

## Flutter app talking to Laravel backend Project v1.0

Max budget: \$2,000 for Stage 1

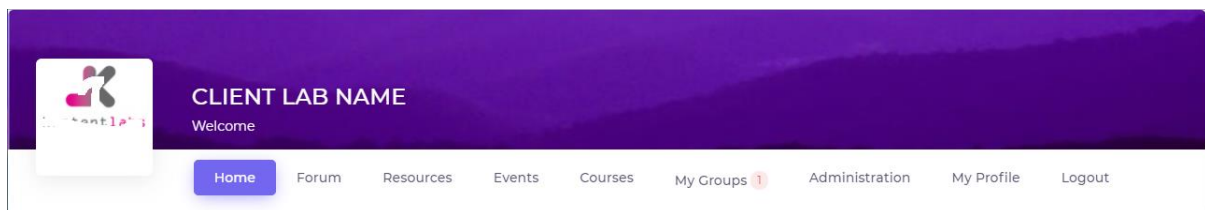
### Brief:

We currently have a full Laravel/php/MySQL portal developed and running and are looking to create a prototype Stage 1 cross platform (Android/IOS) application using Dart/Flutter 2x.

I have purchased a Flutter UI kit specifically for Chat – so this looks like it should provide all the UI we need for this project.

I want this developed using Dart/Flutter 2x. I use IntelliJ IDEA as the IDE and will need to be import this/your project into IntelliJ as required.

The primary colour of the App is Purple (e.g. our corporate header of the Laravel App)



All required Laravel API routes and handling POST/GET requests will be handled by me.

### Auth:

User will login on first access (using existing email and Password). There is no Registration features on App (will assume all users are already active on the Laravel Platform).

The App will then download & save to SQLite:

1. the Auth Token,
2. Welcome Message, [ text ]
3. Portal Name, [ short text ]
4. Portal Logo, [ link to clients' Logo ]
5. Last 3 Discussions (from the live portal) [ Discussion Name, content (text), Image, Author Name and Date of Post ]
6. Notifications (which come from an existing table on Laravel) [ Name and Type in Text ]

The API Route for the Laravel App is already setup and working based on a standard API Post request.

I also want the ability to Logout (and back in again). Each time they Login, the Welcome message is re-downloaded.

The Last 3 discussions need to be downloaded/refreshed every time the user accesses the Home page of the App (checking for changes).

## Chat:

This is the MAIN feature we want to work on the app.

The basic rules of this Chat are:

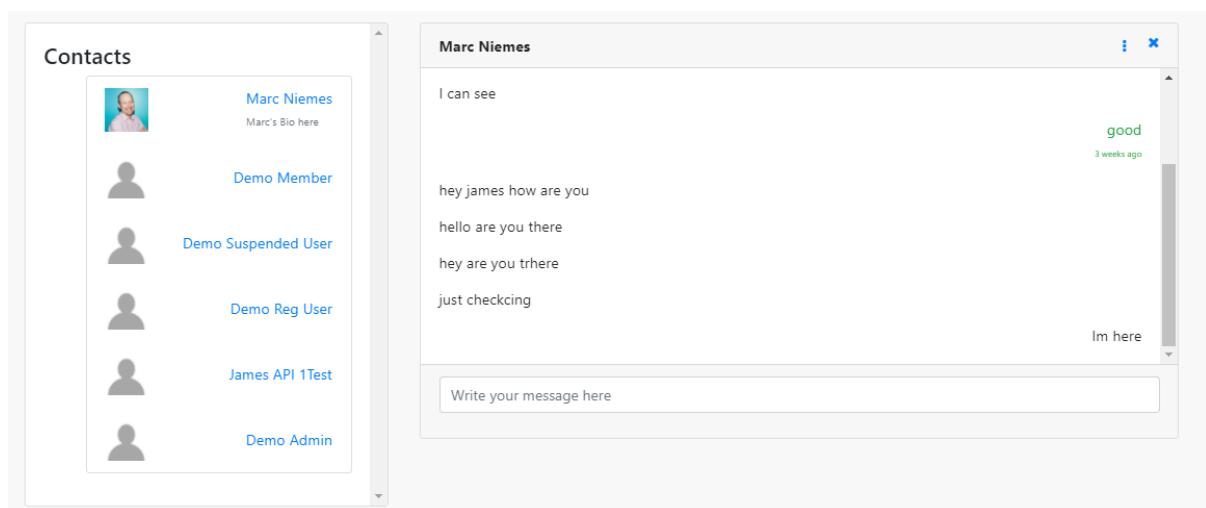
App users are a member of an existing Group on our Laravel Application. They can only connect/chat with members of that Group. Our Laravel App can send/download this list to the Mobile App.

The chat in stage 1 is only one to one (1-1). Later (Stage 2) we will expand this to Group Chat functions.

