

THESIS COMPLETION CERTIFICATE

Student Name: Sarmad Ashfaq Choudary Enrolment No: 01-131182-033

Student Name: Manal Talat Enrolment No: 01-131182-046

Programme of Study: Bachelor of Software Engineering

Project Title: PrepExpert

It is to certify that the above students' project has been completed to my satisfaction and to my belief, its standard is appropriate for submission for evaluation. I have also conducted a plagiarism test of this thesis using HEC prescribed software and found a similarity index at 7% that is within the permissible limit set by the HEC. I have also found the thesis in a format recognized by the department.

Supervisor's Signature: _____

Date: 04-July-2022

Name: Dr Raja M. Suleman

CERTIFICATE OF ORIGINALITY

This is certify that the intellectual contents of the project **PrepExpert** are the product of my/our own work except, as cited properly and accurately in the acknowledgements and references, the material taken from such sources as research journals, books, internet, etc. solely to support, elaborate, compare, extend and/or implement the earlier work. Further, this work has not been submitted by me/us previously for any degree, nor it shall be submitted by me/us in the future for obtaining any degree from this University, or any other university or institution. The incorrectness of this information, if proved at any stage, shall authorise the University to cancel my/our degree.

Name of the Student: Sarmad Ashfaq Choudary

Signature: _____

Date: 04-July-2022

Name of the Student: Manal Talat

Signature: _____

Date: 04-July-2022

Abstract

Effective learning is having sound knowledge and a clear understanding of concepts which should not be confused with cramming information. The typical assessment systems provide MCQs in a sequential manner that does not judge the students' knowledge, instead, provides information to be crammed. Computerised Adaptive Testing (CAT) has revolutionised the assessment methods by providing tailored questions according to the student's ability level. Our goal is to implement computerised adaptive testing (CAT) for the Software Engineering department so that the students are well prepared for their quizzes and exams. Our system measures the latent traits of the students and provides smart insights into their performance. By providing adaptive tests, we provide a personalized learning experience to the students that assess different students on different levels. Our top-notch feature, Guided Learning, provides preparation content to improve the weak areas.

Keywords: CATSIM, Computerised Adaptive Learning, Item Response Theory, Latent Traits, Student's Proficiency, Tailored Testing

Dedication

Praise be to Allah above contentment, desires and for this merciful love.

This hard work is dedicated to our beloved parents for their untiring efforts to see us shine, constant support and uncountable sacrifices that have helped us achieve what we have accomplished so far.

Acknowledgments

*We would like to express sincere gratitude to our supervisor **Dr. Raja M. Suleman** for his keen interest, encouragement and motivation that has helped us develop this project. His valuable feedback and guidance has helped us and motivated us to put our best efforts all along the way.*

We would also like to thank the respected faculty of our department who invested their time into analysing our project, provided valuable suggestions and offered guidance at every step.

Lastly, a warmest gratitude to our friends and classmates for being with us through thick and thin and for their moral support.

Table of Contents

Thesis Completion Certificate	1
Certificate of Originality	2
Abstract	3
Dedication	4
Acknowledgments	5
List of Figures	9
List of Tables	10
Chapter 1	11
Introduction	11
1.1. Motivation	11
1.2. Problem statement or research questions	11
1.3. Objectives	11
1.4. Main contributions	11
1.5. Report organisation	12
Chapter 2	13
Background Study/Literature Review	13
2.1. Latent Traits of Students and their Assessment	13
2.2. Computerised Adaptive Testing	14
2.3. Mathematical Formalisation by Item Response Theory(IRT) Models	14
Chapter 3	15
System Requirements	16
3.1. Use Case Diagram	16
3.2. Functional Requirements	16
3.2.1. Authentication	16
3.2.1.1. Create Account	16
3.2.1.2. Sign In	17
3.2.1.3. Forgot Password	17
3.2.2. Profile Management	18
3.2.2.1. View Account Information	18
3.2.2.2. Update Account Information	18
3.2.3. Test	19
3.2.3.1. Take Test	19
3.2.3.2. View Detailed Performance Report	19
3.2.3.3. Access Guided Learning	20
3.2.3.4. View All Preparation Content	21

3.2.4. Administration	21
3.2.4.1. Manage Admins	21
3.2.4.2. Manage Students and Content Managers	22
3.2.5. Test Management	22
3.2.5.1. Manage Domains	22
3.2.5.2. Manage Subjects	23
3.2.5.3. Manage Topics	23
3.2.5.4. Manage Concepts	24
3.2.5.5. Manage Question Bank	24
3.2.5.6. Manage Linked Video Resources	25
3.2.5.7. Mange Video Resources	25
3.3. Interface Requirements	26
3.3.1. User Interface	26
3.3.2. Physical Interface	26
3.4. Database Requirements	27
3.5. Non-Functional Requirements	27
3.5.1. Performance Requirements	27
3.5.2. Safety Requirements	27
3.5.3. Security Requirements	28
3.5.4. Software Quality Attributes	28
3.6. Project Feasibility	29
3.6.1. Technical Feasibility	29
3.6.2. Operational Feasibility	29
3.6.3. Legal and Ethical Feasibility	30
3.7. Conclusion	31
Chapter 4	32
System Design	32
4.1. Design Approach	32
4.1.1. Backend Implementation	32
4.1.2. Frontend Implementation	32
4.2. Design Constraints	32
4.2.1. System Overload	32
4.2.2. High Bandwidth Server	32
4.3. System Architecture	33
4.4. Logical Design	34
4.5. Dynamic View	35

