

## LMS Training and Quiz

### Objective:

Conducting the training process through Teamchat using InAppSDK API and SDK for customization.

### Implementation:

#### Prerequisite for using APIs:

1. Sign In at [gupshup.io](http://gupshup.io) to get an API Key.
2. Teamchat account.
3. Download latest Bot SDK from [here](#).
4. Create a bot using Bot SDK - createbot API.

#### URL Endpoint and HTTP Methods: CreateBot API

```
POST : http://api.webaroo.com/appsdk/api/components/createbot
```

#### HTTP Request using Curl to attach Teamchat Account:

```
curl --request POST \  
  --url http://api.webaroo.com/appsdk/api/components/createbot \  
  --header 'accept: application/json' \  
  --header 'apikey: YOUR API KEY HERE' \  
  --header 'content-type: multipart/form-data;  
boundary=---011000010111000001101001' \  
  --form 'name=Bot Training' \  
  --form email=bot@domain.com
```

**Response body:** ``JSON String { "created": "2016-02-15T04:51:19.154Z", "email": "bot@domain.com", "id": "56c15947e4b018a7fa636591", "lastUpdated": "2016-02-15T04:51:19.154Z", "name": "Bot Training", "type": "BOT", "verified": false }

Note the 'id' in response as this will be used for verifying the Bot and an code(OTP) will be send to email specified

5. Verify the bot and get the bot apikey using Bot SDK - verifybot API.

**\*\*URL Endpoint and HTTP Methods: VerifyBot API\*\***

```http

POST : http://api.webaroo.com/appsdk/api/components/verifybot/

HTTP Request using Curl to create a bot:

```
curl --request POST \  
  --url http://api.webaroo.com/appsdk/api/components/verifybot/ \  
  --header 'accept: application/json' \  
  --header 'apikey: YOUR API KEY HERE' \  
  --header 'content-type: multipart/form-data;  
boundary=--011000010111000001101001' \  
  --form code=7739 \  
  --form email=bot@domain.com \  
  --form id=56c15947e4b018a7fa636591
```

**Response body:** ```JSON String { "apikey": "1fa71ea53e4f4deacdb2a21ec123111b", "created": "2016-02-15T05:03:47.609Z", "email": "bot@domain.com", "id": "56c15947e4b018a7fa636591", "lastUpdated": "2016-02-15T05:04:39.9Z", "name": "Bot Training", "type": "BOT", "verified": true }

Once your bot is verified you can go for actual implementation.

##### Actual implementation:

**\*\*1. Create a TeamchatApi Instance**

```Java

```
TeamchatAPI api = TeamchatAPI.fromFile("config.json")  
    .setAuthenticationKey(authKey);
```

2. Create a room say LMS Training and Quiz using Room Class and add members to the room.

```
Room lmsRoom = api.context().create().setName("LMS Training and  
Quiz").welcome("Hello! To take a quiz post the keyword 'Quiz' in the room.")  
    .add(roomMember1)  
    .add(roomMember2);
```

3. Setups bot to listen to the events from teamchat server.

```
api.startReceivingEvents(new ClassConstructor());
```

4. Register for a keyword "Quiz" using @OnKeyword() annotation. when any user posts Quiz in the room, capture the user email and send a p2p post to that user and initiate the flow.

## Creating a P2P Group and send a welcome note

```
@OnKeyword("Quiz")
public void onTest(TeamchatAPI api)
{
    String user = api.context().currentSender().getEmail();
    String userName = api.context().currentSender().getName();
    String id = createp2pRoom(user,userName);
    Room p2pRoom = api.context().byId(id);
    p2pRoom.post(new PollChatlet()
        .setQuestionHtml("<img src=\"http://gs.tc.im/kZtXdIFd9MQ\">")
        .alias("welcomePoll"));
    api.perform(p2pRoom);
}

private String createp2pRoom(String user, String userName)
{
    TeamchatAPI api = TeamchatAPIImpl.fromFile("config.json")
        .setAuthenticationKey(authKey);
    Room p2pRoom = api.context().p2p(user).post(new TextChatlet("Welcome
"+userName));
    api.perform(p2pRoom);
    String id = p2pRoom.getId();
    return id;
}
```

