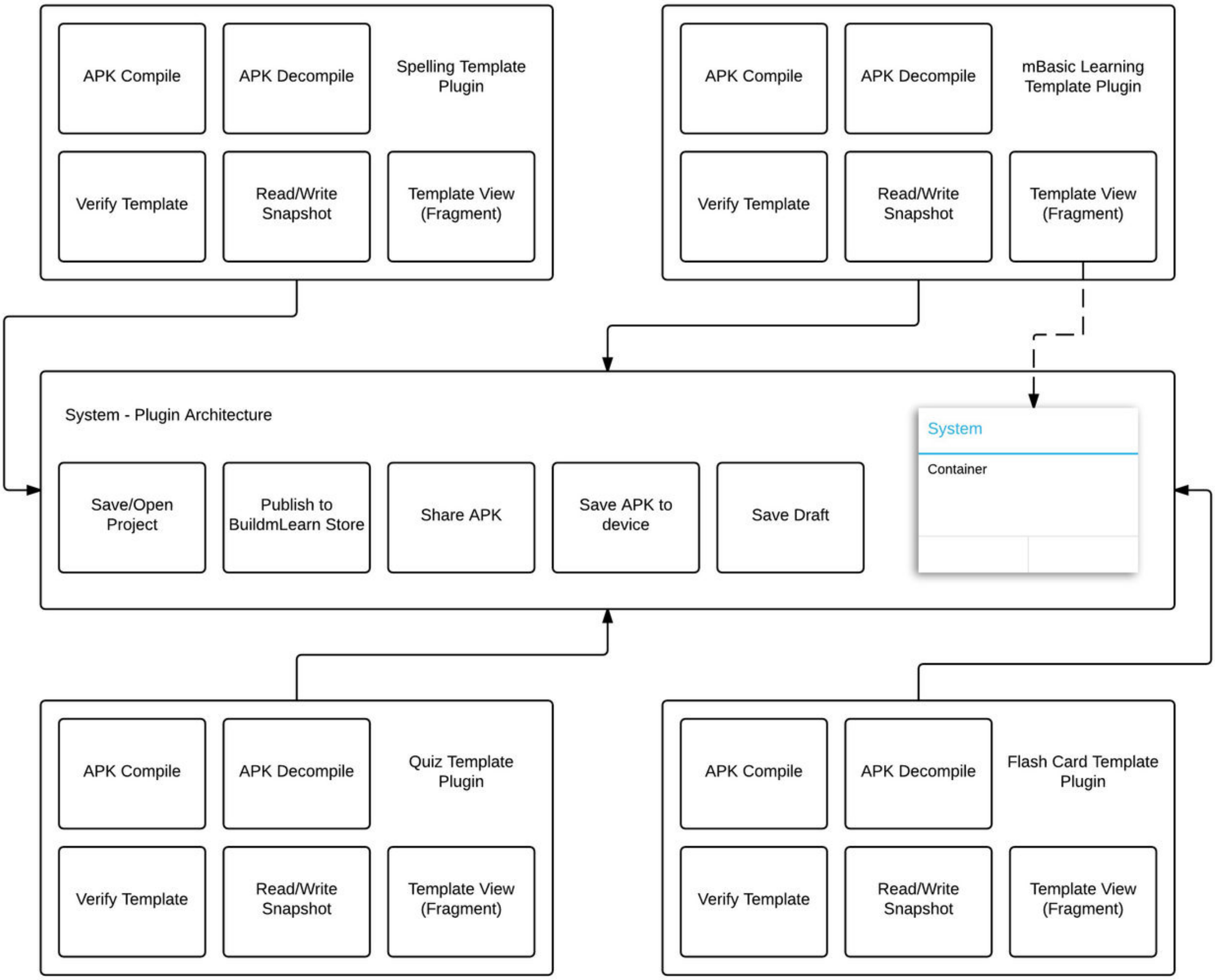
System: Whole App is referred as a system in this proposal. System will allow plugins to be attached without much effort.

Plugin: Each plugin referred here is a template implementation. Quiz template will be a form of plugin

Template Class: Each template will be class that will implementing Base Plugin Interface. Hence, each template will provide all the functionalities from the Base Plugin. In this proposal Template class is a type of template. (Can be quiz template, flash card template etc)

Snapshot: Snapshot is the data entered by the user. Snapshot will follow a schema of the .buildmlearn file.

Implementation



System Development

System Development will consist of following task:

1. Open/Save Project: This task will allow saving project and reading from a project file. Saving project will consist of taking a snapshot of data entered by the user and saving it in a .buildmlearn file on device storage. Snapshot will be provided by the template class, here the only task will be to save snapshot on phone storage. Opening a project will consist of parsing a .buildmlearn file and checking the type of template data saved and passing the snapshot saved to an object of that template, to restore user workspace.
2. Publish to BuildmLearn store: This task will send .buildmlearn project file along with other meta data to the BuildmLearn REST API. .buildmlearn file will be generated from the final Snapshot provided by the Template class.
3. Share: Share task will allow various way of sharing app including gmail, bluetooth etc. This is of great importance for teachers preferring app distribution in a small group.
4. Save APK: This task will get the apk from template class and save the apk on device storage.
5. Instantiation of object of Template class: Whenever the user creates a new project, or open a project, this task will inflate the View (fragment) from the Template class. User will be interacting with this View only for adding question data.

Template Development

This will involve writing template logic for all the four current templates. Each Template will be a class say QuizTemplate.java, FlashCardTemplate.java, BasicmLearningTemplate.java, LearnSpellingsTemplate.java. Each of this class will implement BasePlugin interface (See base plugin in Architecture design). The two important modules of the template are explained here:

1. Simulator Fragment : This module will return a fragment. The Fragment will be the starting view of the simulation. The present template apk source needs to be converted into fragment for simulation. Thus this is straightforward and won't require much effort to convert.
2. PluginView : This module will return a fragment that will be inflated into System View Holder. User will be interacting with this fragment for entering data (NOT THE SYSTEM).

System will work on the objects of Template Class

Simulator

BuildmLearn toolkit will support simulation in two modes: Landscape and Portrait. The main goal here is to simulate the app as close as possible to the real device.