4.6. Component Design	40
4.7. Data Models	42
4.8. User Interface Design	43
4.9. Conclusion	55
Chapter 5	56
System Implementation	56
5.1. Tools and Technologies	56
5.1.1. FastAPI	56
5.1.2. passlib[bcrypt] hash	56
5.1.3. starlette Middleware	57
5.1.4. NumPy for Scientific Computing	57
5.1.5. SQLAlchemy	57
5.1.6. Redis	58
5.1.7. Chart.js and ApexCharts.js	59
5.2. Algorithms Used	60
5.2.1. catsim	60
5.3. Implementation Strategy	65
5.4. Conclusion	65
Chapter 6	66
System Testing & Evaluation	66
6.1. Test Strategy	66
6.2. Component Testing	66
6.3. Unit Testing	66
6.4. Integrated Testing	66
6.5. System Testing	67
6.6. Test Cases	67
6.7. Results & Evaluation	77
6.8. Conclusion	77
Chapter 7	78
Conclusion	78
7.1. Contributions	78
7.2. Reflections	78
7.3. Future work	78
References	79
Appendix A	80

List of Figures

Figure 3.1: System Use Case Diagram	16
Figure 4.1: System Architecture	33
Figure 4.2: Logical Design	34
Figure 4.3: Deployment View	41
Figure 4.4: Data Models	42
Figure 5.1: Basic CAT Life Cycle	61
Figure 5.2: Get Next Question	63

List of Tables

Table 5.1: Estimator Selector Review	63
Appendix A: Dictionary	80

Chapter 1

Introduction

In this report, we document the details of our final year project, PrepExpert - an intelligent testing system that helps the students with the preparation of their exams. The system uses artificial intelligence for adaptive test generation and focuses on improving weak areas of the students.

1.1. Motivation

We all went through difficult times during the COVID-19 where everyone faced immense difficulty in the exams preparation. On top of that, many concepts were not clear due to the inherent challenges of online learning. The existing preparation systems only provided MCQs in sequential manner with no analysis of the student's performance. Those systems were only a method for cramming information.

Therefore, we had an idea that we should develop our own system that provides a true learning experience. The system generates tests at the runtime which adapts to the student's learning and provides intelligent insights after completing the test.

1.2. Problem statement

How can students assess their learning capabilities and have intelligent insights on their weak points to improve their knowledge?

1.3. Objectives

The main objectives of project are:

1. To give students a true learning experience through adaptive testing.
2. A detailed analysis on students' weak areas that need attention.
3. Provide intelligent insights and resources about how the students can improve their weak areas.

1.4. Main contributions

The students will build trust in our system as it will 100% provide accurate insights on their current knowledge about a concept. The students will be challenged to take the test again and again, and then see the difference in their preparation. Feedback plays an important role in effective learning. If we are given constant feedback about

our progress and what we need to improve, it pushes us and motivates us to perform better. The learning experience we provide to the students will be self-directed. The students will be able to set their own goals and will be self accountable.

1.5. Report organisation

Chapter 2 discusses the literature review from various research papers/journals. The key concepts being implemented are discussed, the previous studies on the computerised adaptive tests(CAT), item response theory(IRT) and latent traits are mentioned. We have also highlighted the main problem with the typical test preparation systems. The key aspects of the CAT simulation package, **catsim**, are also discussed.

Chapter 3 focuses on the key functionality. The high level use case diagram shows the overall system functionality and the functional requirements are discussed as use case descriptions. The non-functional requirements depict the constraints on the system and the quality attributes.

Chapter 4 highlights the system design and different design approaches to implement the frontend and backend of the system. The architecture of the system is discussed, various design constraints applying limitations and capturing the dynamic view of the system by sequence and activity diagrams.

Chapter 5 discusses the implementation details of the system and how the key process and workflows are implemented to achieve the key functionality.

Chapter 6 consists of the system testing and evaluation of our system performed in order to determine if the various features of our system conform to the user requirements and satisfy all the guidelines and standards.

Chapter 7 concludes the thesis by summarising different aspects of the work.

Chapter 2

Background Study/Literature Review

2.1. Latent Traits of Students and their Assessment

There are many educational and psychological tests which measure the latent traits. Latent traits are the ones which are not directly measurable and therefore “unobservable”. So, therefore in addition to the student’s performance there are unobservable traits that underlie a test. [1] The tests are designed in such a way that the answers to those questions allow the measurement of his/her underlying traits. When these traits are calculated through the defined models or using some mathematical functions they are called "abilities".

The drawbacks of the linear/typical tests are that students with different abilities have to go through the items with the same difficulty level in order to get their abilities measured. This means a student with low ability will struggle or lose interest in solving difficult questions whereas a student with high ability won't be challenged or get bored solving easy questions. These types of tests judge all the students on the same level and these cannot be used if a student wants to take a test to assess and improve his/her ability.

So, the tailored/adaptive tests define certain latent traits of the students, estimate scores on these traits and then predict and explain the student's performance. [1] Our system is used to assess the students current ability and help him improve his/her weak areas in order to ace the exam. Therefore, it's very important to have an accurate measurement of his current knowledge level by ensuring that the performance of the student in the test is according to the ability predicted by the system.