The Chat only needs to support Text and Photo (from Mobile phone Camera or Gallery)

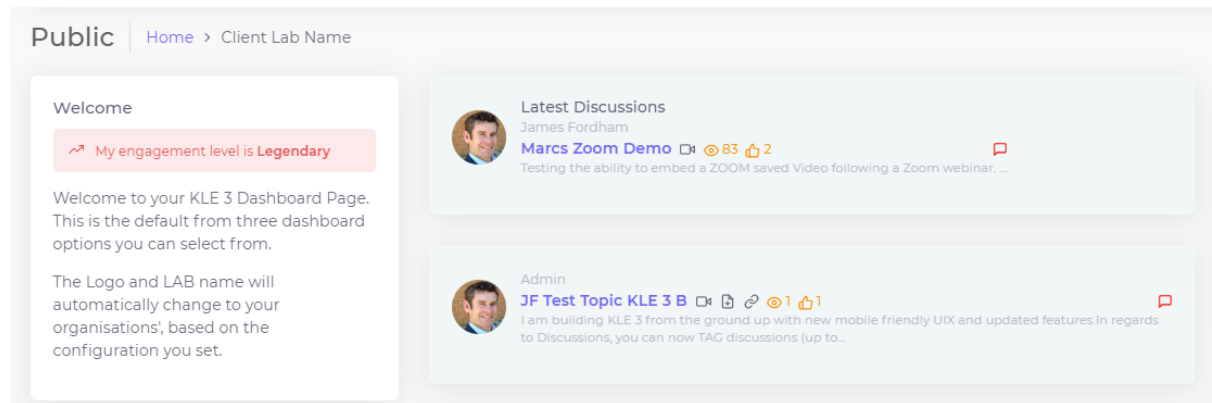
It will need a notification (e.g. little red number next the Chat Icon on home page AND the person sending a message) – whenever someone is sending a chat message to this person. Hence a Listener will need to be setup on the Mobile App for this.

Below is the working prototype of this concept (except it is text only for now) in the Laravel App.



The List of the Left are the Group members this person is a member of. They can start a chat with any of their group members.

## Discussions



Above is a sample of what users see inside the current Laravel Portal.

**Note:** The Welcome Message displayed top-left (the [Engagement Level](#) is NOT required for the [Mobile App](#)) is what needs to be displayed on the Home Page of the Mobile App (mentioned in the Auth section).

This information would be downloaded/sync'd on the Mobile App whenever they access the Home Page. [It does NOT need to display the ICONS as per above](#) – just the Author Image, Title and short Description and date of post.

When they click on it (from the Discussions Screen), it simply displays the FULL message with a COMMENT panel (large text field). They can then POST that comment back to the online Portal.

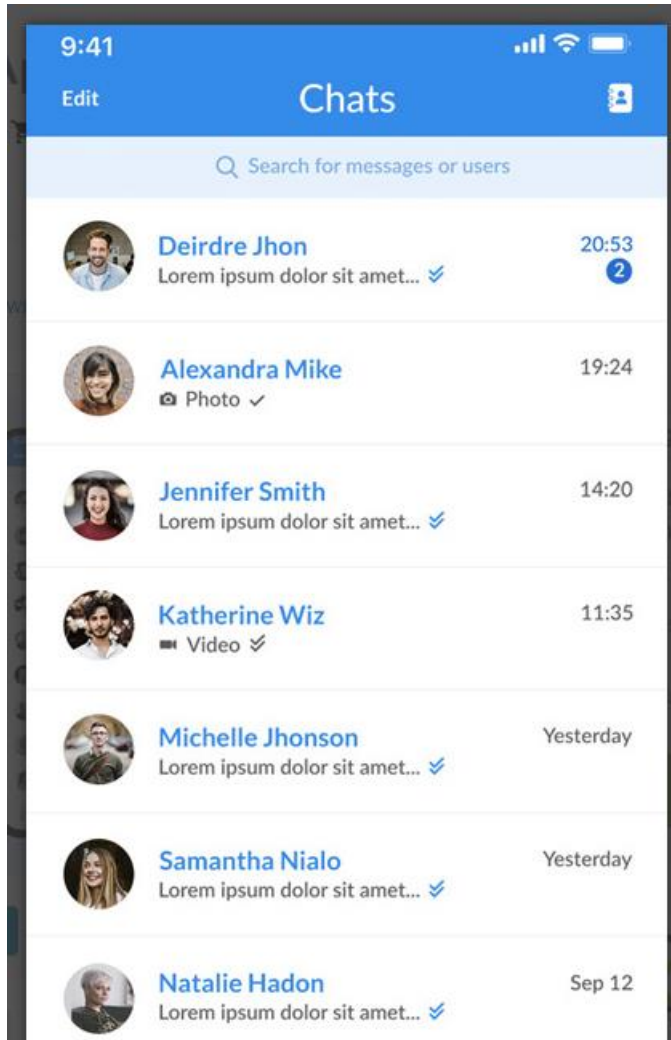
Each Discussion has an array of comments (from multiple members). This needs to be downloaded and viewed when the Member clicks on the Discussion object.

Comments are made up of: Author name (do not need to download the author image for comments here), The Comment (text) and Image (if image is part of the comment response). Comments would be Read only. The only POST request here is when the Mobile App user wants to post their own Comment to the Discussion.

## The Purchase UI

I purchased the following UI theme which I can provide a copy of as part of this project:

<https://codecanyon.net/item/gertsas-flutter-chat-app-ui-template/31488769>



Change primary Blue to **Purple** for Header