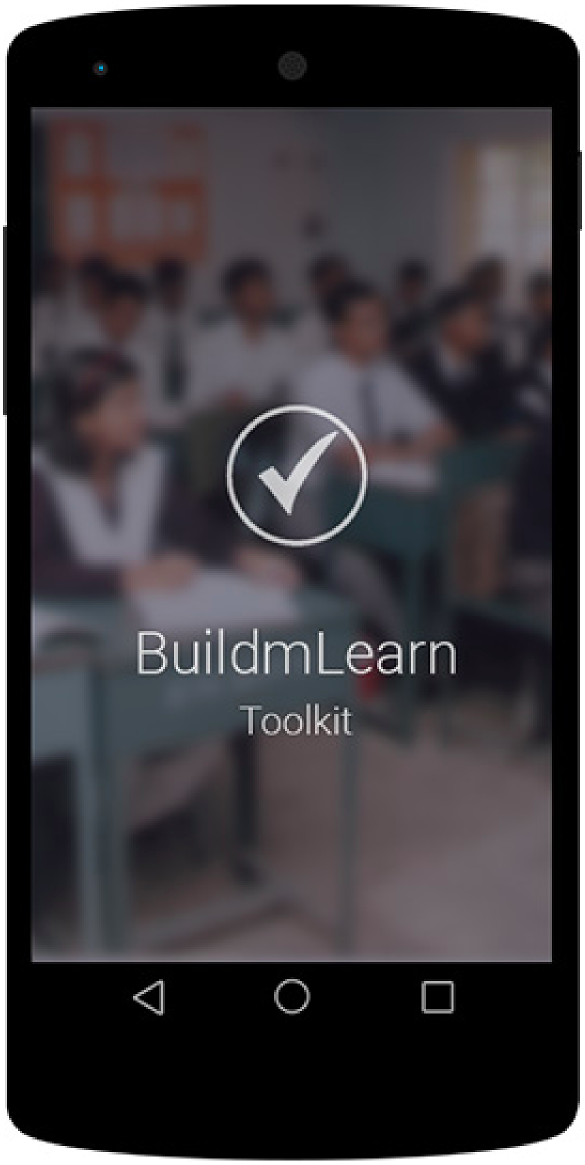
Implementation of Simulator:

Simulator will be an activity and will feature a complete mobile layout of Nexus 6. The screen of the layout will be defined as FrameLayout. This framelayout will be the view holder for simulation. The Simulator Fragment returned by the Template class will be inflated in this ViewHolder.

Also, Simulator will have a functional Navigation pad. This will give similar experience of a real device.

UI Designs/Wireframes

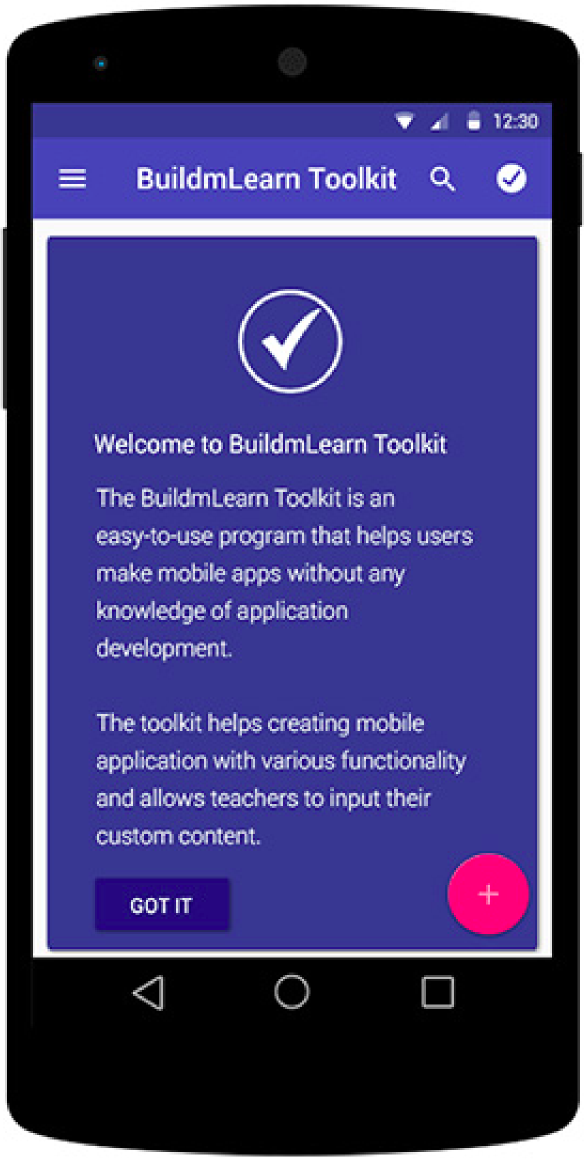
Splash Screen



Any data that is to be feteched from the inernet will be done during the splash screen. For example change logs, news etc.

Home Screen

Home Screen will show information related to BuildmLearn Toolkit, update logs etc.



Welcome to BuildmLearn Toolkit

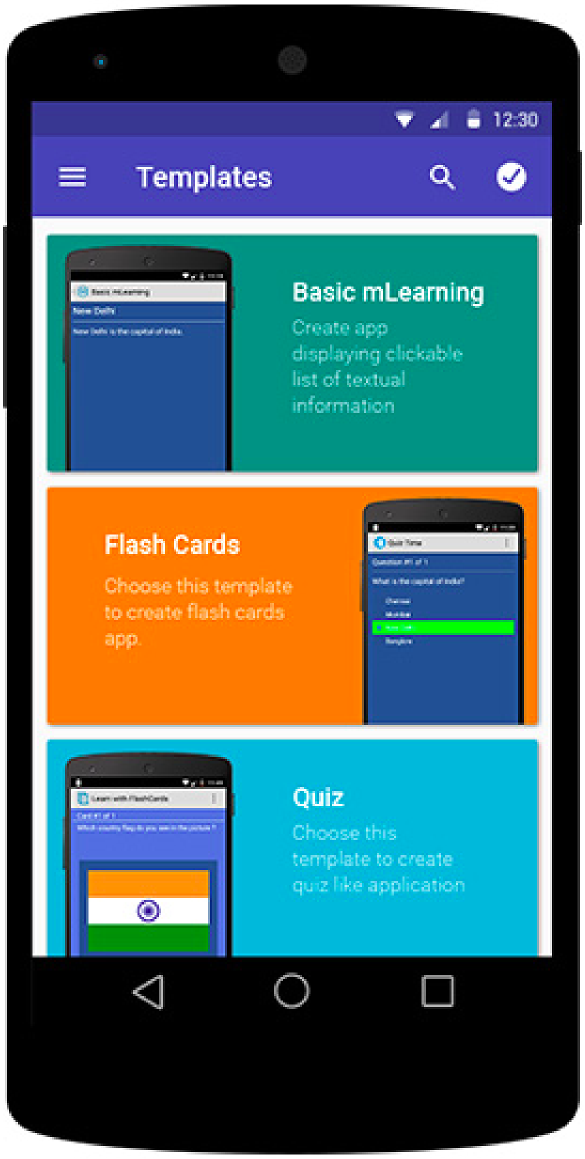
The BuildmLearn Toolkit is an easy-to-use program that helps users make mobile apps without any knowledge of application development.

