Lecture Notes

Design and Prototyping

Introduction

In this module, you learnt about the process that is followed before developing the required web application and after completing the requirement gathering phase. You understood the concepts of design and prototyping in relation to the application that we are developing. You also understood why designing is an important part of the software development process.

Introduction to Design and Prototyping

It is recommended to design the outline of the application and subsequently its prototype before directly proceeding to develop the final product. Building a basic prototype could also give us scope for improvements that were previously unknown.

When it comes to building great products, design is considered the most important 'feature'. Whether you are a designer, developer, product manager, program manager, marketing manager or project manager, it is essential to understand the product development process in order to create your best product.

While thinking about products or features, designers should understand their objectives and be able to answer the following questions:

- What problem are we solving?
- Who has this problem?
- What do we want to achieve?

Answering these questions helps designers understand the user experience of a product as a whole, not purely the interaction (feel) or visual (look) part of the design. Only after answering these questions will it make sense to find a solution for the problem.

Designing a prototype will help you in correctly utilising the resources and effort while developing a project. Prototyping not only provides a clear picture of what you, as a software developer, would need to build but also certain improvements over the existing design flow.

Design

The design process involves a series of steps that product teams follow during the formulation of a product, from start to end. A solid, well-structured process is essential for two reasons: it helps you to stay focused and on schedule.

By building a design for your application, you realise that the scope of the project is not just limited to the requirement. Of course, you need to develop the required components correctly, but there are a number of ways to expand the same project.

At the design stage, a prototype is first created–a black-and-white layout representing a simplified site scheme. It contains all the main elements in the form of blocks, so the client can evaluate the basic concept of the product.

Prototyping not only helps you identify and formulate the main direction of the design but also saves a significant amount of time. After spending a day creating a concept, you do not have to spend a week developing a new site layout.

Prototyping makes it possible to concertise an idea and understand which aspects of the idea would be difficult or impossible to implement. Creating a prototype can highlight unanticipated physical, technical or financial constraints. Therefore, at the prototyping stage, it is possible to identify unnecessary elements that are best left abandoned.

Technology and Development

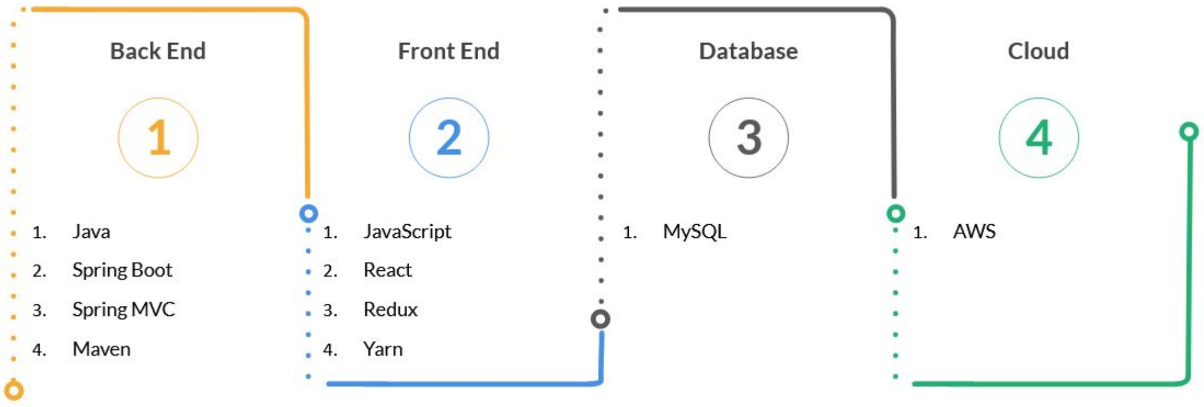
Technologies

The concept of design and prototype can be solidified only if you apply the same to your project as well. There are a number of technologies used in developing the UPSTAC application.

These technologies can be broadly classified into the following four categories on the basis of the usage:

- Back end
- Front end
- Database
- Cloud deployment

TECHNOLOGIES USED – UPSTAC APPLICATION



Back End
1
1. Java
2. Spring Boot
3. Spring MVC
4. Maven

Front End
2
1. JavaScript
2. React
3. Redux
4. Yarn

Database
3
1. MySQL

Cloud
4
1. AWS

Back-End Tech Stack

Java: Java is a class-based, object-oriented programming language and one of the most popular programming languages. It is a general-purpose programming language intended to allow software developers to write once and run anywhere. This is possible because Java is platform-independent.

Spring: Spring is a very popular Java-based framework used for building web and enterprise applications. Spring was primarily created as a framework wherein all objects that were required by a particular object (dependencies) could be passed (injected) via a Dependency Injection container and later added with a huge number of features. However, in order to utilise all the features or modules present in Spring, you need to configure them separately and find compatible libraries, which again increases the amount of manual work and, thereby, contradicts the use of a framework to reduce the manual effort in developing a web application. To tackle this issue, the makers of Spring came up with a simpler and updated version of the framework called Spring Boot.
Spring Boot performs tasks and completes work automatically for you and allows you to override the defaults if needed.

Maven: Maven is a build automation tool used primarily for Java projects. Maven can handle the numerous dependencies required by Spring, instead of manually adding them.

Front-End Tech Stack

JavaScript: HTML and CSS are languages that give structure and style to web pages, whereas JavaScript gives web pages interactive elements that engage a user. JavaScript is the only programming language that is native to the web browser and is also one of the most popular languages.

React: React.js is an open-source JavaScript library that is used for building user interfaces, specifically for single-page applications. A single-page application (SPA) is a web application or website that interacts with the web browser by dynamically rewriting the current web page with new data from the web server, instead of the default method wherein the web browser loads entire new pages. React is used for handling the UI of web and mobile apps. It also allows us to create reusable UI components. React allows developers to create large web applications that can change data without the need for reloading the page. The main advantages of using React is that it is fast, scalable and simple.

