

(1) Register

The screenshot shows the Postman application interface. In the top navigation bar, the 'File', 'Edit', 'View', 'Window', and 'Help' menus are visible. The 'API Network' dropdown is open, showing 'bookmyconsultation' as the selected collection. The 'Explore' tab is active. The main workspace displays a 'POST' request to 'http://localhost:8080/users/register'. The 'Body' tab is selected, showing a JSON payload:

```
1   {
2     "firstName": "fname",
3     "lastName": "lname",
4     "dob": "1993-08-06",
5     "mobile": "1234567899",
6     "password": "test",
7     "emailId": "test@gmasil.com"
8 }
```

Below the body, the 'Test Results' section shows a successful response with status 200 OK, time 42 ms, and size 843 B. The response body is identical to the JSON sent in the request. The bottom of the screen shows the Mac OS X dock with various application icons.

(2) Login

The screenshot shows the Postman application interface. In the center, a POST request is being made to `http://localhost:8080/auth/login`. The 'Authorization' tab is selected, showing 'Basic Auth' as the type. The 'Username' field contains `test@gmasil.com` and the 'Password' field contains `test`. A checked checkbox labeled 'Show Password' is visible. Below the request, the response body is displayed in JSON format:

```
1   "id": "test@gmasil.com",
2   "firstName": "fname",
3   "lastName": "lname",
4   "emailAddress": "test@gmasil.com",
5   "mobilePhoneNumber": "1234567890",
6   "lastLoginTime": null,
7   "accessToken": "eyJraWQiOiI5NDE2MzdiY505ZWfJLTQyNwQtYnFmMy1iN2V1NzRmNGZlYzMlCj0eXai0iJKV10iLChbGci0iJIUzUxMiJ9.
8   eyJhdW1i0iJ0ZXN0ODgtYXNpbC5jb20iLCjpc3M10iJodHRwczovL23vb2ttoWvbnNlbHRhdG1vb15jb26iLCJleHA1oJE2NTg5MjYsImIhdCI6MTY100g5N38.
9   7YoEo-taxhB2p0pxFv9IMH-xYhj098JWWAQ0t05G4h4t0k347hAdcY80uSkKoxvhCV_ziIR70Iw_m_L6Kx1Meu"
```

The status bar at the bottom indicates a 200 OK status with a time of 40 ms and a size of 1.07 KB.

(3) Get User

The screenshot shows the Postman application interface. The top navigation bar includes Home, Workspaces, API Network, Explore, and various system icons. The date and time displayed are Wed Jul 27 12:02 AM.

In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' containing several requests:

- GET get User
- GET get User appointments
- POST register
- POST Login
- POST Logout
- GET get all doctor
- GET get all speciality
- GET get doctors timeslots
- GET get a doctor
- GET __get a doctor Copy
- GET get an appointment
- POST add doctor
- POST book an appointment
- POST post a rating

The current request being edited is 'GET get User' with the URL `http://localhost:8080/users/test@gmasil.com`. The 'Headers' tab is selected, showing a single header entry:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Authorization	Bearer {{BMCAuthToken}}	
Key	Value	Description

The 'Body' tab shows the response body in JSON format:

```
1 "emailId": "test@gmasil.com",
2 "password": "E19908A202CC08",
3 "firstName": "fname",
4 "lastName": "lname",
5 "dob": "1993-08-06",
6 "mobile": "1234567890",
7 "createdDate": "2022-07-26",
8 "salt": "zH4VLNkG03idOKuidk6pYy10a8G072U7Xq26zI3gw="
```

The status bar at the bottom indicates: Status: 200 OK Time: 31 ms Size: 887 B Save Response.

(4) Get all speciality

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' which contains several API endpoints. One endpoint is selected: 'GET get all speciality'. The main panel displays the request details:

- Method:** GET
- URL:** http://localhost:8080/doctors/speciality
- Params:** (empty)
- Headers:** (empty)
- Body:** (empty)
- Tests:** (empty)
- Settings:** (empty)

The response body is shown in a code editor:

```
1  ["CARIOLOGIST",
2  "GENERAL_PHYSICIAN",
3  "DENTIST",
4  "PULMONOLOGIST",
5  "ENT",
6  "GASTRO"]
```

The status bar at the bottom indicates: Status: 200 OK Time: 19 ms Size: 700 B Save Response.