The students prepare better if they are given special attention. Our system will be like a personal tutor which helps in preparation by giving more attention to their weak areas/concepts of that subject.

2.2. Computerised Adaptive Testing (CAT)

The difficulty of the questions may differ from person to person and hence can never be certain[3]. In systems providing a sequential set of questions, the marks of questions are allotted without considering the difficulty level. The name of previously known tutoring systems[2] was changed to computerised adaptive testing(CAT) due to the technology advancements and the availability of the apps providing assessments.

The computerised adaptive testing(CAT) depends on the responses given by the users. The user's ability is estimated based on these responses and is used to select the next item, which is closer to the student's real ability.

Hence, the students with high ability won't have to answer too many questions which they think are easy for them, the system will automatically choose hard questions for them.

To practically implement CAT for students we need mathematical calculations to predict the student's ability accurately. For that purpose, item response theory was made with a goal of measuring latent traits.

2.3. Mathematical Formalisation by Item Response Theory (IRT) Models

Today the major educational tests including Graduate Record Examination(GRE) and Scholastic Aptitude Test (SAT) uses IRT to improve the accuracy in assessment of student abilities in much less time[4].

As Item response theory is a psychometric theory therefore it's not just limited to the field of education and testing students' aptitudes and abilities, it is also applicable in psychiatry, behavioural medicine, psychological tests and statistical calculations. So, in short IRT is used to observe the relationship between the latent traits and their observations or responses. IRT can also help to validate and improve the measurement scales.

Item Response Theory(IRT) consisting of one-, two-, three and four-parameter logistic models was created for adaptive testing in examination systems. IRT models judge and compare the student's current abilities and then allocates a certain ability level to each student. When the student takes the test again, the performance of the previous tests are stored in a record and is used for keeping the track of his performance and as an input to recalculate his ability.

In the IRT logistic models the students' ability is denoted as θ . As the students' ability cannot be measured directly therefore estimates need to be made, known as estimated ability.

Following four parameters are considered in these models[5]:

- a is known as item's *discrimination* parameter, how well it discriminates the students with different abilities.
- b is known as an item's *difficulty* parameter, how difficult it is to achieve 0.5 probability of a correct response under current students' ability.
- c is known as a pseudo-guessing parameter, probability of the students with low ability to still answer items correctly.
- d is known as *upper-asymptote*, probability of the students with high ability to still answer items incorrectly.

There are various mathematical formulas to calculate the different aspects of student abilities, these are discussed in the application of the **catsim** package and the details are in Chapter 4 of implementation.