Database

MySQL: MySQL is a relational database management system based on SQL (Structured Query Language). The application is used for a wide range of purposes, including data warehousing, e-commerce and logging applications. However, ySQL is most commonly used as a web database. It can be used to store anything, from a single record of information to an entire inventory of available products for an online store.

Cloud Deployment

AWS (Amazon Web Services) is a comprehensive, cloud computing platform provided by Amazon. AWS includes various tools and services to help users migrate applications, databases, servers and data to the cloud. It also provides command-line tools and built-in software development kits (SDKs) to deploy and manage applications and services.

Status codes play an extremely important role during the development phase. Some of the commonly used status codes are as follows:

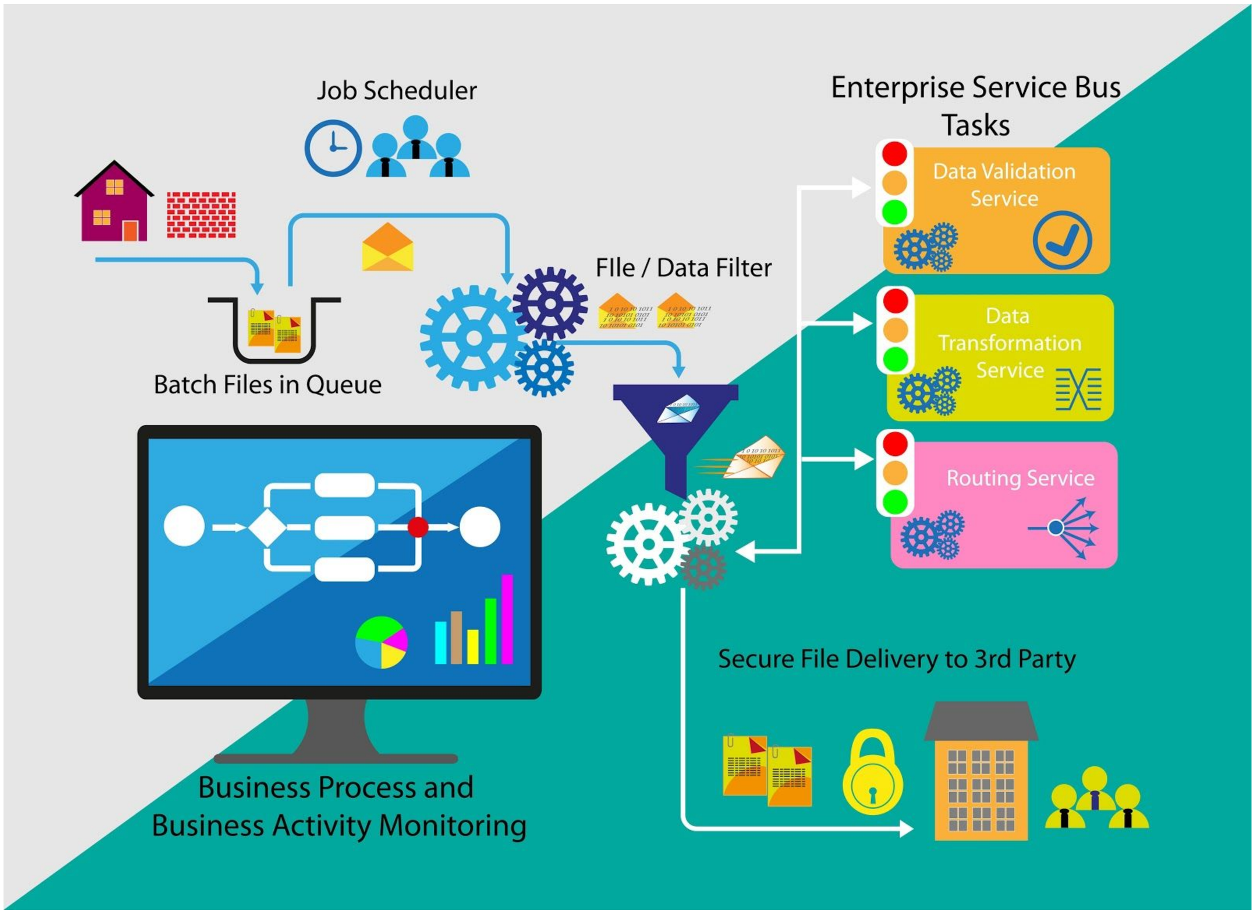
- **2XX Successful**: This class of status code indicates that the client's request was successfully received, understood and accepted.
- **200 OK**: This indicates that the request has succeeded.
- **4XX Client Error**: The 4xx class of status code is intended for cases in which the client seems to have erred.
- **400 Bad Request**: This indicates that the request could not be understood by the server due to incorrect syntax.
- **404 Not Found**: This indicates that the server has not found anything that matches the Request-URI.
- **5XX Server Error**: Response status codes beginning with the digit '5' indicate that the server is aware that it has erred or is incapable of performing the request.
- **500 Internal Server Error**: This indicates that the server has encountered an unexpected condition that prevented it from fulfilling the request.

Architecture

Software architecture refers to the fundamental structures of a software system and the discipline of creating such structures and systems. Each structure comprises software elements, the relations among them and the properties of both elements and relations.