## Embed HTML

```
```java. @OnAlias("welcomePoll") public void onreplytopoll(TeamchatAPI api) { String pollReply =
api.context().currentReply().getField("resp"); if(pollReply.equalsIgnoreCase("Yes")) {
api.performPostInCurrentRoom(new PollChatlet().setQuestionHtml("<
```

```
" + "
" + "Click to watch the video" + "
" + "Once you watch the video, click on Thumbs Up to start the quiz." + "
").alias("Questionnaire")); }else { api.performPostInCurrentRoom(new TextChatlet("If you wish to
continue the quiz later post Quiz in room to get started!")); } }
```

Through Teamchat, employees could initiate training modules based on their interest. The training bot would first send instructional material, which can be in any file format, including videos, audios, images and pdfs. The content is followed by assessment tests.

![] (http://i1.wp.com/www.teamchat.com/wp-content/uploads/2015/07/32.png?zoom=1.5&resize=700%2C460)

**\*\*5. &nbsp;&nbsp; Pseudo code for triggering questions:\*\***

```Pseudo

1. Retrive a question from your database and post it to the user.

```
@OnAlias("Questionnaire")
public void quizQuestions(TeamchatAPI api)
{
    api.perform(api
```

```

        .context()
        .currentRoom()
        .post(new PrimaryChatlet()
            .setReplyLabel("Submit")
            .setQuestionHtml("http://gs.tc.im/RfVfYRYw")
            .setReplyScreen(
                api.objects()
                    .form().addField(api.objects()
                        .input()
                            .name("userAnswer").label("Enter Option
Number").enableLastReplyUpdate()))
                .alias("reply")));
    }

```

2. Get the user reply, retrieve the correct answer from your database and accordingly notify the user as to their answer is correct or not. Loop Questionnaire Alias until questions are attempted by the user (build your logic accordingly).

```

@OnAlias("reply")
public void processReply(TeamchatAPI api)
{
    String userAnswer = api.context().currentReply().getField("userAnswer");
    String databaseAnswer = "1";
    if(userAnswer.equals(databaseAnswer)) // correct answer
    {
        api.perform(api
            .context()
            .currentRoom()
            .post(new PrimaryChatlet()
                .setQuestionHtml("http://gs.tc.im/kZviNa8WyJp")
                .alias("Questionnaire")));
    }else //incorrect answer
    {
        api.perform(api
            .context()
            .currentRoom()
            .post(new PrimaryChatlet()
                .setQuestionHtml("http://gs.tc.im/kZviN435a8WyJp")
                .alias("Questionnaire")));
    }
}