Chapter 3

System Requirements

3.1. Use Case Diagram

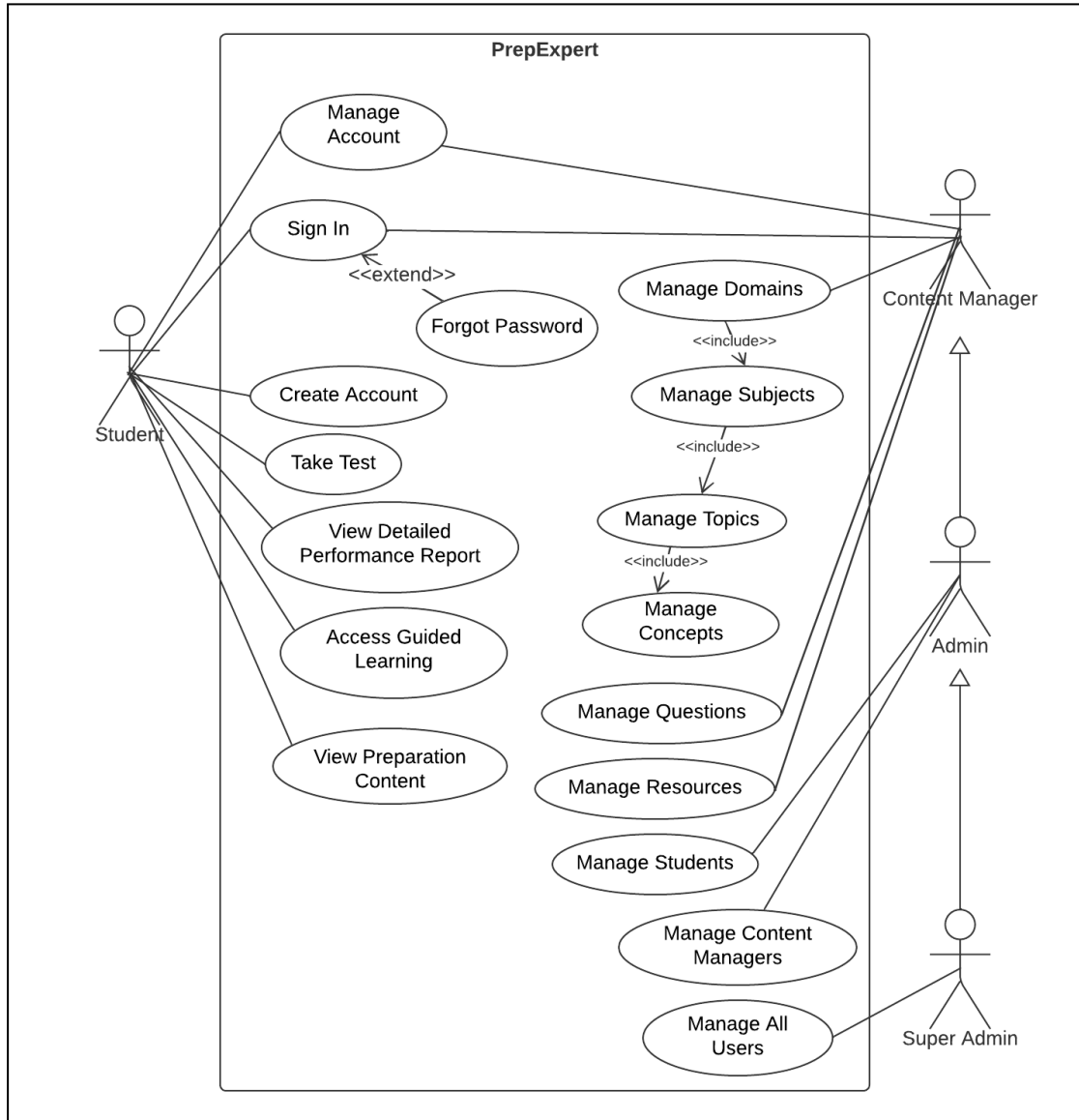


Figure 3.1: System Use Case Diagram

3.2. Functional Requirements

3.2.1. Authentication

3.2.1.1. *Create Account*

Included Use Cases	-
Primary Actor(s)	Student
Description	The user can create an account on the system.
Precondition	The user is not registered.
Trigger	The user clicks the “Create Account” button on the website.
Main Success Scenario	<ol style="list-style-type: none">1. The user opens the registration page.2. The user enters the required information and clicks the “Submit” button.3. System validates the user information.
Post Condition	<ol style="list-style-type: none">1. If the user information is valid, the system creates an account on the system and redirects the user to the “Login” page.2. If the user details are invalid, the system shows an error message against the fields.

3.2.1.2. *Sign In*

Included Use Cases	-
Primary Actor(s)	Student, Admin, Content Manager
Description	The user can sign in to the system.
Precondition	The user must have a valid account on the system.
Trigger	The user clicks the “Sign In” button.
Main Success Scenario	<ol style="list-style-type: none">1. The user opens the sign in page.2. The user enters their sign in credentials.3. System verifies the credentials from the database.
Post Condition	<ol style="list-style-type: none">1. If the entered credentials are correct, the user is signed in to the dashboard respective to their role type.2. If the user credentials are incorrect, the system shows an error message.

3.2.1.3. *Forgot Password*

Included Use Cases	-
Primary Actor(s)	Student, Admin, Content Manager

Description	The user can reset their account password.
Precondition	The user must have a valid account on the system.
Trigger	The user clicks the “Reset Password” button on the password recovery page.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user navigates to the password recovery page by clicking “Forgot Password” on the sign in page. 2. The user enters their email address. 3. System matches the user’s email address from the database.
Post Condition	<ol style="list-style-type: none"> 1. If the entered email address is correct, the system sends a password recovery link to the user’s email address. 2. If the entered email is not found in the database, the system shows an error message.

3.2.2. Profile Management

3.2.2.1. *View Account Information*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Student, Admin, Content Manager
Description	The user can view their account information.
Precondition	The user must be signed in to the system.
Trigger	The user clicks on the “Profile” icon in the navigation menu.
Main Success Scenario	The user navigates to the dashboard and clicks on the “Profile” icon. The user is able to see the account information.
Post Condition	The user is navigated to the Account Information page where they can see all their account details.

3.2.2.2. *Update Account Information*

Included Use Cases	Use Case 2.2.2.1 (View Account Information)
Primary Actor(s)	Student, Admin, Content Manager
Description	The user can update their account information.
Precondition	The user must be signed in to the system.
Trigger	The user clicks on the “Update Profile” button on the Account Information page.

Main Success Scenario	<ol style="list-style-type: none"> 1. The user is navigated to the page with editable account information fields. 2. The user updates the name and clicks the “Update” button.
Post Condition	<p>If the entered information is valid, the system updates the user information to the database.</p> <p>If the entered information is invalid, the system shows an error message.</p>

3.2.3. Test

3.2.3.1. *Take Test*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Student
Description	The user can take a test.
Precondition	The user must be signed in.
Trigger	The user clicks on the “Start” button under test information.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user navigates to the Take Test page. 2. System shows a list of all the tests (subject-wise) that are available to the user. 3. The user clicks the “Start Test” button to start the test which navigates the user to the Test screen. 4. On the test screen, the system shows test information, progress and time remaining to the user and starts the test. 5. System generates each question using the CAT algorithm on the user's test answers and previous user proficiency in the subject (if any). 6. Each question is offered on the basis of the user's performance, i.e. if the user performs poorly then he's offered maximum questions of “Easy” difficulty level. 7. Users cannot skip any questions but can discard or end the test. 8. On completing the test, the system saves the test progress to the database and shows the Test Result to the user.
Post Condition	-

3.2.3.2. *View Detailed Performance Report*

Included Use Cases	Use Case 2.2.3.1 (Take Test)
Primary Actor(s)	Student, Super Admin
Description	The user can view detailed performance reports.
Precondition	The user must be signed in.