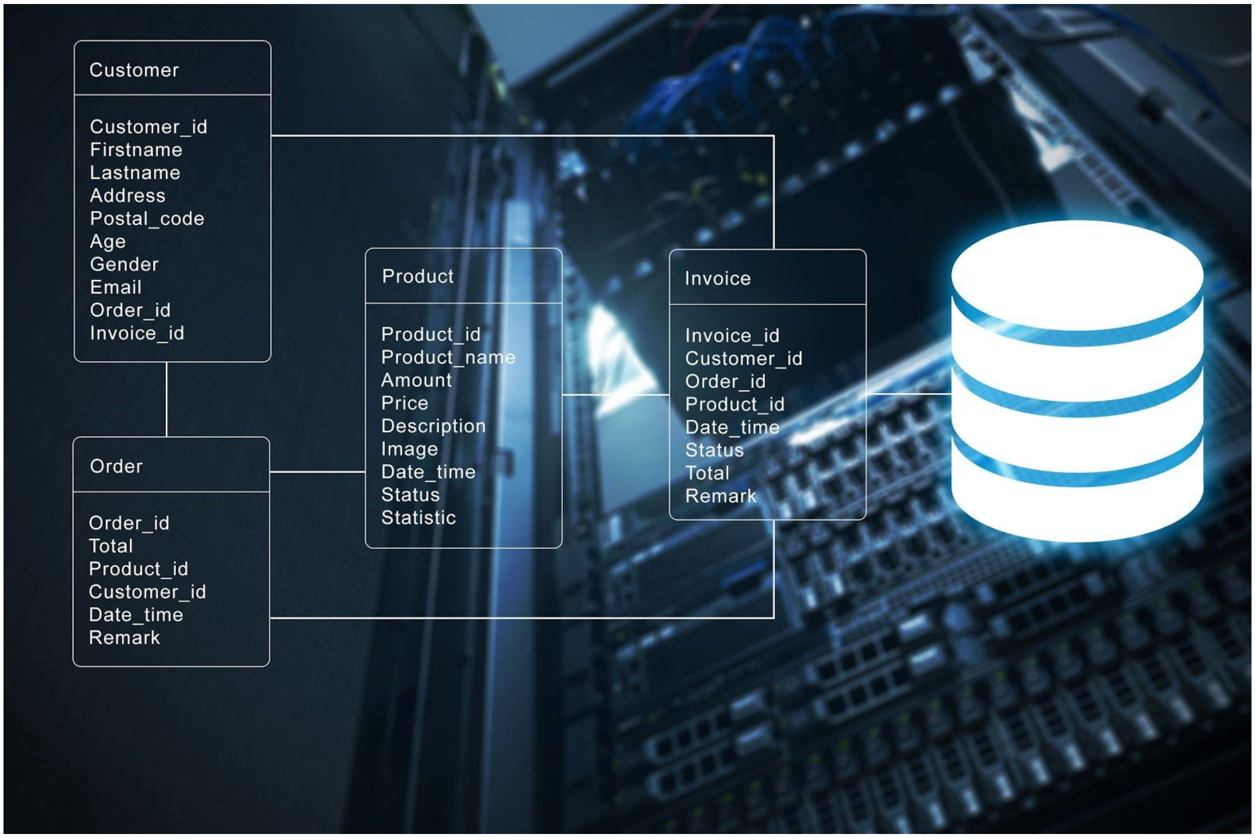
Software architecture creates a concrete foundation for your project in order to make it quantifiable and scalable. The architecture of your software defines the qualities of your system in terms of security, scalability, performance, profitability and much more.

The good thing about creating viable software architecture is that it helps you identify risks early in the process.


Job Scheduler
Enterprise Service Bus Tasks
Data Validation Service
Data Transformation Service
Routing Service
FIle / Data Filter
Batch Files in Queue
Business Process and Business Activity Monitoring
Secure File Delivery to 3rd Party

There are no sharp distinctions between software architecture and design. They are all part of the 'chain of intentionality' from high-level intentions to low-level details. Therefore, software architecture is considered an 'intellectually graspable' abstraction of a complex system.
The software architecture artefacts facilitate communication between the stakeholders. All of these stakeholders have their own concerns with respect to the system. Balancing these concerns and demonstrating how they are addressed is part of designing the system.

A database is an organised collection of data, generally stored and accessed electronically from a computer system. Databases are complex and often developed using formal designing and modelling techniques.
Database management system (DBMS) is a software that interacts with end users, applications, and the database itself to capture and analyse data.



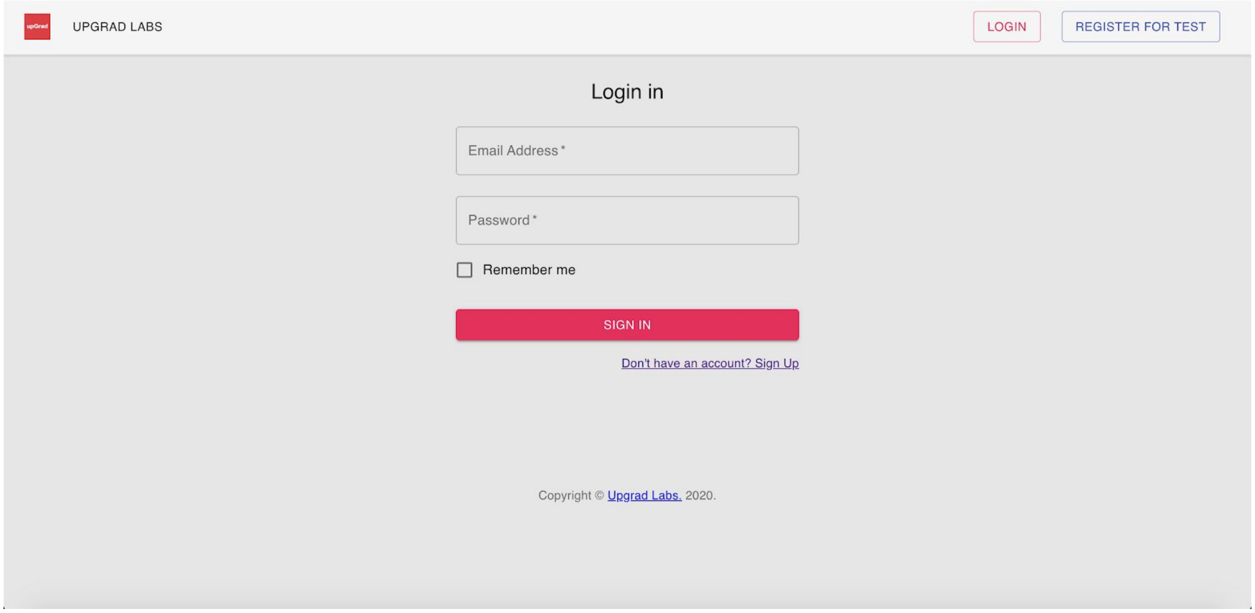
Well-designed databases are easy to maintain, improve data consistency and are cost-effective in terms of disk storage space. A database designer decides how the data elements should correlate and what data must be stored.

