**Kumulos Documentation**

**Background**

Kumulos is a mobile backend as a service (MBaaS) utilized for the EPAP project. The EPAP application accesses and writes to the Kumulos database using API methods. It serves as the backend infrastructure on the server-side.

Data is stored in table structures and API methods are used by the application to access, add, modify, or delete the data stored in Kumulos. The requested data is returned and displayed in the app in various ways.

**Accessing Kumulos**

Kumulos can be accessed via a web browser at <https://kumulos.app.delivery> provided the user has login credentials. The project is listed under EPAP Prototype. As of April 2018, the service is subscribed and active. The project sponsors (Dr. Vikas O’Reilly-Shah and Dr Aalap Shah) also have been granted access.

Most of the features relevant to this project can be accessed in the left sidebar after logging in under the Build section, under which there are four subsections:

* Tables: New tables may be created and existing tables can be accessed. The contents of the individual tables, such as fields and the data itself, can be created or modified from this menu.
* API: Access to the data is achieved through custom API methods
* Deploy: Deployment platforms may be chosen in this menu (iOS and Android for this project)
* Download: Native libraries may be downloaded

**Build: Tables**

Fourteen tables are used by the EPAP application:

|  |  |
| --- | --- |
| Table Name | Fields |
| Drug Plan Items | drugPlanItemNotes, drugPlanItemDosage, drugPlan, drug |
| Drug Plans | drugPlanStatus, drugPlanNotesToPhysician, drugPlanNotesToPharmacist, drugPlanExtubation, drugPlanMaintenance,  primaryPharmacist, plan |
| Drugs | drugName, drugCategory, drugType, drugUnitOfMeasure, drugNeedsNotes, drugNeedsDosage, createdByUser |
| Equipment | equipmentName, equipmentCategory, equipmentType, equipmentUnitOfMeasure, equipmentNeedsLocation, equipmentNeedsSize, equipmentNeedsNotes, createdByUser |
| Equipment Plan Items | equipmentPlanItemLocation, equipmentPlanItemSize, equipmentPlanItemNotes, equipmentPlanItemQuantity, equipment, equipmentPlan |
| Equipment Plans | equipmentPlanStatus, equipmentPlanNotesToPhysician, equipmentPlanLabValuesType, equipmentPlanHasCSpineOrTMJIssue, equipmentPlanPosition, equipmentPlanTrend, equipmentPlanProneInfo, equipmentPlanNotesToTech, equipmentPlanRotation, plan, primaryTechnician |
| Messages | messageSubject, messageBody, messageSender, messageRecipient, messagePlan |
| Plans | planDescription, planStatus, planPatientInitials, planPatientHeight, planPatientWeight, planPatientAge, planPatientSex, planORDate, planAnesthesiaType, planNPODate, planSavePoint, primaryPhysician, procedure, equipmentPlan, drugPlan, createdByUser,  startRoom, operatingRoom, endRoom |
| Procedure Drugs | procedureDrugQuantity, pharmaceutical, procedure |
| Procedure Equipments | procedureEquipmentQuantity, equipment, procedure |
| Procedures | procedureName, procedureDescription, createdByUser |
| Roles | roleName |
| Rooms | roomNumber, roomType |
| Users | username, userPassword, userFirstName, userLastName, userIsActive, userIsAdministrator, userJoinDate, userAuthToken, userResetCode, userIsRegistered, userResetCodeDate, userAuthTokenDate, role |

**Build: API**

There are numerous API methods used by the EPAP application:

|  |  |
| --- | --- |
| Table | Aliases |
| Drug Plan Items | * createDrugPlanItem * deleteDrugPlanItems * getDrugPlanItems |
| Drug Plans | * createDrugPlan * getDrugPlan * updateDrugPlan * updateDrugPlanStatus |
| Drugs | * addDrug * deleteDrug * drugDQ * getAllDrugs * getDrug * updateDrug |
| Equipment | * addEquipment * deleteEquipment * getAllEquipment * getEquipment * updateEquipment |
| Equipment Plan Items | * createEquipmentPlanItem * deleteEquipmentPlanItems * epiDQ * getEquipmentPlanItems |
| Equipment Plans | * createEquipmentPlan * getEquipmentPlan * updateEquipmentPlan * updateEquipmentPlanStatus |
| Messages | * addMessage * deleteMessage * getAllMessages * getMessage |
| Plans | * createPlan * deletePlan * getAllPlans * getPlan * getPlanRawData * getPlansByStatus * getUserPlans * getUserPlansByStatus * planDQ * updatePlan * updatePlanStatus |
| Roles | * createRole * selectRoles |
| Rooms | * addRoom * deleteRoom * getAllRooms * updateRoom |
| Users | * authenticateUser * changeRole * createUser * getUser * selectAllUsers * setUserAuthToken * setUserResetCode * updateUserByAdmin * updateUserPassword * updateUserProfile * validateResetCode |