**Diabetes Management System - Technical Reference**

**Core Components Overview**

**Application Structure**

* **Entry Point**: main.py - Flask app initialization, blueprint registration
* **Config**: MongoDB (localhost:27017/native\_new), CORS enabled for localhost:3000
* **Authentication**: JWT-based, 24hr expiry, token validation middleware

**Key Routes**

Copy

/auth/\* - User authentication, registration

/doctor/\* - Patient management (protected)

/food/\* - Food database and calculations

/test - Health check endpoint

**Data Models**

1. Food Categories:

python

Copy

{

'basic\_foods': FOOD\_DATABASE,

'starch': STARCH\_LIST,

'vegetables': STARCHY\_VEGETABLES,

'fruits': FRUITS,

'dairy': MILK\_AND\_DAIRY,

'cuisines': {

'indian': INDIAN\_DISHES,

'chinese': CHINESE\_DISHES,

'italian': ITALIAN\_DISHES

}

}

1. Treatment Constants:

python

Copy

{

'absorption\_factors': {

'fast': 1.0,

'medium': 0.9,

'slow': 0.8

},

'timing\_factors': {

'breakfast': 1.2,

'normal': 1.0,

'late\_night': 0.9

},

'defaults': {

'insulin\_to\_carb\_ratio': 1/10,

'correction\_factor': 50,

'target\_glucose': 100

}

}

**Critical Functions**

1. Food Service:

python

Copy

get\_food\_details(food\_name)

search\_food(query, category=None)

calculate\_absorption\_factor(type, timing)

convert\_to\_standard\_nutrients(portion, measurement, details)

1. Authentication:

python

Copy

@token\_required *# Decorator for protected routes*

validate\_token(token)

generate\_token(user\_data)

**API Integration Examples**

1. Authentication:

python

Copy

POST /login

{

"username": str,

"password": str,

"user\_type": str

}

1. Food Management:

python

Copy

GET /api/food/search?q=query&category=category

POST /api/food/nutritional-summary

{

"meal\_items": [

{

"name": str,

"portion": float,

"measurement": str

}

]

}

1. Doctor Routes:

python

Copy

GET /api/doctor/patients

POST /api/doctor/update-patient-constants

{

"patientId": str,

"constants": {

"insulin\_to\_carb\_ratio": float,

"correction\_factor": float,

"target\_glucose": float

}

}

**Project Structure**

backend/

├── main.py

├── config.py

├── activity.py

├── blood\_sugar.py

├── constants.py

├── enhanced\_meal\_insulin.py

├── meal\_insulin.py

├── models/

│ ├── \_\_init\_\_.py

│ ├── food\_data.py

│ └── portion\_converter.py

├── routes/

│ ├── \_\_init\_\_.py

│ ├── auth\_routes.py

│ ├── doctor\_routes.py

│ ├── food\_routes.py

│ └── test\_routes.py

├── services/

│ └── food\_service.py

└── utils/

├── \_\_init\_\_.py

├── auth.py

└── error\_handler.py