UPSTAC Demonstration

UPSTAC is a healthcare application that is designed to help users get themselves tested easily by simply requesting for a test on the application. The testers can view these test requests as soon as they are, assign the requests to themselves, perform the tests and update the results on the application. Doctors can check the results on the application and provide appropriate prescriptions based on the patients' conditions.

Let's understand the workings of the UPSTAC application in detail.

Users can choose one of the given three roles: patient, tester and doctor, as shown in the images below



Login in

Email Address *

Password *

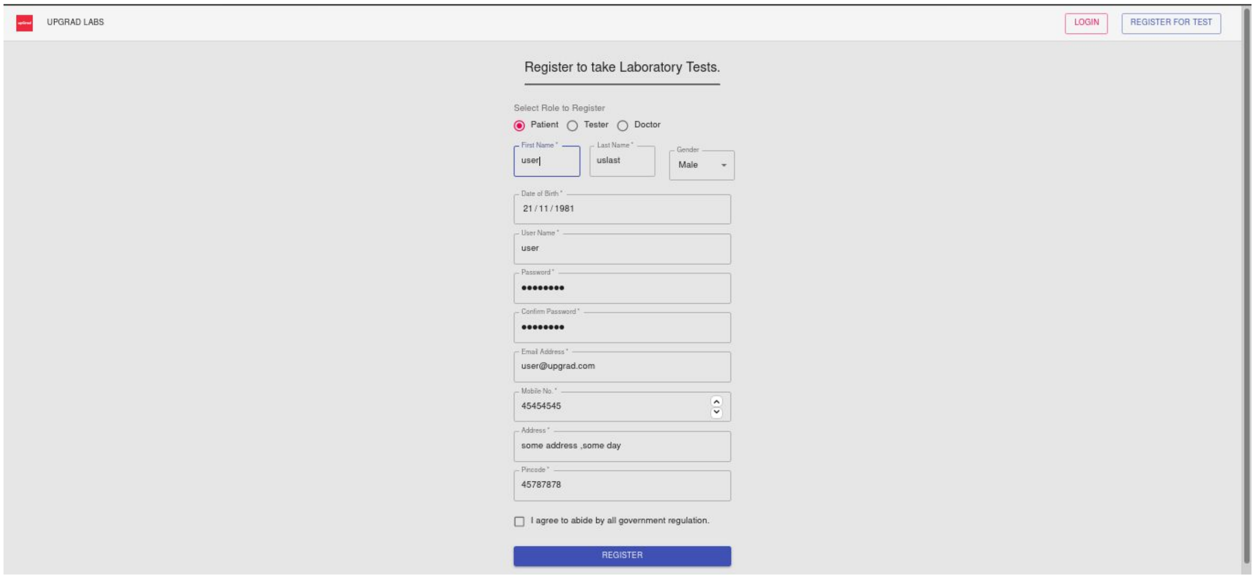
□ Remember me

SIGN IN

Don't have an account? Sign Up

Copyright © Upgrad Labs. 2020.

Login Page



Register to take Laboratory Tests.

Select Role to Register

● Patient ○ Tester ○ Doctor

First Name * user	Last Name * uslast	Gender Male

Date of Birth *: 21 / 11 / 1981

User Name *: user

Password *: ●●●●●●●●

Confirm Password *: ●●●●●●●●

Email Address *: user@upgrad.com

Mobile No. *: 45454545

Address *: some address ,some day

Pincode *: 45787878

□ I agree to abide by all government regulation.

REGISTER

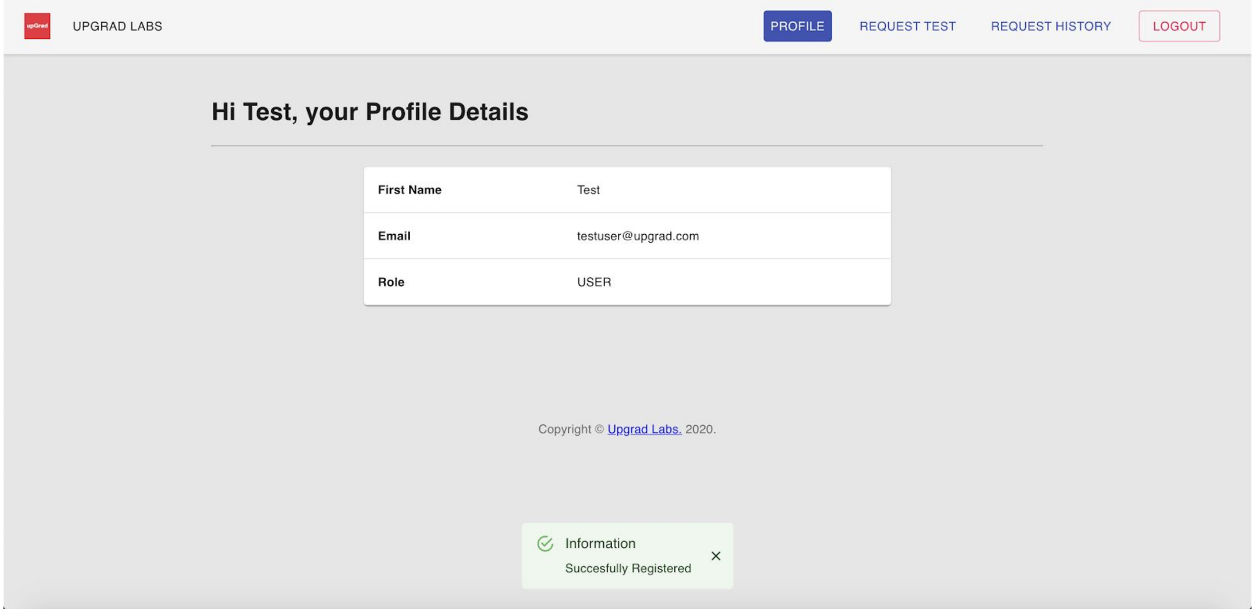
Registration Page

Patient

Users can register themselves as a patient. Once the patient is registered, their details are stored inside the user information and user role tables.

