Low Level Design

Exam Proctor

Contents

1 Introduction

1.1 What is a Low-Level design document?

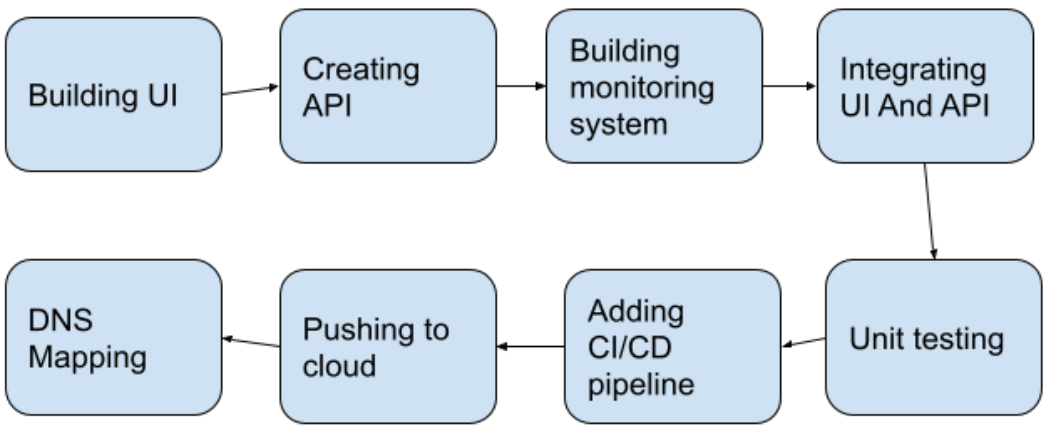
The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code Exam proctor(SLAT) . LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

1.2 Scope

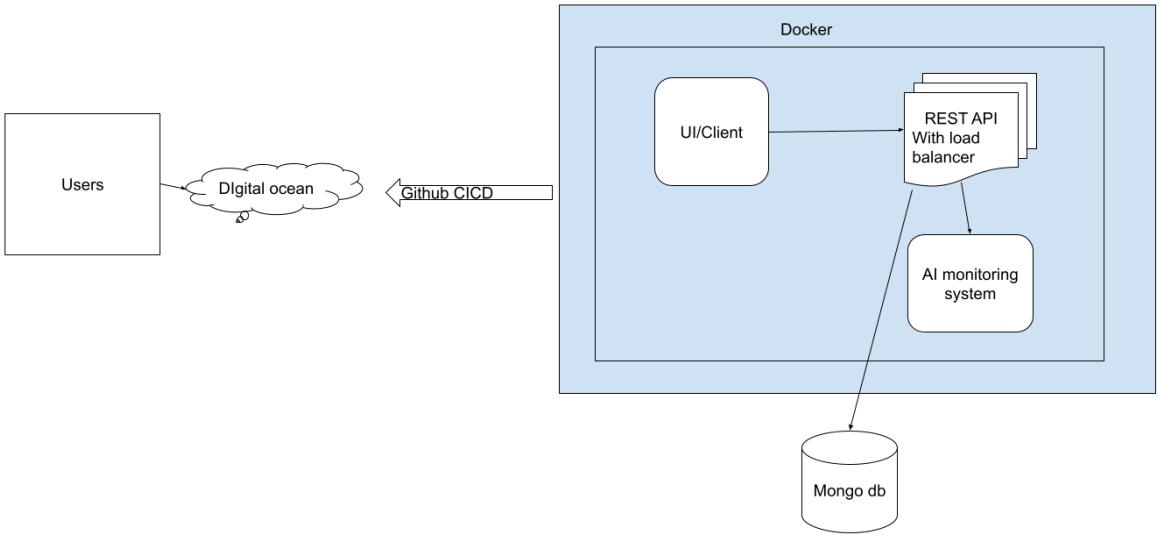
Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work

2 Architecture

2.1 Development flow:



2.2 Architecture diagram:



3 Architecture Description

3.1 UI/Development:

- React js along with bootstrap is used for front end development. The main reason for going with react is for the reusability and light weight and bootstrap came out of the box for styling.
- Draft.js is being used as an inline text editor to create courses.
- Fetch and axios is used for making API interactions

3.2 Backend development

For backend i have used express and with node for creating REST API which will interact seamlessly with the UI and Mongo

3.3 AI monitoring system

For monitoring exams i have used python along with deepface and mediapipe and exposed it in REST API using flask.

3.4 Database

I have used mongo db Atlas which is a managed cloud database for storing and retrieving data.

3.5 Containerization

I have used docker and docker compose to containerize the application.

3.6 GIT and CI/CD:

I have used git for code maintenance and github actions to perform CI/CD operations

3.7 Deployment:

I have used the following for deployment purposes.

- Digital ocean droplets - Deploying our application
- Caddy - For Https, reverse proxy and load balancer
- Godaddy - For name server and domain

4 Unit Test Cases

Test Case Description	Pre-Requisite	Expected Result
Adding new user	Application URL should be defined	Should be able to add user of both type creator and learner
Login as creator	Application URL should be defined Creator should have created an account	Should be able to navigate to the dashboard page
Login as learner	Application URL should be defined Learner should have created an account	Should be able to navigate to the dashboard page
Trying to login with wrong password	Application URL should be defined	Error notification should be show to the user
Create course as Creator	Application URL should be defined User should have created account as a Creator	Should be able to add a course and course should be visible in the dashboard Should be able to navigate to the course page
Add Content	Application URL should be defined User should have created account as a Creator	Creator should be able to navigate to the course page Create should be able to add course title and

	Navigate to the course page	enter the content in the text editor And when submitted he should be navigated to the content page see the content listed there
View content	Application URL should be defined User should have created account as a Creator Navigate to the course page	User to select the add resource button and navigate to content page Click the content and user should be able to view the content in read only mode
Edit content	Application URL should be defined User should have created account as a Creator Navigate to the course page	User to select the add resource button and navigate to content page Click the edit button and user should be able to view the content in view and edit only mode Once edited and changed the content will be reflected
Delete Content	Application URL should be defined User should have created account as a Creator Navigate to the course page	User to select the add resource button and navigate to content page Click the delete button and user should be able to view the content in view and

		edit only mode Once deleted the content will be removed from the dashboard
Add/edit Exam and exam time	Application URL should be defined User should have created account as a Creator Navigate to the course page	User should select Add exam Should taken to page where he can add/ edit questions and edit time Once edited the needed fields it will be reflect the question listed side
View content as a learner	Application URL should be defined User should have created account as a learner Navigate to the course page	User should be able to click and view content in read only mode
Enrol in exam	Application URL should be defined User should have created account as a learner Navigate to the course page Select Enroll course	A pop with razorpay dialogue box will appear. Once payment is successful status will be changed to take exam

Search according to categories	Application URL should be defined User should have created account as a learner In the dashboard there should be a search options	You be able to click search button and search course according the role which is needed
Take exam	Application URL should be defined User should have created account as a learner User should submit proper image with face in submit portal	User will be navigated to exam page Timer will be present Web camera will be enabled Should be able to answer the questions Page will redirect and you should be able to download certificate incase if you passed Exam will auto submit if you perform any malpractice