The toolkit helps creating mobile application with various functionality and allows teachers to input their custom content.

Clicking on the floating button (pink circular button) will navigate the user to template activity from where the user can select template. This activity will also show Navigaation Drawer

Template Screen

This activity will be shown when floating button in the home screen is clicked. This screen will allow the user to select the template



Basic mLearning

Create app displaying clickable list of textual information

Flash Cards

Choose this template to create flash cards app.

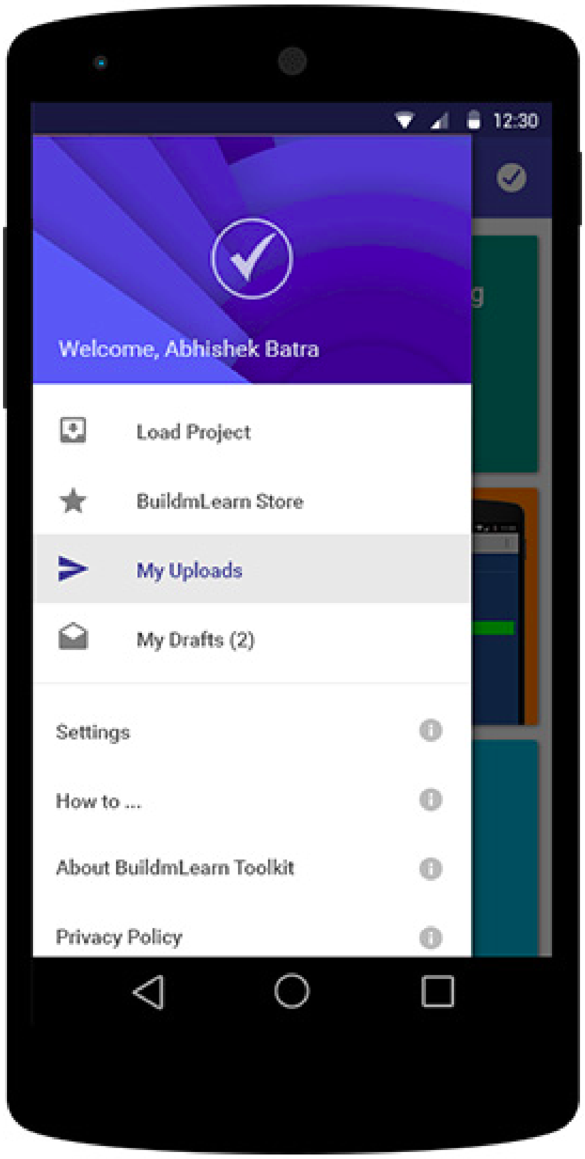
Quiz

Choose this template to create quiz like application

ListView showing each template as one row. Clicking on any template will launch the corressponding editor for that template.

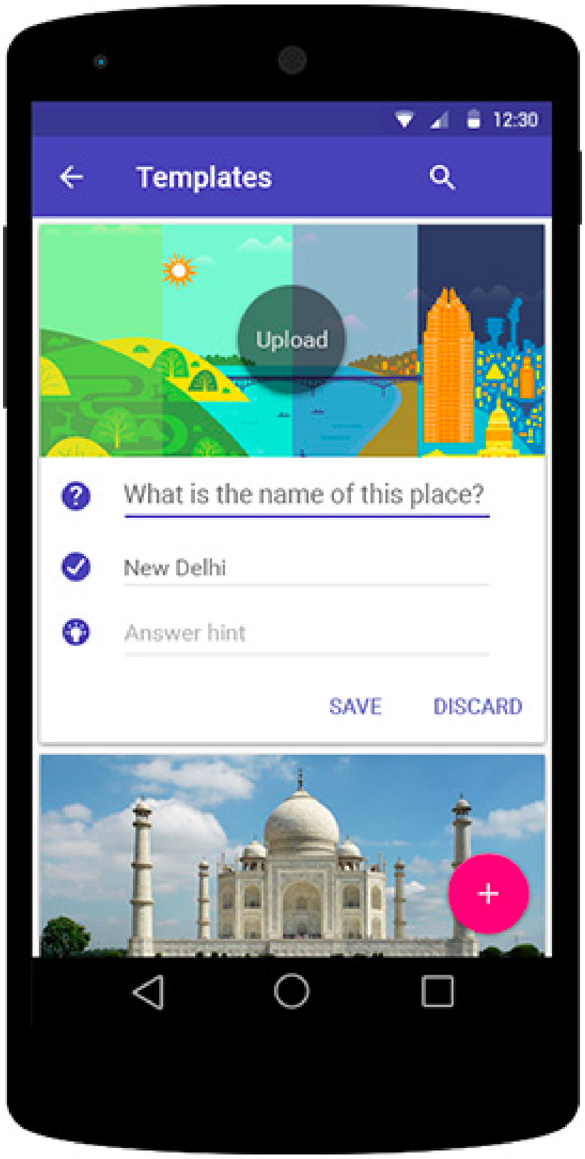
Navigation Drawer

Navigation Drawer will consist of menu items such as Load Project, Store and Settings.