(5) Get all doctors

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' containing several API endpoints. One endpoint is selected: 'GET get all doctor'. The request URL is set to `http://localhost:8080/doctors?specialty=CARDIOLOGIST`. A 'Params' table is visible with one entry: 'specialty' set to 'CARDIOLOGIST'. The 'Body' tab shows a JSON response with a single object containing a doctor's details. The response body is:

```
1  {
2   "id": "UUID-34",
3   "firstName": "Meghan",
4   "lastName": "Jordan",
5   "specialty": "CARDIOLOGIST",
6   "dob": "2022-04-03",
7   "address": [
8     {
9       "id": "UUID-34",
10      "addressLine1": "P.O. Box 985, 6300 Odio, Road",
11      "addressLine2": "57,4999, -121.87884",
12      "city": "Oud-Turnhout",
13      "state": "Antwerpen",
14      "postcode": "3514"
15    },
16    {
17      "mobile": "16891025 2785",
18      "emailId": "vel@itamecdapibus.net"
19    }
20  ]
```

(6) Get a doctor

The screenshot shows the Postman application interface. In the top navigation bar, the title "Postman" is visible along with "File", "Edit", "View", "Window", and "Help". The date "Wed Jul 27 12:07 AM" is displayed in the top right corner. The main workspace is titled "My Workspace" and contains a collection named "bookmyconsultation". Under this collection, there are several requests listed:

- GET get User
- GET get User appointments
- POST register
- POST Login
- POST Logout
- GET get all doctor
- GET get all speciality
- GET get doctors timetables
- GET get a doctor
- GET __get a doctor Copy
- GET get an appointment
- POST add doctor
- POST book an appointment
- POST post a rating

The "GET get a doctor" request is currently selected. The request details show a "GET" method pointing to `http://localhost:8080/doctors/UUID-34`. The "Headers" tab is active, containing a single header entry:

KEY	VALUE	DESCRIPTION
Authorization	Bearer {{BMCAuthToken}}	Description

The "Body" tab displays the response body in JSON format:

```
1  {
2   "id": "UUID-34",
3   "firstName": "Maghan",
4   "lastName": "Jordan",
5   "speciality": "CARDIOLOGIST",
6   "dob": "2022-04-03",
7   "address": [
8     {
9       "id": "UUID-34",
10      "addressLine1": "P.O. Box 985, 6300 Odio. Road",
11      "addressLine2": "57.40999, -121.87884",
12      "city": "Oud-Turnhout",
13      "state": "Antwerp",
14      "postcode": "3514"
15    },
16    {
17      "mobile": "+4089025 2785",
18      "emailId": "veil@itametaapibus.net",
19      "pan": "JF153IQX9CP",
20    }
21  ]
```

The status bar at the bottom indicates "Status: 200 OK" and "Time: 31ms". The bottom dock contains various Mac OS X application icons.

(7) Get doctors timeslots

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' containing several API endpoints. One endpoint, 'GET get doctors timeslots', is highlighted. The main workspace shows a 'GET' request to the URL `http://localhost:8080/doctors/UUID-34/timeSlots?date=2021-05-26`. The 'Params' tab is selected, showing a parameter 'date' with the value '2021-05-26'. Below the request, the 'Body' tab displays the JSON response:

```
1  {
2     "doctorId": "UUID-34",
3     "availableDate": "2021-05-26",
4     "timeSlots": [
5         "11AM-12AM",
6         "06PM-07PM",
7         "05PM-06PM",
8         "07PM-08PM",
9         "08PM-09PM",
10        "10AM-11AM"
11    ]
12 }
```

The status bar at the bottom indicates a 200 OK status, 34 ms time, and 759 B size.

(8) Add doctor

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' which contains several API endpoints. One endpoint, 'POST add doctor', is selected. The main panel displays a POST request to 'http://localhost:8080/doctors'. The 'Body' tab is active, showing a JSON payload:

```
1 {
2     "firstName": "fname",
3     "lastName": "lname",
4     "dob": "1903-08-06",
5     "mobile": "1234567890",
6     "password": "test",
7     "emailId": "test@gmail.com",
8     "pan": "EDLSS5080L",
9     "address": {
10         "id": "aefb7990-cc24-4c6a-96e2-af0e2d76ba8",
11         "addressLine1": "add1",
12         "addressLine2": "add1",
13         "city": "BHM",
14         "state": "TN",
15         "postcode": "123456"
16     },
17 }
```

The status bar at the bottom indicates a successful response: Status: 200 OK, Time: 157 ms, Size: 970 B. The Mac OS X Dock is visible at the bottom, showing various application icons.

