JobConnect Android APP

Front-end: Flutter

Backend: REST API using AWS API Gateway, Lambda and Dynamodb no-sql

App Features:

JobConnect app has the basic features like Sign-up to create a new user account and login screens. The basic validation has been done on user creation.

|  |  |
| --- | --- |
| A screenshot of a login screen  Description automatically generated |  |

After login, user can switch between Job Provider and Job Seeker roles. Job Seeker can filter the jobs based on the Job Types like ‘PAINTER’, ‘HOME NURSE’, ‘FARM WORKERS’, ‘TRACTOR DRIVERS’ etc. When Job Seeker apply for a job, the Job Provider will get a Email notification (Currently only few emails are enabled for testing).

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

As a Job Provider, a new job can be created and published for Job Seekers. They can select from an existing list of Job Types as listed in the image above. If they key-in a new Job Type, that will be automatically added to the list of Job Types and will be available for all users to filter and search for a particular job type.

If there are more than one vacancy in a job, the Job poster can indicate that while posting/creating a new job. As soon as the the Job seeker applies for a job and a job seeker has accepted the job application, the number of vacancies will get reduced automatically.

|  |  |
| --- | --- |
|  | The job seeker can view the AVAILABLE jobs and they can also edit and change the Asking Salary if they want, so that the Job Provider can view the new Requested wage for the role and either accept or decline the request.  The Pay Type can be one of: Daily, Weekly, Monthly and Annual. So this app can support all types of job seekers and providers in their near by areas. The thought behind this app is that ‘There are people like to work even for low wages, but they need job regularly’.  App bar has menu has options for filtering jobs based on Job Types, Job Status. There is also option available for changing the user role from ‘Job Seeker’ to ‘Job Provider’.  The Job Provider role can see only the jobs posted by him and its progress in various stages. If the Job Provider wants to view and apply for jobs posted by other Job providers, then he/she has to switch role to ‘Job Seeker’ so that he can filter, view and apply for jobs.  The ‘Add New Job’ option is available for both the roles but when a new job is created, the logged person will be the Job provider. |

|  |  |
| --- | --- |
|  | Job Workflow:  When a Job Poster has published a job, the status of the Job will be “AVAILABLE”.  Job Seeker can view all the “AVAILABLE” jobs and he can click on the Request button to apply for the job. The status of the job will be “REQUESTED”. Until the job poster, views and accepts the job application, the number of vacancies in the job remains the same, so that any number of applicants(job seekers) can apply for the job.  Job Provider, can view the ‘REQUESTED’ jobs from various applicants and either ‘ACCEPT’ or ‘DECLINE’ the application of any particular job seeker.  Both the Job Seeker or Job Provider can cancel a ‘ACCEPTED’ or “REQUESTED” job any time. In that case, the job status will change to “REQUEST\_CANCELLED’ or ‘ACCEPT\_CANCELLED’ respectively.  Job provider can “DELETE” a job any time until its status is “READY”, When a job is “DELETED”, the job seekers can view the status from their login.  A Job will move to “READY” status when all the vacancies of a job is completely filled-in, meaning if there are 5 vacancies then 5 job seekers have applied and all the 5 requests have been ACCEPTED by the Job provider then the job status will move to “READY”. |

Other screens from the App:

|  |  |
| --- | --- |
|  |  |

By clicking the ‘View Details’ option, the Job Seeker can view the Job description and the Job provider’s contact details. Similarly, the Job Provider can view the applicant’s contact details.

|  |  |
| --- | --- |
|  |  |

A screenshot of a phone

Description automatically generated

There are lot of improvements possible in the app including localization of regional languages, UI improvements, data validations, mobile/email otp etc.

The Forgot Password link is not enabled yet as of submission day.

APK links and references:

**Verify Login:**

API URL: <https://dk7vxhqz9b.execute-api.ap-south-1.amazonaws.com/jobConnect/Login>

Input:

{

"Login\_Id": "Ranjan102",

"Password": "1234"

}

API response example:

{

    "statusCode": 200,

    "body": "{\"Login\_Id\": \"RanjanRajesh09\", \"Status\": \"Email not yet verified\", \"User\_Name\": \"Rajesh Ranjan\", \"Mobile\_Number\": \"9840690981\", \"Email\": \"testmail123@gmail.com\"}"

}

“Email not yet verified” status indicates that the Login\_Id and Password matches but the Email of the user is yet to be verified. If we make Email verification as optional then we can consider this as a successful verification of Login\_Id and Password.