Naviagtion Drawer will be visible when the user swipes the screen to the right and will be accessible from Home Screen

Flash Card Template : Edit

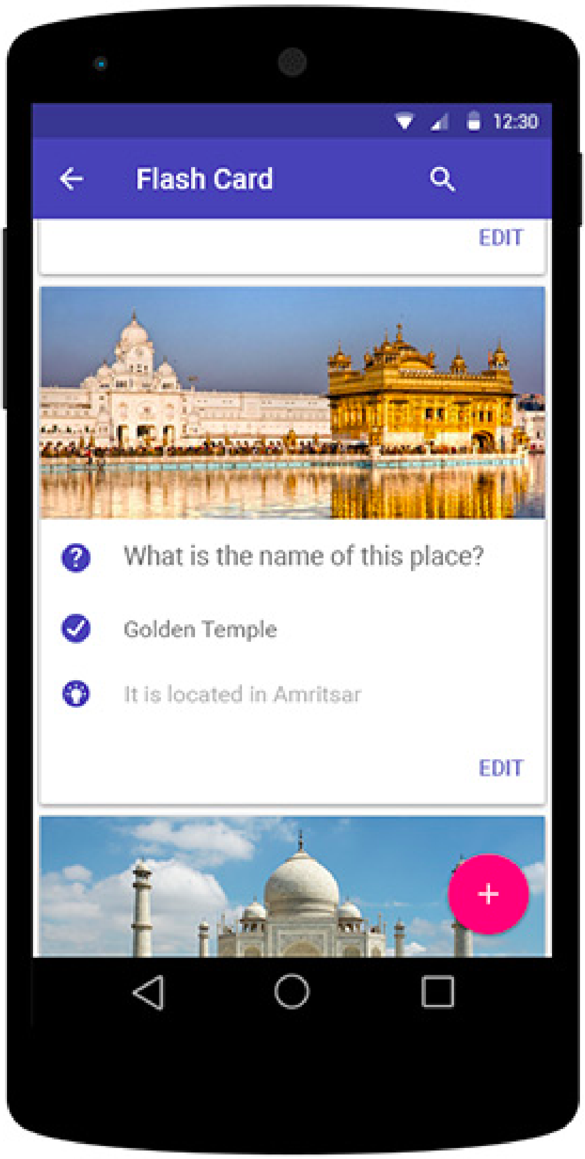


What is the name of this place?

New Delhi

Answer hint

Flash Card Template : View



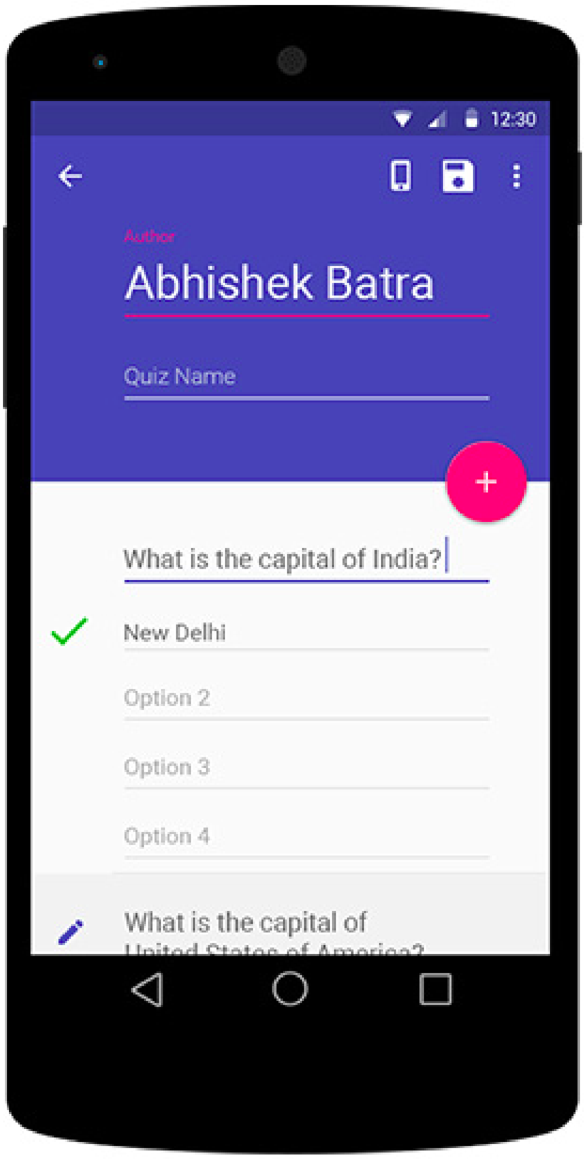
What is the name of this place?

Golden Temple

It is located in Amritsar

Following screen shows Flash card template in edit and view mode. In edit mode user can upload an image or take a new image from camers. Clicking on Pink button will add a new flash card.

Quiz Template



Author

Abhishek Batra

Quiz Name

What is the capital of India?

✓ New Delhi

Option 2

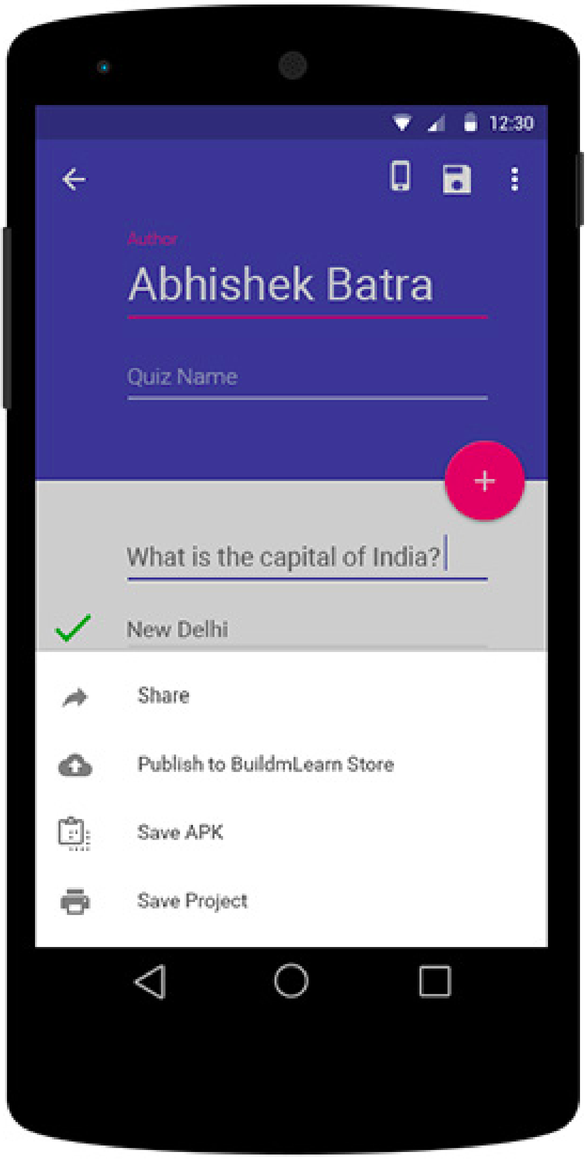
Option 3

Option 4

What is the capital of United States of America?