Patients can request for a test and fill out the required information. After the patients have raised a request, it is sent to all the testers so that any one of them can assign the request to themselves. The patients can view the status of their test request as initiated.

The patients should not be able to spam by requesting for tests repeatedly. They should be able to request for a test only after the previously requested test is completed.



Hi Test, your Profile Details

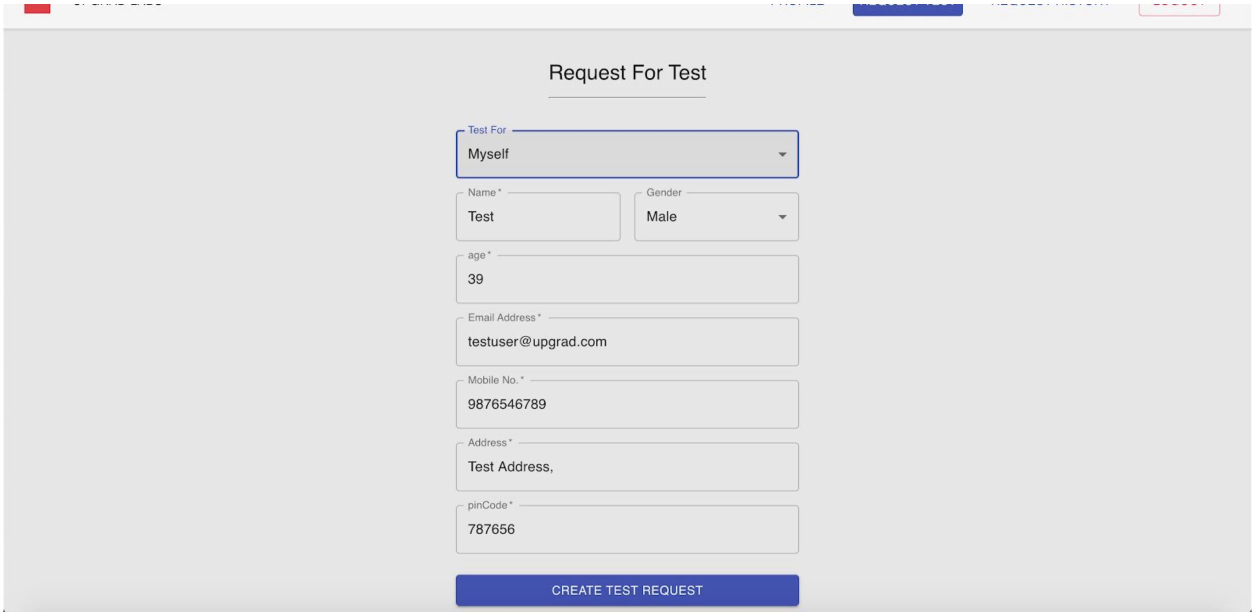
First Name	Test
Email	testuser@upgrad.com
Role	USER

Copyright © Upgrad Labs. 2020.

Information

Succesfully Registered

Profile Details of Patients



Request For Test

Test For: Myself

Name *: Test

Gender: Male

age *: 39

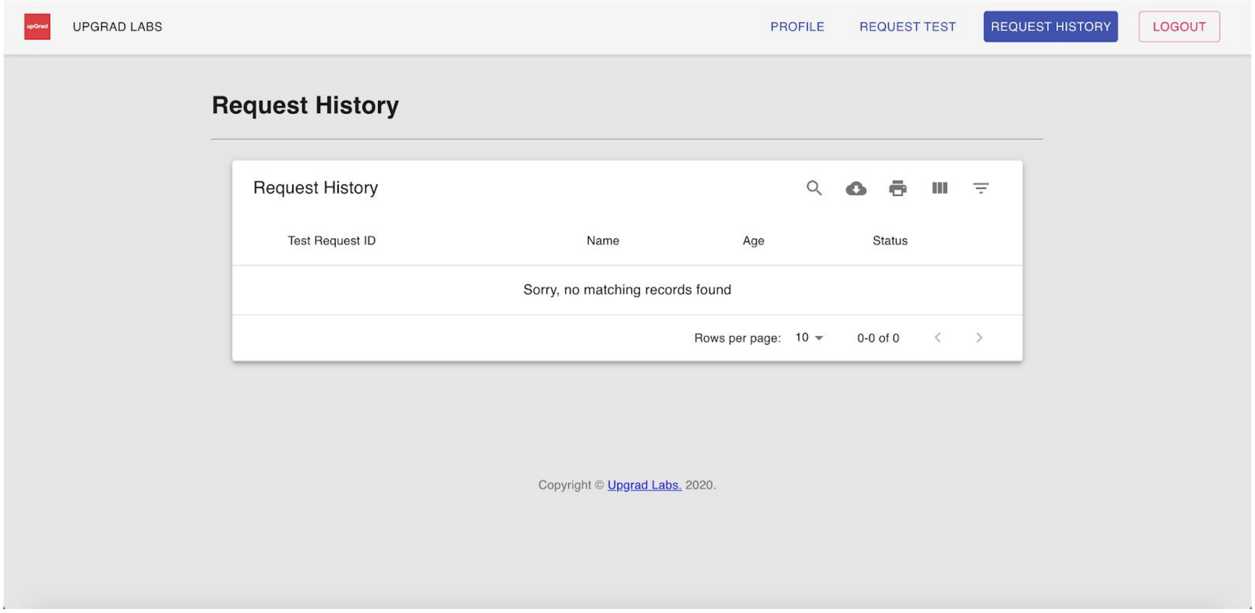
Email Address *: testuser@upgrad.com

Mobile No. *: 9876546789

Address *: Test Address,

pinCode *: 787656

Test Request Screen for Patients



Request History

Request History

Test Request ID	Name	Age	Status
Sorry, no matching records found			

Rows per page: 10 0-0 of 0

Copyright © Upgrad Labs. 2020.

