|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URI | HTTP Method | Description | Request Object | Response Object |
| /api/contacts | GET | Gets all contacts to display in lenders list | n/a | {  text: string, value: integer  } |
| /api/categories | GET | Gets all categories | n/a | {  text: string,  value: integer  } |
| api/loans/summary/:id/:type | GET | Get all borrowing entries | Params: Id = user id, type = “borrow” | { id: integer,  date: datetime, remarks: text, name: string, contact\_number: string,  initial\_amount: decimal,  status: string,  category\_name: string, totalpaid: decimal,  currentamount: decimal }  **Notes:**  Calculation can be in the codes (but requires for all related data taken from history) or SQL select will do a sum/calculation using inner join  Totalpaid, currentamount |
| /api/loans | POST | Insert a loan i.e. borrow activity/info | Json -  Body: {  user\_id: integer,  contact\_id: integer, date: datetime,  initial\_amount: decimal,  category\_id: integer  remarks: text,  status: string,  type: string  }  E.g.:  {  "user\_id": 1,  “contact\_id": 2,  "date": "2020-12-31 11:24:08",  "initial\_amount": 10,  "category\_id": 8,  "remarks": "sometjing 5435435",  "status": "active",  "type": "borrow"  } | []  ~~{ Status: “Inserted” }~~ |
| /api/loans/:id | DELETE | Delete borrow data from loan and payment tables by the loan id | Param:  Id = loan id | []  ~~{ Status: “Deleted” }~~ |
| /api/loans/:id/:status | PUT | Update the borrow record to done/complete by the loan id | Param:  Id = loan id,  status = loan status e.g. “done” | []  ~~{ Status: “Updated” }~~ |
| /api/payments/:id | POST | Insert paid information for the specific borrow activity | Body: {  loan\_id: integer,  amount\_paid: float,  date: datetime  } | []  ~~{~~  ~~Status: “Inserted”~~  ~~}~~ |
| /api/payments/:id | GET | Get all paid info of a loan | Param:  Id = loan id | {  date\_entered: datetime, amount\_paid: decimal  } |
| /api/threshold/:id/ | GET | Get the threshold\_limit and total current amount of borrowed amount (after payment) | Param:  Id = user id | {  threshold\_limit: decimal, cur\_amt: decimal  } |
|  |  |  |  |  |