Author and Quiz name section is scrollable and they will take place as ListView header.

Publish Options



Author

Abhishek Batra

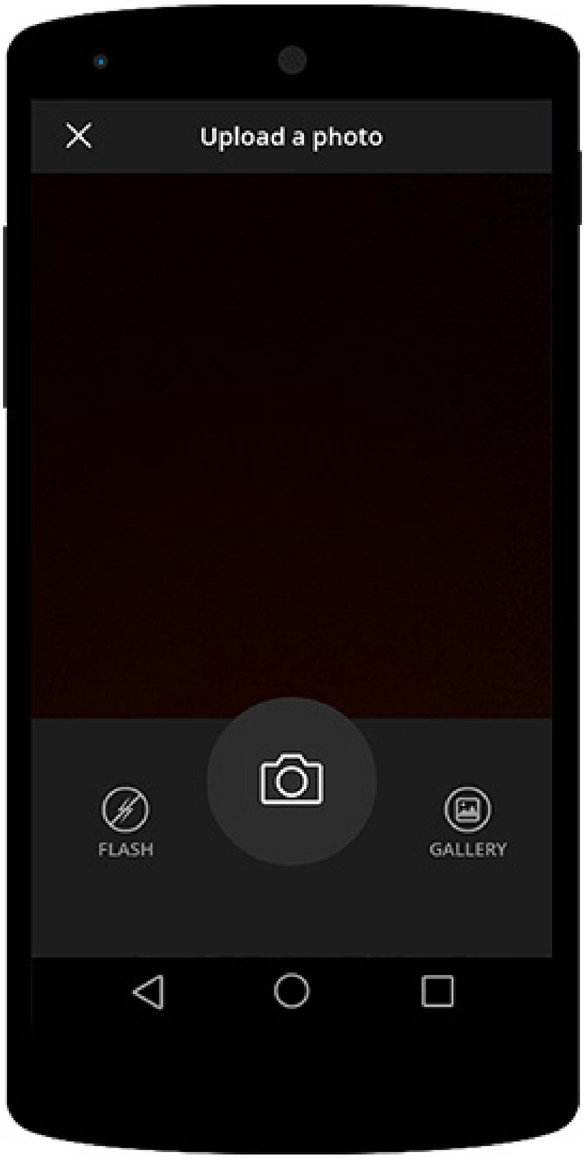
Quiz Name

What is the capital of India?

✓ New Delhi

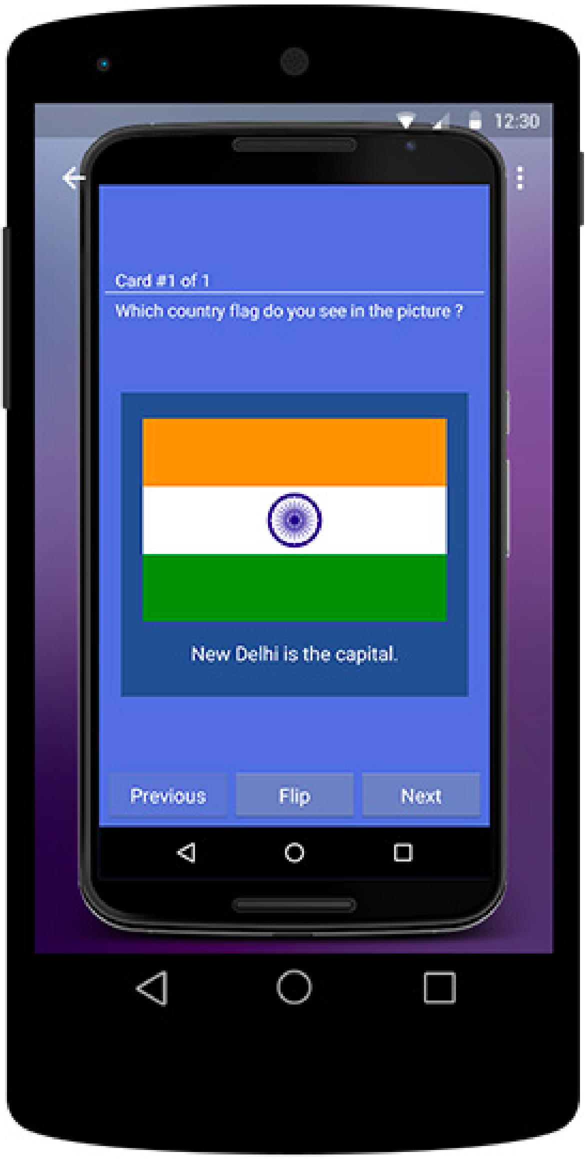
Publish options will consist of sahre apk, publishing to BuildmLearn Store, saving apk, and saving project