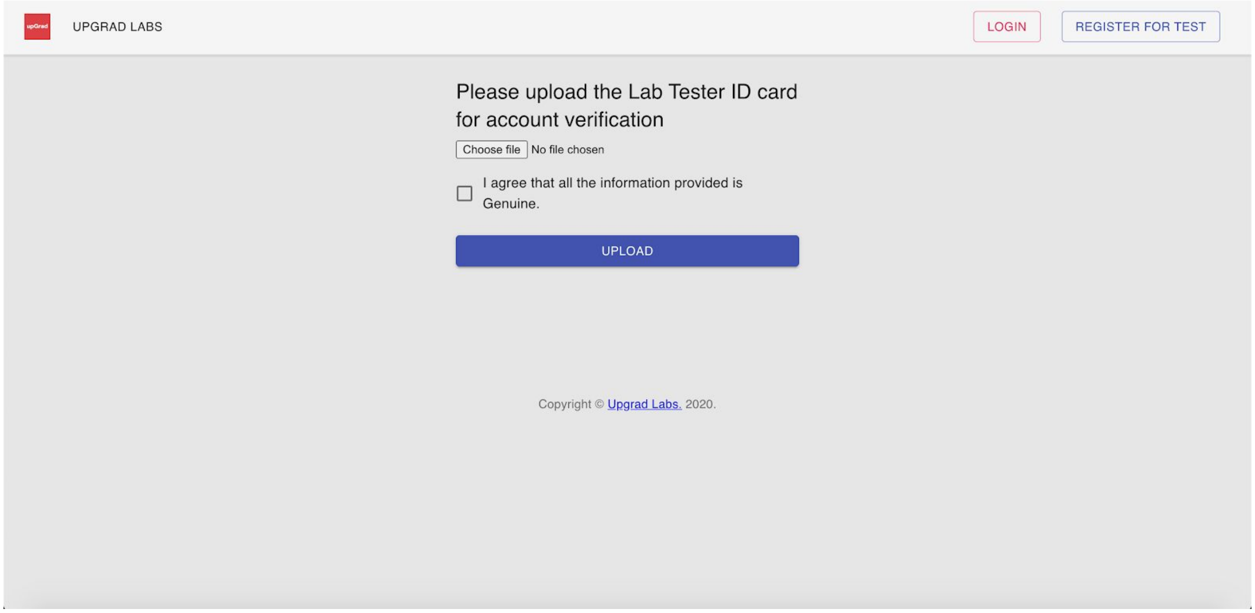
Request History Screen of Patients

Tester

Testers should also be able to register themselves, but this process should be authorised to prevent unauthorised users to register themselves as testers and misuse tester privileges.

After the testers are registered and approved by the government authority, which acts as the super admin, the testers can view the requested tests and assign them to themselves.

The testers then test the patients and update their lab results. As the testers update the test results, two things should happen: the status of the test request should change from ‘initiated’ to ‘lab test completed’ and the test request should be passed on to the doctor for consultation.



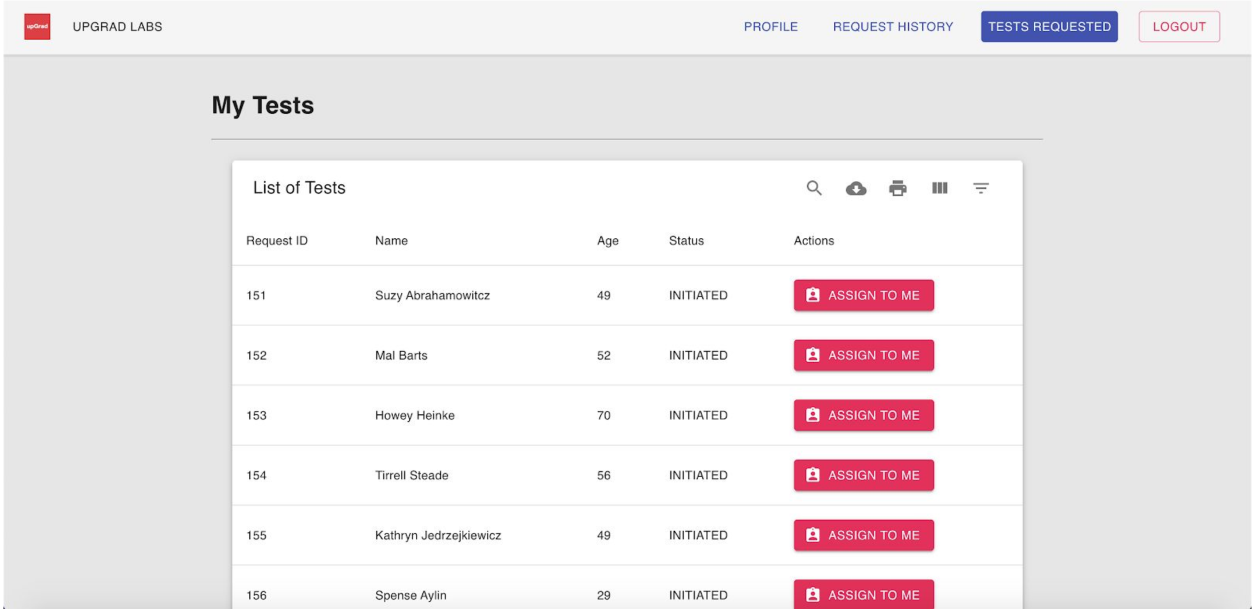
Please upload the Lab Tester ID card for account verification

No file chosen

□ I agree that all the information provided is Genuine.