“Success” status indicates that the Email verification is already done and the Login\_Id and Password also matches.

“Failure” status indicates that the Login\_Id and Password did not match.

**Verify Email:**

API URL: https://ck67zhqfe7.execute-api.ap-south-1.amazonaws.com/JobConnect/verifyEmail

To send Access\_Code via E-Mail. Access\_Code will be generated in backend and send via E-Mail corresponding to the Login\_Id, if exists.

{

"Login\_Id": "Ranjan100",

"Action": "SendAccessCode"

}

To verify the Access\_Code. The Access\_Code sent in API request will be verified again the Access\_Code for the Login\_Id. Front-end application has to send the Access\_Code entered by Customer which was received via his/her E-Mail.

{

"Login\_Id": "Ranjan100",

"Action": "VerifyEmail",

"Access\_Code": "694630"

}

The Status field will have "Email Sent" when Access\_Code has been successfully sent via email for the Login\_Id given in input request.

The status field will have "Email Verified" when the Access\_Code has been successfully verified.

**New User Registration:**

API URL: <https://1d51g1pe1g.execute-api.ap-south-1.amazonaws.com/jobConnnect/registerUser>

Input:

{

"Login\_Id": "Ranjan100",

"Mobile\_Number": "9840690984",

"Login\_Type": "JS",

"Password": "1234",

"User\_Name": "Ranjan",

"Gender": "M",

"Age": "42",

"Profession": "Painter",

"Experience\_Years": "3",

"Experience\_Months": "6",

"Average\_Rating": "4.5",

"Street": "MG Road",

"City": "Yerpedu",

"State": "Andhra Pradesh",

"Pincode": "517619",

"Email": "ranjan.murali82@gmail.com",

"Email\_Status": "UNVERIFIED"

}

API Response, Example:

{

    "statusCode": 202,

    "body": "{\"Login\_Id\": \"RanjanRajesh09\", \"Status\": \"User Already Exist\"}"

}

**Add New Job:**

API URL: <https://hsfq9bsuv0.execute-api.ap-south-1.amazonaws.com/jobConnect/newJob>

{

  "Login\_Id": "Ranjan100",

  "Job\_Id": "240805182309",

  "Job\_Type": "Home Nurse",

  "Date\_From": "20-Sep-2024 8:00",

  "Date\_To": "24-Oct-2024 18:00",

  "Job\_Description": "Home nurse required for a lady senior citizen who is coming from Australia for a week. Female nurse preferred.",

  "Job\_Openings": "1",

  "Job\_Status": "Available",

  "Job\_City": "Yerpedu",

  "Job\_Pincode": "517619",

  "Job\_State": "Andhra Pradesh",

  "Job\_Street": "Kamma Street",

  "Lattitude": "",

  "Longtitude": "",

  "Payment\_Frequency": "Daily",

  "Wage\_Minimum": "Rs. 650",

  "Wage\_Maximum": "Rs. 750",

  "Name\_JP": "Ranjan100",

  "Mobile\_Number\_JP": "9840690980",

  "Email\_JP": " ranjan.murali82@gmail.com "

}

**Get all the Job\_Types:**

API URL: <https://dpezos8c2l.execute-api.ap-south-1.amazonaws.com/JobConnect/jobTypes>

**Get Job Details:**

API URL: <https://w1n2r0iaaa.execute-api.ap-south-1.amazonaws.com/JobConnect/getJobs>

“City\_Filter” option available only for Job Seekers to filter the jobs available in a particular city.

Job Provider can view the jobs with status in Job\_Status\_List = ["AVAILABLE", "READY", "DELETED"] that are relevant to him.

Both Job Provider and Job Seekers should be able to filter and view the jobs with the below status that are relevant to them.

Job\_Progress\_Status\_List = ["REQUESTED", "ACCEPTED", "REQUEST\_CANCELLED", "ACCEPT\_CANCELLED", "DECLINED"]

While creating a new Job, the Job\_Openings should be >= 1 and should not be zero or negative value. So we can show a default value of 1 for Job\_Openings while creating a new job and allow edit but validate for non-zero or negative value when the Job Poster submits the job.