Custom Camera View



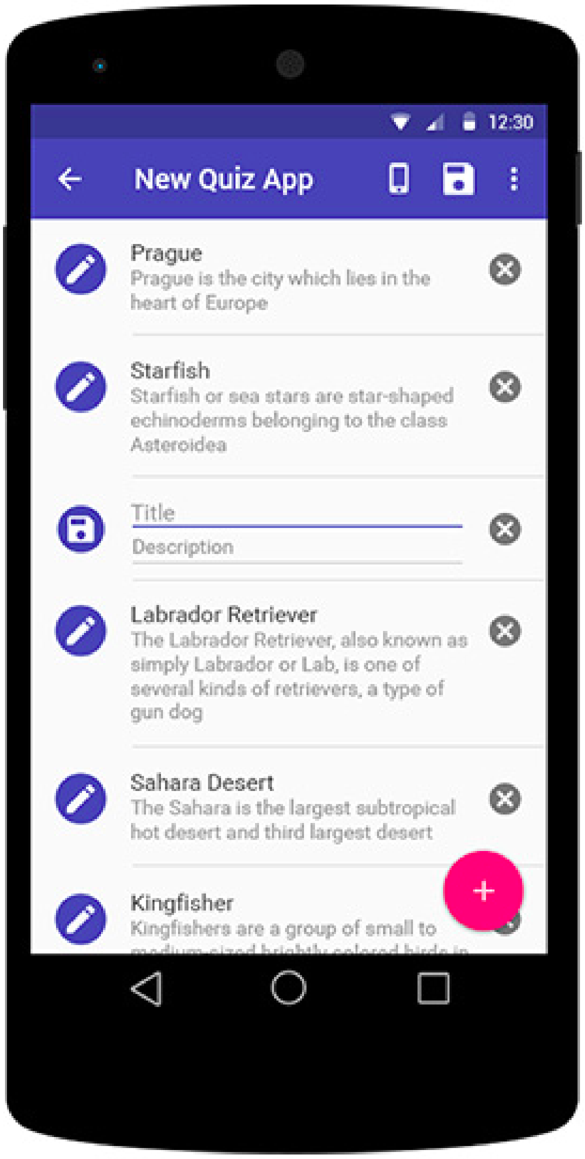
Custom camera view for taking photos or selecting images from gallery for flash card template. Design inspiration of this view taken from Zomato App.

Simulator


12:30
Card #1 of 1
Which country flag do you see in the picture ?
New Delhi is the capital.
Previous
Flip
Next

Simulator design that will be built-in in the toolkit.

Basic mLearning/Learn Spelling Template


12:30

New Quiz App

Prague
Prague is the city which lies in the heart of Europe

Starfish
Starfish or sea stars are star-shaped echinoderms belonging to the class Asteroidea

Title
Description

Labrador Retriever
The Labrador Retriever, also known as simply Labrador or Lab, is one of several kinds of retrievers, a type of gun dog

Sahara Desert
The Sahara is the largest subtropical hot desert and third largest desert

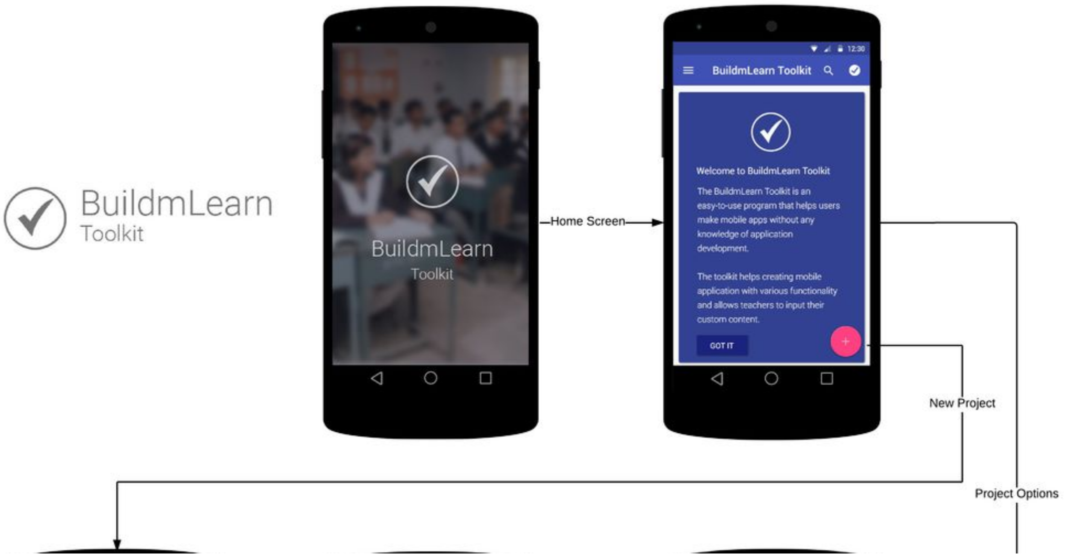
Kingfisher
Kingfishers are a group of small to medium-sized brightly colored birds in

Plus Button (Pink button) will add new item.

Each row is one item

Clicking the edit button will allow the user to edit the item.