Trigger	The user clicks on the “Reports”.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user completes a test and is navigated by the system to “Reports” page. 2. The user selects the test. 3. Detailed Performance Report page shows: <ol style="list-style-type: none"> a. User’s test score, completion time, subject percentile. b. User’s performance topic-wise and concept-wise along with a comparison chart with the peers (other users performing the same test). c. Option to review the test session and see all the correct and incorrect attempts along with their correct answers. d. Option to view preparation content in order to improve their subject performance.
Alternate Scenario	<ol style="list-style-type: none"> 1. The user completes a test and navigates to the “View Reports” page from the navigation menu. 2. On the View Reports screen, the user clicks View Detailed Performance against any test to see the Detailed Performance Report. 3. Detailed Performance Report page shows: <ol style="list-style-type: none"> a. User’s test score, completion time, subject percentile. b. User’s performance topic-wise and concept-wise along with a comparison chart with the peers (other users performing the same test). c. Option to review the test session and see all the correct and incorrect attempts along with their correct answers. d. Option to view preparation content in order to improve their subject performance.
Post Condition	-

3.2.3.3. Access Guided Learning

Included Use Cases	-
Primary Actor(s)	Student
Description	The user can access the guided learning highlighting weak areas that need improvement.
Precondition	The user must be signed in.
Trigger	The user clicks on the “Guided Learning” in the detailed performance report or from the sidebar menu by selecting the subject.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user completes a test and views a Detailed Performance Report. 2. From the Detailed Performance Report, the user clicks the “Improve Score” button that takes the user to the Guided Learning page.

	3. Guided Learning page highlights all the weak topics and concepts of the related subject and shows Resources to improve the test score.
Alternate Scenario	<ol style="list-style-type: none"> 1. The user navigates to the Guided Learning page from the navigation menu on the sidebar. 2. The Guided Learning page lists all the subjects that need improvement. 3. On clicking “View Improvements” against any subject on the Guided Learning page, the system highlights all the weak topics and concepts of the related subject and shows Resources to improve the test score.
Post Condition	-

3.2.3.4. View All Preparation Content

Included Use Cases	-
Primary Actor(s)	Student
Description	The user can view all the preparation content and resources.
Precondition	The user must be signed in.
Trigger	.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user navigates to the Resources page from the navigation menu. 2. Resources page lists down all the subjects along with the options to view resources. 3. On clicking the “View Resources” button against any subject, the system shows Resources against all the subject topics and the included concepts. 4. Concepts have multiple resources, shown in the form of embed content, such as videos from YouTube, Vimeo, etc.
Post Condition	-

3.2.4. Administration

3.2.4.1. Manage Admins

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin
Description	Super admin can manage admin accounts and information.
Precondition	The user must be signed in as Super Admin.

Trigger	The user clicks on the Manage Admins from the menu.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the admin dashboard and clicks on Manage Admins from the menu. 2. System shows a list of admins that are registered. 3. The user clicks the “Add” button, fills in the admin details to add a new admin to the system. 4. The user clicks the “Edit” or “Delete” button against an entry to modify or remove that admin.
Post Condition	System updates the admin information to the database.

3.2.4.2. *Manage Students and Content Managers*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin
Description	Super Admins & Admins can manage students and content manager’s accounts and information.
Precondition	The user must be signed with an admin/super admin account.
Trigger	The user clicks on the Manage Users from the menu.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the admin dashboard and clicks on Manage Users from the menu. 2. System shows a list of users that are registered. 3. The user clicks the “Add” button, fills in user details to add a new user to the system. 4. The user clicks the “Edit” or “Delete” button against an entry to modify or remove that admin.
Post Condition	System updates the user information to the database.

3.2.5. Test Management

3.2.5.1. *Manage Domains*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage the domain from where the test is taken from.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the Manage Domains from the menu.

Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on Manage Domains under the Test Content in the menu. 2. System shows a list of Domains that are included in the test. 3. The user adds a new Domain by clicking the “Add” button on top of the list. 4. The user modifies the existing domains by clicking the “Edit” or “Delete” button against the domain entries.
Post Condition	System updates the domain’s information in the database. In case of deleting a domain, the related subjects, topics and concepts are also deleted through object relational mapping.

3.2.5.2. *Manage Subjects*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage subjects in a domain.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the “View Subjects” against each domain in the domain table.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on Domains and then “View Subjects”. 2. System shows a list of Subjects that are included in the domain. 3. The user adds a new Subject by clicking the “Add” button on top of the list and specifies the subject information along with the domain that it belongs to. 4. The user modifies the existing subjects by clicking the “Edit” or “Delete” button against the subject entries.
Post Condition	System updates the subject’s information in the database. In case of deleting a subject, the related topic and concepts are also deleted through object relational mapping.

3.2.5.3. *Manage Topics*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage topics in a subject.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the “View Topics” against each subject in the Subjects table.

Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and then “View Topics” on the Subjects table. 2. System shows a list of Topics that are included in that Subject. 3. The user adds a new Topic by clicking the “Add” button on top of the list and specifies the topic information along with the subject that it belongs to. 4. The user modifies the existing topics by clicking the “Edit” or “Delete” button against the topic entries.
Post Condition	System updates the topic’s information in the database. In case of deleting a topic, the related concept and questions are also deleted through object relational mapping.

3.2.5.4. *Manage Questions*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage questions in a concept.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the View Questions option shown on the concepts page.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “View Questions” option in the concepts. 2. System shows a list of Questions that are included in the concept. 3. The user adds a new Question by clicking the “Add” button on top of the list and specifies the question details. 4. The user modifies the existing questions by clicking the “Edit” or “Delete” button against the question entries.
Post Condition	System updates the questions in the database.