Copyright © Upgrad Labs. 2020.

Document Upload Screen for Testers



My Tests

List of Tests

Request ID	Name	Age	Status	Actions
151	Suzy Abrahamowitcz	49	INITIATED	ASSIGN TO ME
152	Mal Barts	52	INITIATED	ASSIGN TO ME
153	Howey Heinke	70	INITIATED	ASSIGN TO ME
154	Tirrell Steade	56	INITIATED	ASSIGN TO ME
155	Kathryn Jedrzejkiewicz	49	INITIATED	ASSIGN TO ME
156	Spense Aylin	29	INITIATED	ASSIGN TO ME

Test Requested Screen for Testers

Doctor

The flow of the doctor would also be more or less the same as that of the testers. Doctors should be approved in order to prevent misuse of information. The doctors should assign the ‘lab test completed’ requests to themselves, view the test data and provide prescriptions.

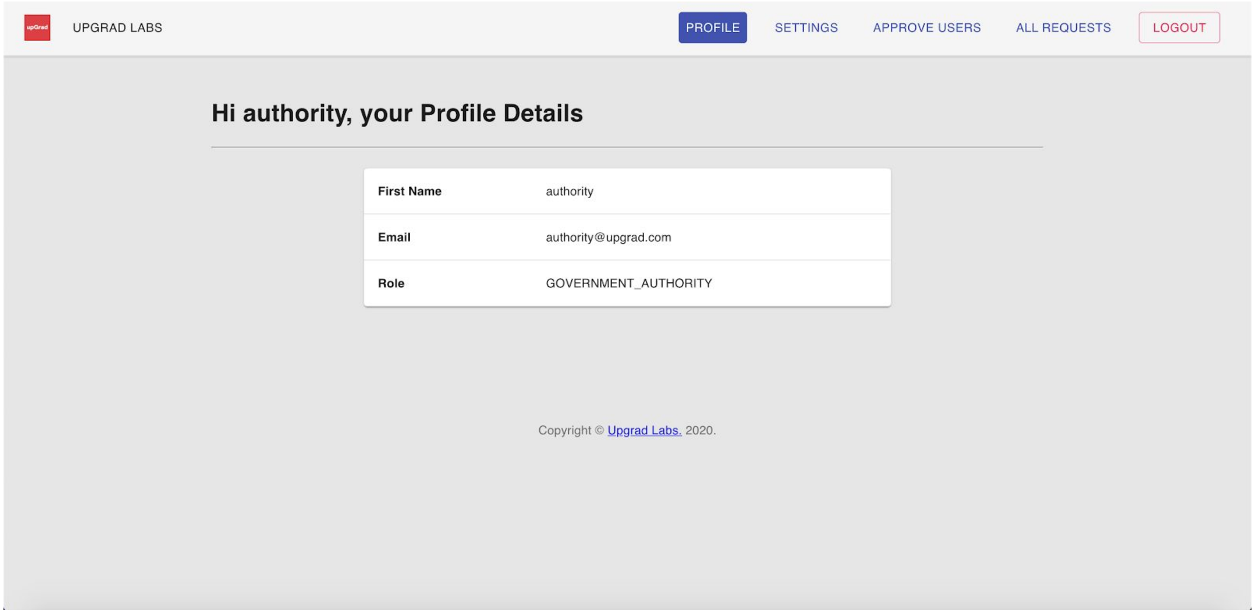
After the doctors have updated the test prescription requests, the patients should be able to view the request status as ‘completed’.

Government Authority

The government authority would be the admin here. It would approve the testers and doctors after going through their uploaded credentials and verifying them.It would be able to view all the requests that have been raised and all the information about the requests such as who made the request, who was the tester, who was the doctor and what were the test details.

The government authority requires all the information to design policies and allocate resources nationally or for a particular area.

Visual representations in the form of pie charts and bar graphs are available to aid data analysis. Also, the super admin can set a certain threshold of positive cases for any disease in any pincode based on the population density of a region so that the area can be marked as different zones according to its threat level.