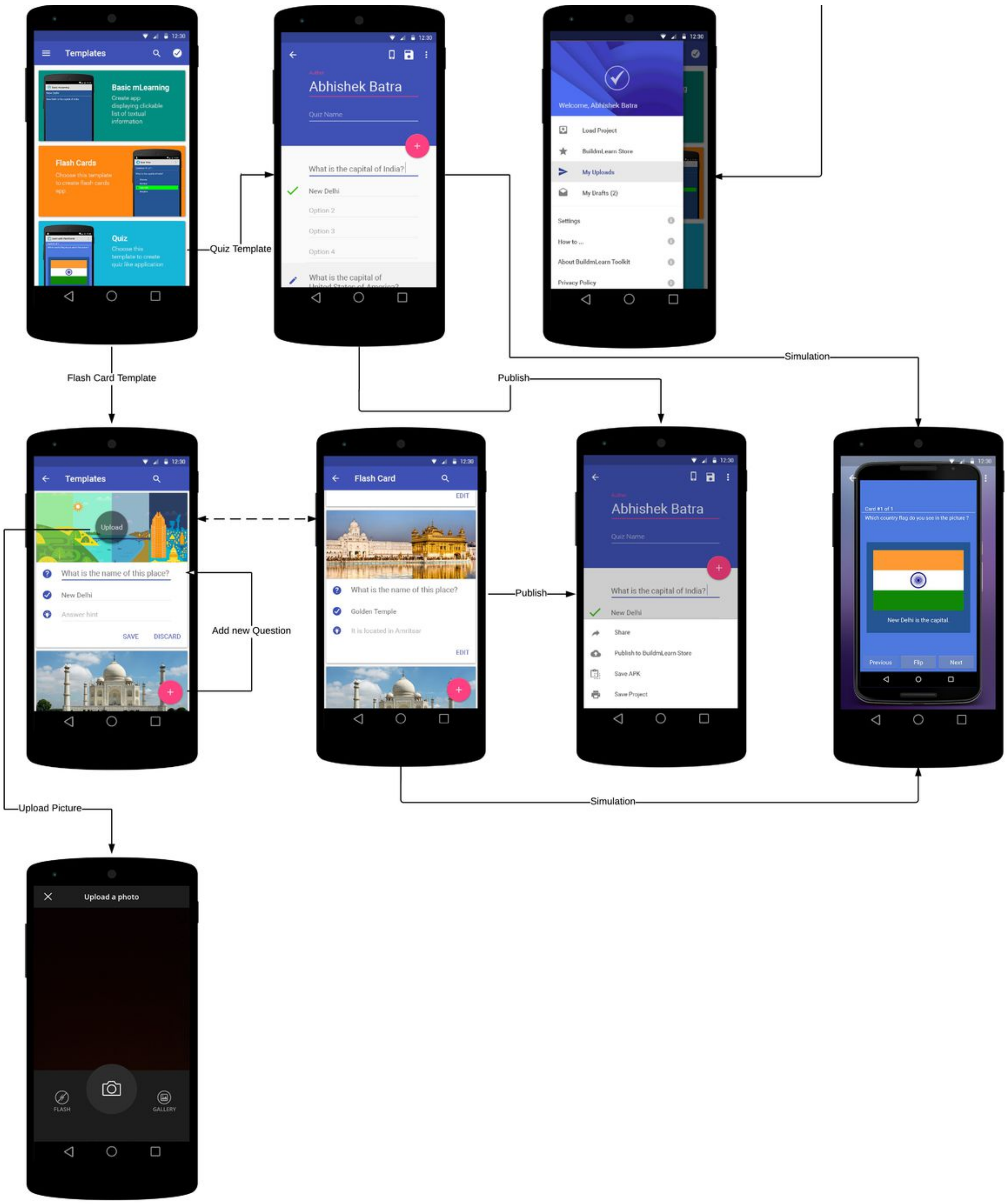
Application Flow Diagram



Welcome to BuildmLearn Toolkit

The BuildmLearn Toolkit is an easy-to-use program that helps users make mobile apps without any knowledge of application development.

The toolkit helps creating mobile application with various functionality and allows teachers to input their custom content.

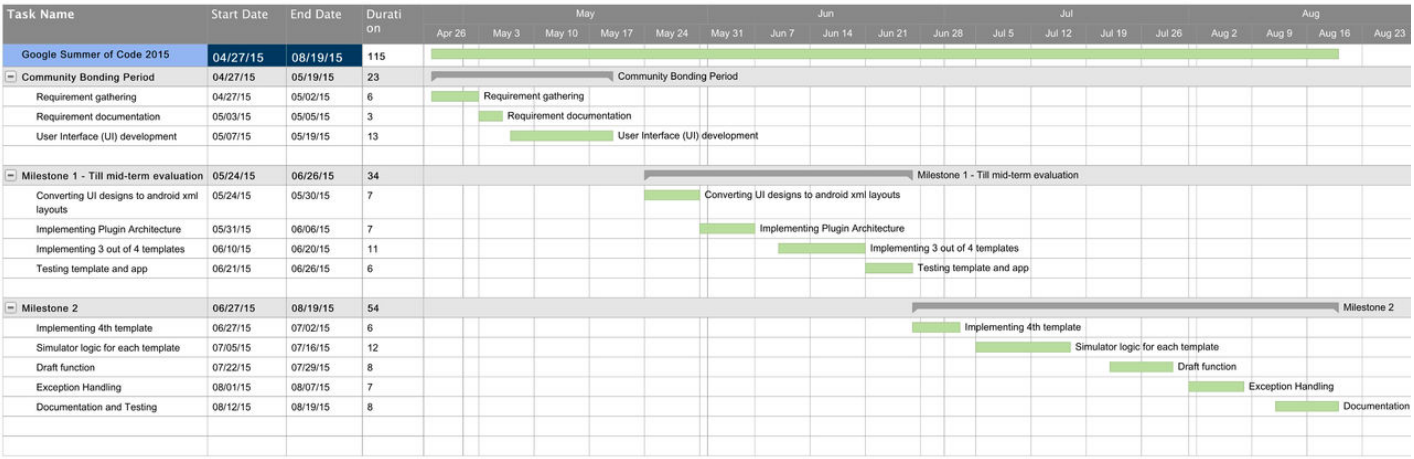


Full resolution image

Timeline:

GSoC is round about 13 week duration, with about 25 days of Community Bonding Period in addition. I will be spending 50% time in templates generation (currently four templates), 25% time on implementing the plugin architecture. And remaining 25% time on testing.

I am presenting the following timeline to keep things maintained during the whole period:



Task Name	Start Date	End Date	Duration
Google Summer of Code 2015	04/27/15	08/19/15	115
Community Bonding Period	04/27/15	05/19/15	23
Requirement gathering	04/27/15	05/02/15	6
Requirement documentation	05/03/15	05/05/15	3
User Interface (UI) development	05/07/15	05/19/15	13
Milestone 1 - Till mid-term evaluation	05/24/15	06/26/15	34
Converting UI designs to android xml layouts	05/24/15	05/30/15	7
Implementing Plugin Architecture	05/31/15	06/06/15	7
Implementing 3 out of 4 templates	06/10/15	06/20/15	11
Testing template and app	06/21/15	06/26/15	6
Milestone 2	06/27/15	08/19/15	54
Implementing 4th template	06/27/15	07/02/15	6
Simulator logic for each template	07/05/15	07/16/15	12
Draft function	07/22/15	07/29/15	8
Exception Handling	08/01/15	08/07/15	7
Documentation and Testing	08/12/15	08/19/15	8

May
Jun
Jul
Aug
Apr 26
May 3
May 10
May 17
May 24
May 31
Jun 7
Jun 14
Jun 21
Jun 28
Jul 5
Jul 12
Jul 19
Jul 26
Aug 2
Aug 9
Aug 16
Aug 23
Community Bonding Period
Requirement gathering
Requirement documentation
User Interface (UI) development
Milestone 1 - Till mid-term evaluation
Converting UI designs to android xml layouts
Implementing Plugin Architecture
Implementing 3 out of 4 templates
Testing template and app
Milestone 2
Implementing 4th template
Simulator logic for each template
Draft function
Exception Handling
Documentation

Full Resolution

Community Bonding Period

Creating all the UI screens on photoshop. Discussing detailed requirements with the mentor, and documenting the requirements.