3.2.5.5. *Manage Question Bank*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage a question bank.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the Question Bank from the sidebar menu.

Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “Question Bank” option in the sidebar menu. 2. System shows a list of all the questions in all the concepts. 3. The user adds a new Question by clicking the “Add” button on top of the list and specifies the question details along with the concept it belongs to. 4. The user modifies the existing questions by clicking the “Edit” or “Delete” button against the question entries.
Post Condition	System updates the questions in the database.

3.2.5.6. *Manage Linked Video Resources*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage resources in a concept.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the “View Resources” option from the concepts page.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “View Resources” option in the concepts. 2. System shows a list of video resources that are linked to that concept. 3. The user adds a new video resource by clicking the “Add” button on top of the list and specifies the video resource details. 4. The user modifies the existing video resource by clicking the “Edit” or “Delete” button against the video resource entries.
Post Condition	System updates the video resource in the database.

3.2.5.7. *Manage Video Resources*

Included Use Cases	Use Case 3.2.1.2 (Sign In)
Primary Actor(s)	Super Admin, Admin, Content Manager
Description	The user can manage all video resources.
Precondition	The user must be signed as a Super Admin/Admin or Content Manager.
Trigger	The user clicks on the “Resources” option from the sidebar menu.
Main Success Scenario	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “Resources” option in the sidebar menu.

	<ol style="list-style-type: none"> 2. System shows a list of video resources that are available in general. 3. The user adds a new video resource by clicking the “Add” button on top of the list and specifies the video resource details. 4. The user modifies the existing video resource by clicking the “Edit” or “Delete” button against the video resource entries.
Post Condition	System updates the video resource in the database.

3.3. Interface Requirements

3.3.1. User Interface:

The system consists of three main dashboards described below with their key features:

1. Student Dashboard
 - a. View Available Tests
 - b. Take Test
 - c. View Detailed Report
 - d. View Guided Learning
 - e. View Resources
 - f. Intelligent insights and analytics
2. Content Manager Dashboard
 - a. Manage Test Content (Domains, Subjects, Topics and Concepts)
 - b. Manage Resources
3. Admin Dashboard
 - a. Manage Students and Content Managers
 - b. Manage Test Content (Domains, Subjects, Topics and Concepts)
 - c. Manage Resources
 - d. View All Students’ Test Records
4. Super Admin Dashboard
 - a. Manage All Users(Admins, Content Managers, Students)
 - b. All features available to content managers and admins

3.3.2. Physical Interface

There is no hardware or physical interface needed for our system.

3.4. Database Requirements

The system must use MariaDB Server based on MySQL 8 as its database component. Communication with the DB is through MySQLi connection.

3.5. Non-Functional Requirements

3.5.1. Performance Requirements

3.5.1.1. Capacity

In capacity we discuss the minimum number of objects that the system can support:

- The system shall support a minimum of 100 concurrent tests.
- The system shall support a minimum of 500 active users at a time.
- The system shall support a minimum of 100 simultaneous interactions.

3.5.1.2. Latency

Latency is the maximum time that is permitted for the system to execute specific tasks (i.e., system operations).

- The system must start a test within 30 seconds of the user request
- The performance reports must be generated within 60 seconds of the test completion.

3.5.1.3. Response Time

In response time, we discuss the maximum time that is permitted for the system to respond to requests.

- All system responses shall occur within 30 seconds.

3.5.2. Safety Requirements

- The system interface must be designed in compliance with the Web Design Guidelines to make it friendly for colour blind users.

- The colour palettes and interface controls must be highlighted for easy navigation.

3.5.3. Security Requirements

3.5.3.1. Data Transfer

- The system shall use secure sockets in all operations that include any confidential customer information.
- The system shall automatically log out all customers after a period of inactivity.
- The system will log out all customers automatically if they are inactive for a certain period of time.

3.5.3.2. Data Storage

- The system's back-end database shall be encrypted.
- The customer's web browser shall never display a customer's password. It shall always be echoed with special characters representing typed characters.

3.5.4. Software Quality Attributes

3.5.4.1. Availability

- The system shall be available for 99% of the time during peak traffic hours (day time).
- Any maintenance work must take place within an hour, specifically after midnight when user traffic is the lowest.

3.4.4.2. Reliability

- The system shall provide storage of all databases on redundant servers to automatically switchover when the main server shuts down.
- The system shall be able to restart itself if it crashes or shuts down unexpectedly.

3.4.4.3. Reusability

The system structure and classes must be designed in such a way that the code can be reused.

3.4.4.4. Portability

The system services must be platform independent so that the software can be developed for multiple platforms, such as Android, iOS, Web, etc.

3.4.4.5. Flexibility

The software structure must be flexible and the database should be normalised so that there is minimum dependency between modules and changes can be easily made in future.

3.6. Project Feasibility

3.6.1. Technical Feasibility

Given the technical complexity of the project and open source technologies available, it was certainly feasible for our team to build this project.