```

Bot Training 10:04 am

Posted on: 07 Jul 2015 02:29 PM IST

**A loan may be considered usurious because of excessive or abusive interest rates or other factors.**

1. Yes
2. No

Submit 1 Replies 0

Bot Training 10:04 am

Posted on: 08 Jul 2015 10:04 AM IST

**Correct Answer**



your current score is 2.0

0

Bot Training 10:04 am

Posted on: 07 Jul 2015 02:29 PM IST

**A loan may be considered usurious because of excessive or abusive interest rates or other factors.**

1. Yes
2. No

Submit 1 Replies 0

Bot Training 10:04 am

Posted on: 08 Jul 2015 10:04 AM IST

**Will stocks come in asset?**

1. Yes
2. No

Submit 0 Replies 0

Bot Training 10:06 am

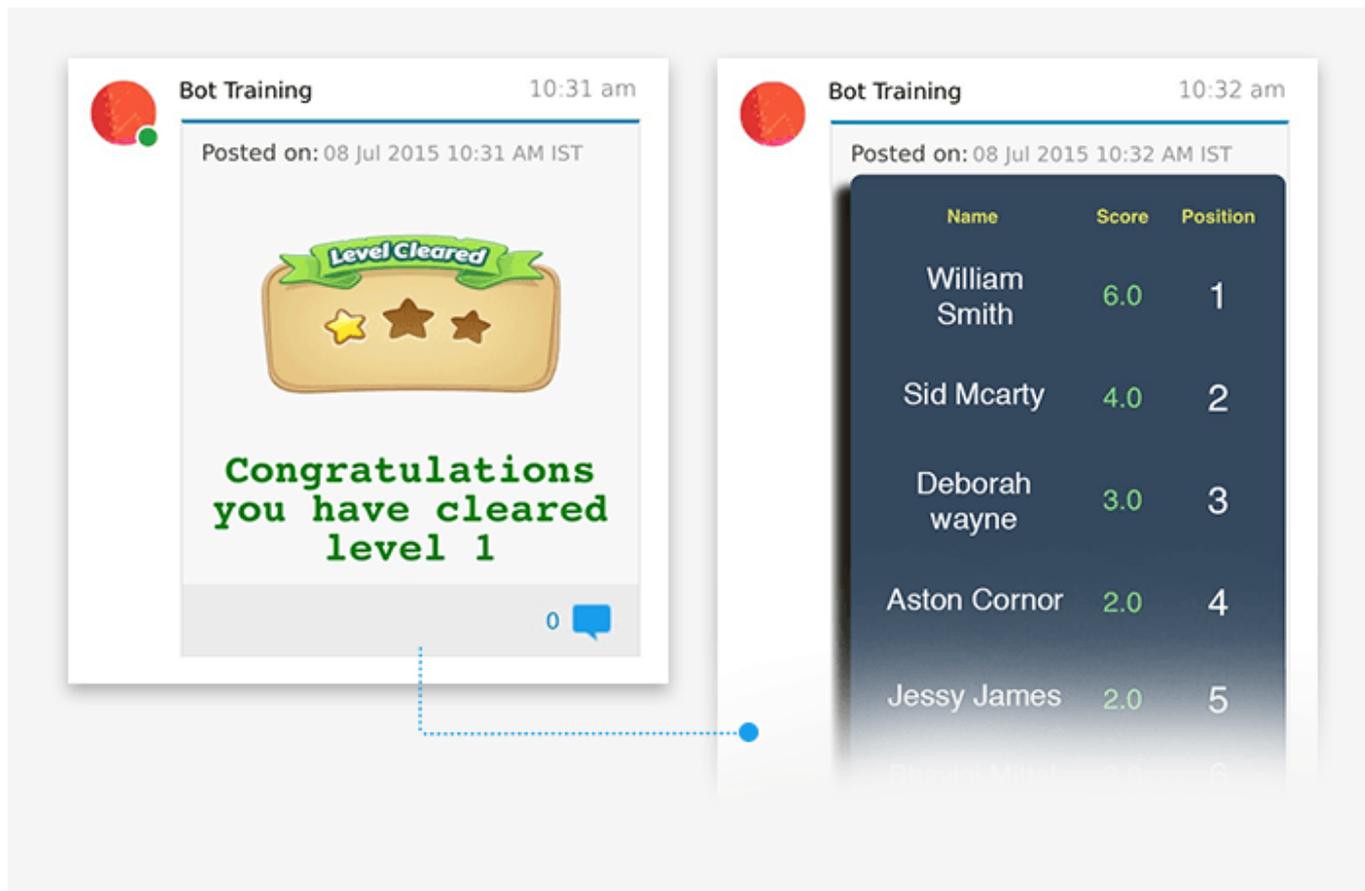
Posted on: 08 Jul 2015 10:06 AM IST

**What is the money left after paying taxes called?**

1. Savings
2. Net Income
3. Wealth
4. Assets

Submit 0 Replies 0

As employees complete a level, they become eligible to take the next level for the same topic, as well as other related topics. The training bot publishes individual accomplishments on the leader board and graduation board, that is sent out to the entire team. This drives peer recognition.



#### Results:

1. Employees were able to get the training they wanted anytime, anywhere. The app was simple and intuitive to use. Content retention was higher due to continuous reinforcement.
2. The training team could track if their training content was complete and helpful, and was able to make changes in real time.
3. Managers were able to track training adoption in real time. They could instantly verify that the training objectives of their employees were being met. The company could now train everyone all the time.
4. Managers were able to identify self-motivated individuals that were hungry for personal and professional growth.
5. The gamification elements such as leaderboard and graduation board drove the recognition of highly trained individuals leading to healthy competition amongst peers.
6. By tying individual performance and business outcomes to training achievement, the senior executives were able to close the feedback loop. They were able to directly measure and optimize training efficacy.
7. Compared to the prior training methods, the Teamchat solution proved to be better, faster and cheaper.