(9) Book an appointment

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a collection named 'bookmyconsultation' containing several requests:

- GET get User
- GET get User appointments
- POST register
- POST Login
- POST Logout
- GET get all doctor
- GET get all speciality
- GET get doctors timeslots
- GET get a doctor
- GET __get a doctor Copy
- GET get an appointment
- POST add doctor
- POST book an appointment
- POST post a rating

The 'book an appointment' request is selected. The 'Body' tab is active, showing the following JSON payload:

```
1  ...
2  ...
3  ...
4  ...
5  ...
6  ...
7  ...
8  ...
9  ...
10 ...
11 ...
12 ...
13 ...
14 ...
15 ...
```

The 'Body' tab also includes tabs for Pretty, Raw, Preview, Visualize, and Text.

At the bottom of the interface, there is a toolbar with various icons and a status bar indicating 'Status: 200 OK'.

(10) Post a rating

The screenshot shows the Postman application interface. The top navigation bar includes Home, Workspaces, API Network, Explore, and various system icons. The main workspace on the left is titled "My Workspace" and contains a collection named "bookmyconsultation". This collection includes several requests: GET User, GET get all, GET get all, GET get db, GET get ..., GET get an ..., POST add ..., POST book, POST post ..., POST register, POST Log ..., and POST post a rating. The "post a rating" request is currently selected. The request details panel shows a POST method to the URL `http://localhost:8080/ratings`. The "Body" tab is active, displaying the following JSON payload:

```
1  {
2   "appointmentId": "1646f379-bc4b-47e3-a335-01e4704931cb",
3   "doctorId": "UUID-34",
4   "rating": "1",
5   "comments": "assad"
```

The "Headers" section lists "Content-Type: application/json". Below the request details, the "Test Results" panel shows a successful response with a status of 200 OK, time 65 ms, and size 582 B. The bottom of the screen shows the macOS Dock with various application icons.

(11) Get user appointments

The screenshot shows the Postman application interface. The left sidebar displays collections, environments, mock servers, and history. The main workspace shows a collection named "bookmyconsultation" with a sub-collection "get User". A specific GET request titled "get User appointments" is selected. The request details show a GET method, URL `http://localhost:8080/users/test@gmasil.com/appointments`, and a "Params" tab. The response body is displayed in "Pretty" format, showing a JSON object with appointment details:

```
1 {
2   "appointmentId": "b562dc39-bab7-4452-b847-34c9aceb05fc",
3   "doctorId": "UUID-34",
4   "doctorName": "Meghan Jordan",
5   "userId": "test@gmasil.com",
6   "userName": "fname",
7   "userEmailId": "test@gmasil.com",
8   "timeSlot": "08PM-06PM",
9   "appointmentDate": "2021-05-26",
10  "symptoms": "TEST",
11  "priorMedicalHistory": "NA"
12 }
13
14 }
```

The status bar at the bottom indicates the request was successful with a status of 200 OK, time 62 ms, and size 908 B.

(12) Logout

The screenshot shows the Postman application interface. The top navigation bar includes 'File', 'Edit', 'View', 'Window', and 'Help' menus. The date 'Wed Jul 27 12:17 AM' is displayed in the top right corner. The main workspace is titled 'My Workspace' and contains a collection named 'bookmyconsultation'. Under this collection, there are several API endpoints listed under the 'Logout' section: 'GET get User', 'GET get User appointments', 'POST register', 'POST Login', and 'POST Logout'. The 'Logout' endpoint is selected. The request details show a 'POST' method to 'http://localhost:8080/auth/logout'. The 'Headers' tab is active, showing a single header 'Authorization' with the value 'Bearer {{BMCAuthToken}}'. The 'Body' tab is selected, showing a simple JSON response with the number '1'. The bottom status bar indicates a 'Status: 200 OK' response.