For Job Seeker, the input format is: (Please note the City\_Filter is a mandatory field)

{

"Login\_Id": "Hiji1490",

"Login\_Type": "JS",

"Job\_Status": "AVAILABLE",

"Jobs\_Filter": [“PAINTER”],

"City\_Filter":[]

}

For Job Provider, the input format is:

{

"Login\_Id": "Ranjan100",

"Login\_Type": "JP",

"Job\_Status": "AVAILABLE",

"Jobs\_Filter": [

"PAINTER",

"PAINTER",

"PLUMBER",

"CARPENTER",

"HOME NURSE"

]

}

Response Example for success scenario :

“statusCode” : 200

"Status": "Success",

    "body": "[

{"Job\_Description": "Home nurse required for a lady senior citizen who is coming from Australia for a week. Female nurse preferred.", "Longtitude": "", "Name\_JP": "Ranjan100", "Job\_Openings": "1", "Mobile\_Number\_JP": "9840690980", "Job\_Id": "240805182309", "Job\_City": "YERPEDU", "Wage\_Minimum": "Rs. 650", "Date\_From": "20-Sep-2024 8:00", "Date\_To": "24-Oct-2024 18:00", "Job\_Street": "Kamma Street", "Job\_Type": "HOME NURSE", "Wage\_Maximum": "Rs. 750", "Job\_State": "ANDHRA PRADESH", "Job\_Status": "AVAILABLE", "Email\_JP": "madhusudhanan.moorthy@gmail.com", "Lattitude": "", "Job\_Pincode": "517619", "Payment\_Frequency": "Daily", "Login\_Id": "Ranjan100"},

{"Job\_Description": "Home nurse required for a lady senior citizen who is coming from Australia for a week. Female nurse preferred.", "Name\_JP": "Ranjan Roy", "Longtitude": "", "Job\_Openings": "1", "Mobile\_Number\_JP": "9840690980", "Job\_Id": "240805182500", "Job\_City": "YERPEDU", "Wage\_Minimum": "Rs. 450", "Date\_From": "20-Sep-2024 8:00", "Date\_To": "24-Oct-2024 18:00", "Job\_Street": "Kamma Street", "Job\_Type": "PAINTER", "Wage\_Maximum": "Rs. 450", "Job\_Status": "AVAILABLE", "Job\_State": "ANDHRA PRADESH", "Email\_JP": "madhusudhanan.moorthy@gmail.com", "Lattitude": "", "Payment\_Frequency": "Daily", "Job\_Pincode": "517619", "Login\_Id": "Ranjan100"}

]"

**Job Progress Update:**

API URL: https://60dcdk6lff.execute-api.ap-south-1.amazonaws.com/JobConnect/[**updateJobProgress**](https://ap-south-1.console.aws.amazon.com/apigateway/main/apis/60dcdk6lff/stages?api=60dcdk6lff&experience=rest&region=ap-south-1)

Input Example:

{

"Job\_Id": "240805182309",

"Login\_Id\_JS": "Hiji1490",

"Login\_Id\_JP": "Ranjan100",

"Mobile\_Number\_JS": "8395999965",

"Mobile\_Number\_JP": "9840690980",

"Date\_From": "10-10-2024",

"Date\_To": "15-10-2024",

"Job\_Pincode": "517619",

"Job\_Type": "PAINTER",

"Job\_Status": "REQUESTED",

"Name\_JP": "Rajini",

"Name\_JS": "Jeeva",

"Payment\_Frequency": "Daily",

"Requested\_Wage": "550",

"Wage\_Maximum": "750",

"Email\_JP": "madhusudhanan.moorthy@gmail.com",

"Email\_JS": "madhusudhanan.moorthy@gmail.com",

"Lattitude": "",

"Longtitude": ""

}

Example Job Progress Update fields from Front-end to Back-end

{

"Job\_Id": event["Job\_Id"],

"Login\_Id\_JS": event["Login\_Id\_JS"],

"Login\_Id\_JP": event["Login\_Id\_JP"],

"Mobile\_Number\_JS": event["Mobile\_Number\_JS"],

"Mobile\_Number\_JP": event["Mobile\_Number\_JP"],

"DateFrom": event["DateFrom"],

"DateTo": event["DateTo"],

"Job\_Address\_Pincode": event["Job\_Address\_Pincode"],

"Job\_Type": event["Job\_Type"],

"Job\_Status": event["Job\_Status"].upper(),

"Name\_JP": event["Name\_JP"],

"Name\_JS": event["Name\_JS"],

"Payment\_Frequency": event["Payment\_Frequency"],

"Requested Wage": event["Requested Wage"],

"Wage\_Maximum": event["Wage\_Maximum"],

"Email\_JP": event["Email\_JP"],

"Email\_JS": event["Email\_JS"],

"Lattitude": event["Lattitude],

"Longtitude": event["Longtitude"]

}

Team:

Dhanya

Niharika

Pranati

Indu

Sowmyasri

Srikrishna M