1. Modern dashboard with all the statistics and charts of user's performance is done using React JS which offers component based approach for making web pages and widgets that can be reused.
2. Redis Cache DB is used to store and load all the test data while taking a test. This helps in fast processing of CAT algorithms for adaptive test generation on the go.
3. FastAPI is used to retrieve and store all the persistent data to MySQL DB. FastAPI has a very low response time and is highly scalable - can be extended to other apps such as Android, iOS, etc.
4. Under the supervision of our project supervisor, our technical team consisting of two developers, Frontend and Backend, have been able to successfully develop and deploy all the documented requirements.
5. We utilised free resources available to us in GitHub Student Pack such as Amazon EC2, Netlify and Heroku to test and deploy our project. When deploying to Production, we will move to paid plans.

3.6.2. Operational Feasibility

PrepExpert has covered almost all the functionalities that were documented in the initial proposal and the SRS document.

1. Students can perform tests, view insights and detailed performance reports on their performance. Our system lets students rely on the analytics that we provide for their preparation for the exam.
2. Students get to know their weak performance areas that need attention and can access resources such as videos, etc. to improve their weak areas.
3. Content Managers can manage all the test content and the resources that are provided to the students keeping in mind that the test content is effective and covers each part of the test.
4. Administrators can manage students and content managers as well as see system statistics such as no. of users, no. of tests taken, etc. They can also view students' performance as well as manage the test content.

Therefore, the usability of the system is very much high as we have customised the dashboards for each type of user.

5. During a test, the student's proficiency in a subject is recalibrated using the CAT algorithm after each attempt. This helps in providing questions to students based on their current and previous performances.
6. System provides simple navigation menus, role based access and data validations on both the frontend and backend.

3.6.3. Legal and Ethical Feasibility

Our system conforms to all legal and ethical regulations.

- We do not take any sensitive information from the user. All the user information is kept confidential and the passwords are encrypted using BCrypt Password Hash.
- The test content is taken from the copyright-free books and online sources that are publicly available for all types of use.
- Reference links and copyrights are provided with all the preparation resources embedded in our system.
- We use open-source React JS and FastAPI packages for development of the system.
- Human Interface Guidelines have been followed to offer maximum usability and accessibility for the users.

3.7. Conclusion

The high level use case diagram shows the overall system functionality and the functional requirements are discussed as use case descriptions. The non-functional requirements depict the constraints on the system and the quality attributes.

Chapter 4

System Design

In this section, we will discuss the design and architecture of our system and how it has been implemented to develop the system.

4.1. Design Approach

Software design is a process to convert the software requirements into actual implementation.

4.1.1. Backend Implementation

- The backend system uses **Function Oriented Design Approach**.
- The system is divided into multiple subsystems (API endpoints) that act as independent services and handle requests from the application.
- These subsystems (API) are further divided into many atomic functions that perform simple query and processing tasks.

4.1.2. Frontend Implementation

- The frontend application uses **Structured Design Approach**.
- The application comprises many well-organised elements known as Components that represent Web Pages and Frontend functionality.
- These components may also consist of smaller components known as widgets that can be reused and improve modularity of the application.

4.2. Design Constraints

4.2.1. System Overload

The use of service oriented architecture allows for the conjunction of different services. All inputs are validated before they are sent to the service so the use of multiple services can cause validation overload on the system.

4.2.2. High Bandwidth Server

The web services in the software system send and receive information frequently which can result in a million requests per day. Therefore, a high bandwidth server is required to deploy the web services.

4.3. System Architecture

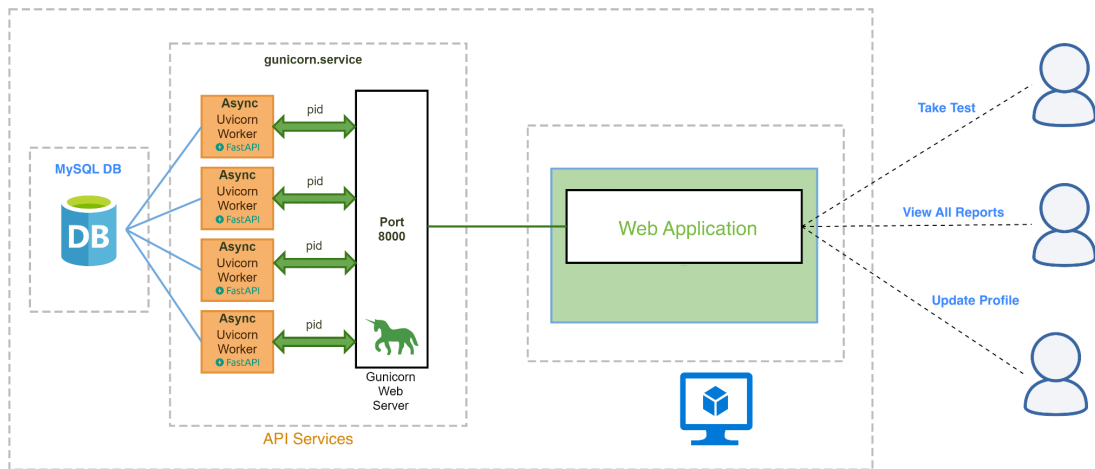


Figure 4.1: System Architecture

We are using a **Service Oriented Architecture (SOA)**. The system communication with the database takes place using small light-weight services (REST API) that are platform independent.

- SOA allows us to make multiple independent services that handle requests from the web application.
- FastAPI is a web framework written in Python that uses ASGI (Asynchronous Gateway Interface) and REST interface to process API requests at blazing fast speed. FastAPI along with SOA helps in creating API endpoints for these services.
- Service Oriented Architecture is highly scalable which means that our application can easily be developed for other platforms such as Android, iOS, Linux, etc.
- The component-based approach in ReactJS improves the overall performance by rendering each component/widget and its data from API as needed.

