### Specification:

The goal of this project is to allow the developer or end user to create a MVP of their project (idea) using this platform. It will be open source and deployable locally or on cloud to create MVPs. it will allow exporting the complete MVP to re-import at any other deployment. Multiple MVPs can be created individually and can be shared between users but owned by a single user that created it.

### Modules:

##### SETUP RELATED

#### Create MVP:

Users will be able to create and own multiple MVPs. Each new mvp will have a unique name.

#### MVP Collections:

The creator will define collections in the MVP being used. Collections will be stored in the database for each MVP. the views can fetch collections and load data based on the collection. Criteria for CRUD operations will be defined on the setup and can be used in views to filter or fetch custom data from collections or perform custom update on collection based on the view user is interacting in. the permissions will also be created and available for each new collection.

#### MVP Forms:

MVP forms can be defined and be used in views to create/update collections.

#### MVP VIEWS:

Users can create and customize views based on components available that are commonly used to create views ( these include simple test, headings, paragraphs, table, list, grids and taking input data from forms, interactive event buttons to create/update or delete collections data or perform custom API actions.

#### User Module:

Each user can sign up to be a customer on the MVP builder. They will have permission to create their own mvp, each mvp will have their own user list and the creator will be treated as admin. Each MVP can have multiple user roles defined by the creator and will interact with the MVP areas based on permissions.

### Task planning:

| **Task** | **Status** | **Date** | **Notes** |
| --- | --- | --- | --- |
|  | Not started |  |  |
|  | Not started |  |  |
|  | Not started |  |  |

### 