Week 1: Layouts for the application

One of the most important part of the Android application is the User Interface. I plan to spend one week for converting the PSD’s to xml layout files. This will also include layouts for list items, grid items etc.

Week 2: Implementing the Plugin Architecture

Implementing the Plugin Architecture will involve making the SYSTEM module. SYSTEM modules consist of functionalities like Reading, writing to disk, making REST calls, parsing schema file for .buildmlearn files, publishing application to buildmlearn store.

Week 3-4 : Implementing three templates

During these 2 weeks I will be working on three templates out of four. Templates will be based on base plugin interface, thus they must override all the interface functions and provide their own logic. By the end of this week,

Week 5: Testing the templates and app functionality

This week will be spend on testing the app and bug fixing. By the end of this week I plan to release the first version of the toolkit on play store.

Week 6: Implementing the 4th Template

Last template will be implemented during this week. All the functionalities except Simulator logic will be completed by this week. By the end of this week we will having all the four templates integrated into toolkit.

Week 7-8 Simulator logic for each template

Simulator logic will involve making use template apk source and convert them into fragments instead of activities. These fragments will be passed to Simulator, which will then simulate the app in real time.

Week 9: Implementing draft functionality

Draft functionality will continuously save the date entered by the user. This will allow the user to pick up from the point, from where he previously left. This is highly useful in the situation when the user has entered a lot of data, and app crashes or phone is switched off due to low battery.

Week 10: Handling Exceptions

All the exception handling like network connectivity, memory space etc. will be done during this week.

Week 11-13: Documentation and Testing

Completing the documentation of the system and compensate for the elongated goals, along with rigorous testing of the app.

About Me

I'm a final year undergraduate student. I am pursuing my under-graduation in Computer Science and Engineering from Vellore Institute of Technology (VIT University), Vellore, India. I have a strong experience in programming in Java, PHP/MySQL, Html, CSS, JavaScript, C and C++ for last three years. I am passionate about programming, developing intelligent and efficient solutions to complex real world problems. Besides these, I love interacting with programmers, learning new ways to program better. Currently I am put up in Delhi doing internship at Zomato, which will end by 30th April 2015. I see GSoC with BuildmLearn as a great opportunity to meet great coders, hone my software development skills and precursor to a continuous association with BuildmLearn.

Contact Information

Email: abhishekbatra1993@gmail.com

Phone No: +91-8754758134

Some of the projects I have worked on are:

- GSoC'14, Learn From Map: Last year also I was a part of GSoC'14 with **BuildmLearn** as my mentoring organisation. The project involved mining knowledgeable geographical data from various geo data providers and developed an Android gaming app which intelligently formed the question based on mined data and rules by using the sensing and map capabilities.
- MIT SANA Android Client: I am currently working on a project under MIT named as SANA. The project aims at using android and java programming to address healthcare related issues using mobile technologies like OpenMRS.
- Untangle It: Untangle it is an Android multiplayer game. The game consist of ships, birds, plane, lighthouse and trees. Out of these light house and trees are always fixed whereas rest of the items can be moved over a dedicated area. They are all interlinked and connected by ropes. Player has to position all the objects in such a way that no two object intersect. https://play.google.com/store/apps/details?id=com.unary.untangleit
- EasyVote: An online voting system application which aims to service the global voting system. The project made into finals of Microsoft Imagine Cup, where we were ranked top 5 all over India.
- Envisage Wallpaper: The ENVISAGE wallpaper has huge selections of unique and high quality wallpapers. The wallpapers are fetched using the Flickr API. They are sorted under different categories. User can download the wallpaper, set them as home screen and contact image or can even share them on Facebook, twitter or message. Envisage wallpaper currently has 20k+. Play Store Link https://play.google.com/store/apps/details?id=com.shyam.flickrwallpaperapp
- Ally Chat Messenger: It uses Google Cloud Messaging to deliver messages. People can chat with anyone based on their assigned ids. The project is currently in its beta phase, so it's still buggy. https://github.com/batraabhishek/ally

Why are you the right person for this task? Have you worked with us before?

I am the right person for this project because I have knowledge and experience in Android app development. I can use my expertise in Android development and also learn more skills through this project. I have worked with BuildmLearn developers before also.

How many hours are you going to work on this a week? Do you have other commitments that we should know about? If so, please suggest a way to compensate if it will take much time away from Summer of Code.

Though currently I am working as an Intern at Zomato, but by the time GSoC results are out, my internship will be almost finished (finishing on 30th April), After that I will be dedicating all of my time on GSoC. I have the stamina to code and work for more than 10 hours daily. So I will be able to work 60+ hours weekly on my GSoC project.

Are you comfortable working independently under a supervisor or mentor who is just available virtually?

Yes, I am comfortable working under a mentor who is just virtually available. In 2014, I was a part of GSoC'14 with BuildmLearn. My mentors were Neha and Pankaj.I had a great learning experience working with them. I have done various freelancing projects in the past, where the employer was in some different country. I have followed their suggestions and advice and gave them the desired output.

Are you submitting proposals for any other organisations except us? If yes, which orgs have you applied for?

Yes, I am submitting another proposal with FOSSASIA.

Have you participated in any previous GSoC programs? If yes, tell us about your experience briefly and if you completed successfully.

Yes, I participated in GSoC'14 with BuildmLearn only under the mentorship of Pankaj and Neha. I worked on developing Learn From Map and completed it successfully. It was a great experience, the things I learned during GSoC, helped me in securing a good job during on campus placement. I learned how to write proper code, following all the rules. But most important thing that I learned was the Open Source Culture.

My progress so far relating to BuildmLearn Toolkit for android - I have designed a nice and pleasing UI for the app. I have studied and planned different modules that need to be implemented including template logic, simulator logic.

Looking forward to an exciting summer of development with BuildmLearn.