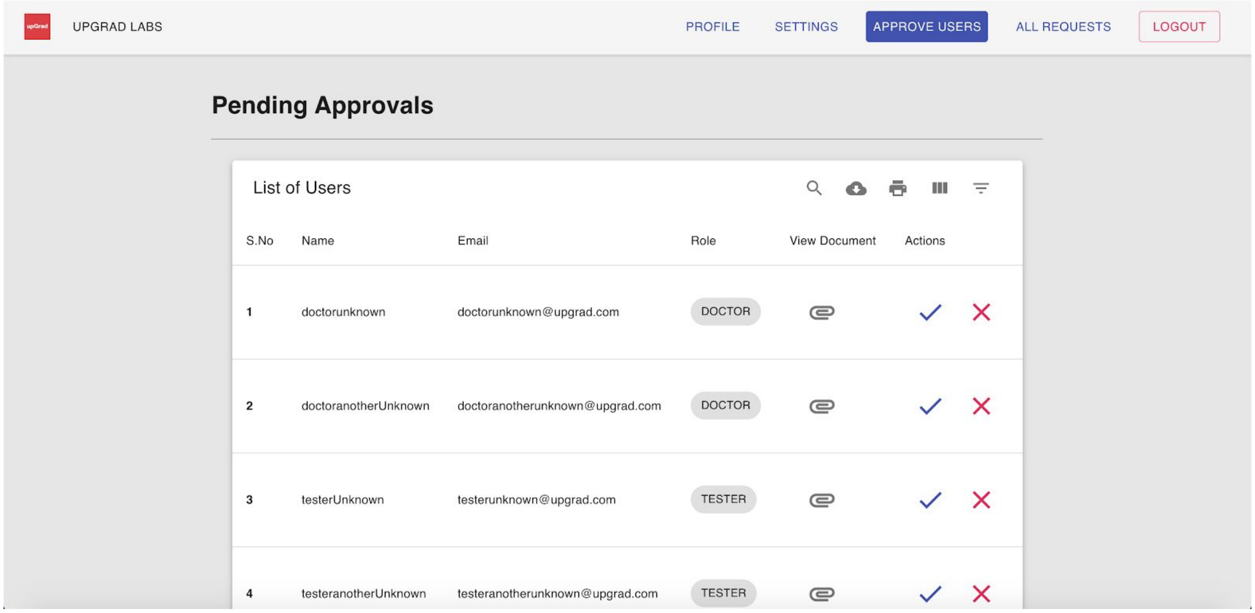


Hi authority, your Profile Details

First Name	authority
Email	authority@upgrad.com
Role	GOVERNMENT_AUTHORITY

Copyright © Upgrad Labs. 2020.

Profile Page for Government Authority

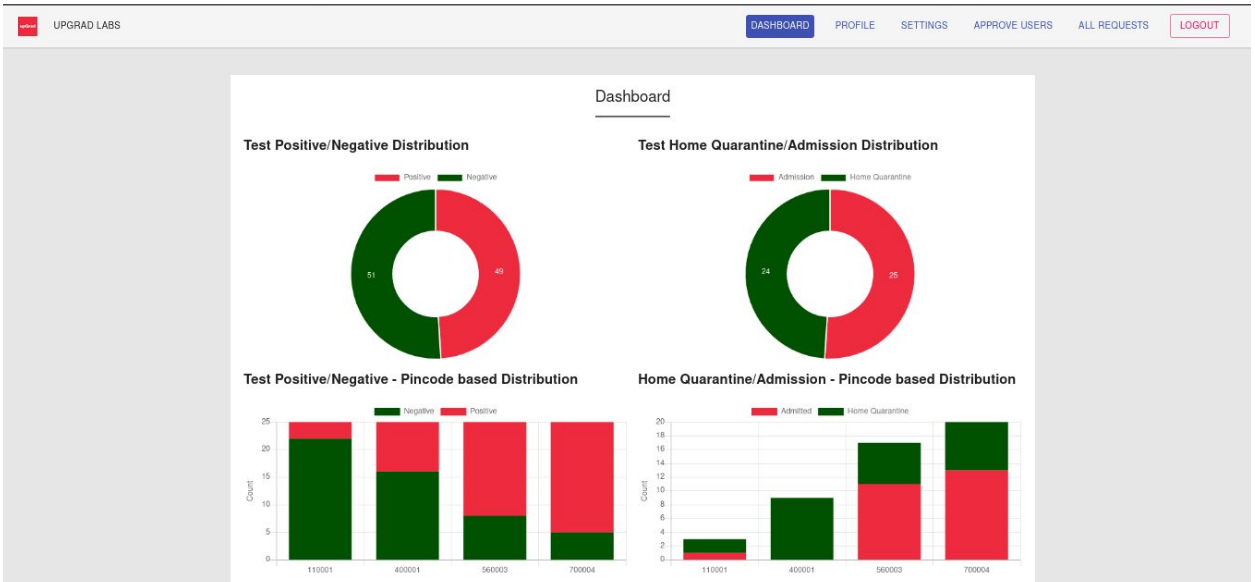


Pending Approvals

List of Users

S.No	Name	Email	Role	View Document	Actions
1	doctorunknown	doctorunknown@upgrad.com	DOCTOR		✓ ✕
2	doctoranotherUnknown	doctoranotherunknown@upgrad.com	DOCTOR		✓ ✕
3	testerUnknown	testerunknown@upgrad.com	TESTER		✓ ✕
4	testeranotherUnknown	testeranotherunknown@upgrad.com	TESTER		✓ ✕

Approve Users Screen for Government Authority



Dashboard for Government Authority

UPSTAC Backend

To run the backend application, you need to import the project path to IntelliJ and run **UpstacApplication.java.**

If the dependencies such as Java 8, Maven and MySQL are correctly installed, the application would start running after displaying log messages.

In order to run the frontend code you need to install the yarn package manager using the following two commands:

- npm install -g yarn
- yarn install

Installation of yarn requires npm or more specifically NodeJs to be installed in your system. Yarn installation is a one time step. You won't have to install yarn every time you need to run any React code. To run the react code you need to run the following code in the folder containing the frontend code:

- yarn start

Once this command runs, yarn will start the local server, open the browser and display the functioning frontend of the application. You need not open the browser or type any link.

Disclaimer: All content and material on the upGrad website is copyrighted material, either belonging to upGrad or its bonafide contributors and is purely for the dissemination of education. You are permitted to access, print and download extracts from this site purely for your own education only and on the following basis:

- You can download this document from the website for self-use only.
- Any copies of this document, in part or full, saved to disk or to any other storage medium may only be used for subsequent, self-viewing purposes or to print an individual extract or copy for non-commercial personal use only.
- Any further dissemination, distribution, reproduction, copying of the content of the document herein or the uploading thereof on other websites or use of content for any other commercial/unauthorised purposes in any way which could infringe the intellectual property rights of upGrad or its contributors, is strictly prohibited.
- No graphics, images or photographs from any accompanying text in this document will be used separately for unauthorised purposes.
- No material in this document will be modified, adapted or altered in any way.
- No part of this document or upGrad content may be reproduced or stored in any other web site or included in any public or private electronic retrieval system or service without upGrad's prior written permission.
- Any rights not expressly granted in these terms are reserved.