4.4. Logical Design

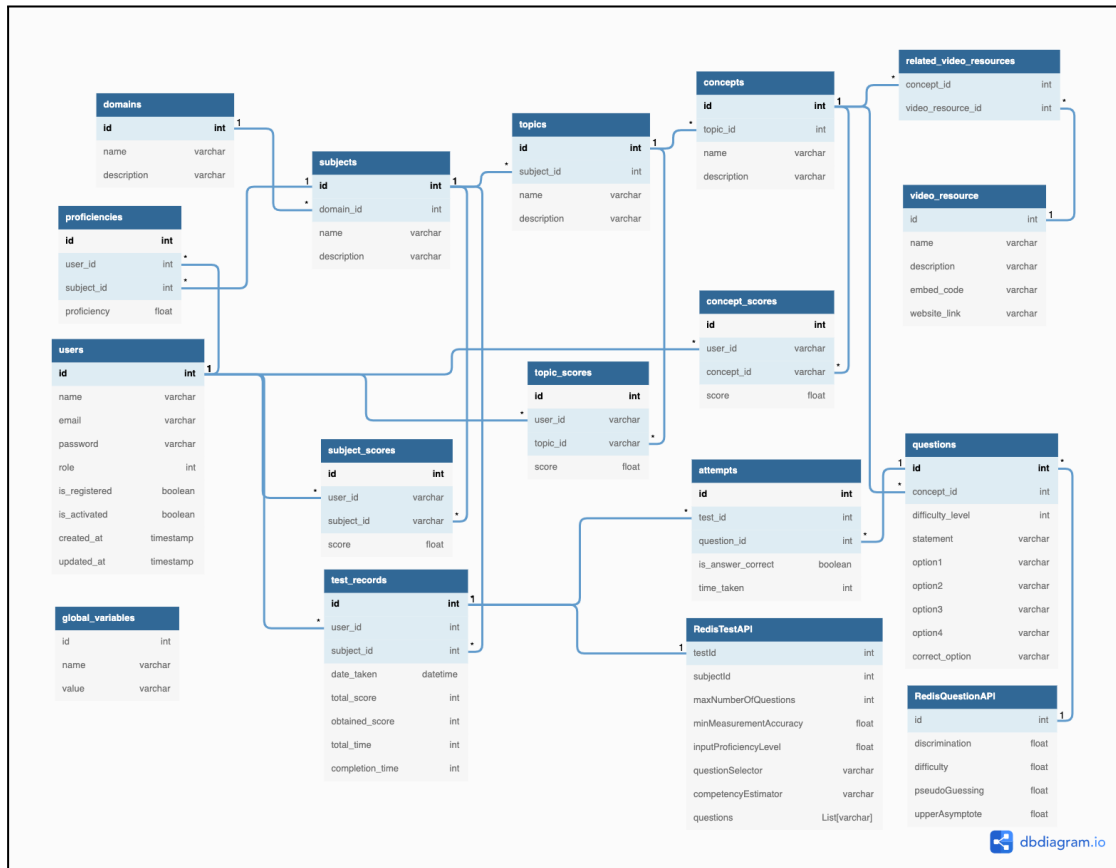
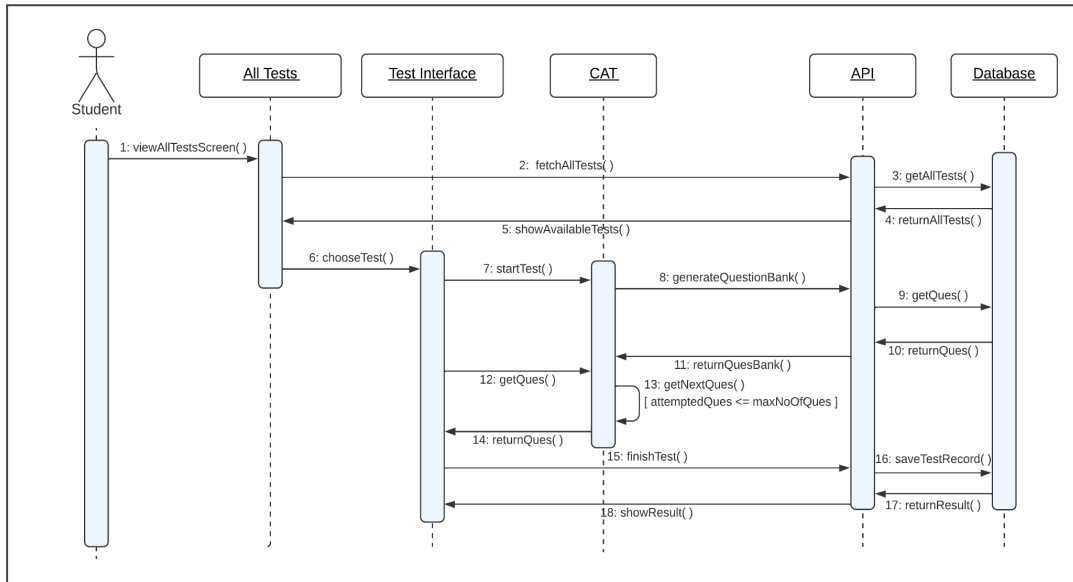


Figure 4.2: Logical Design

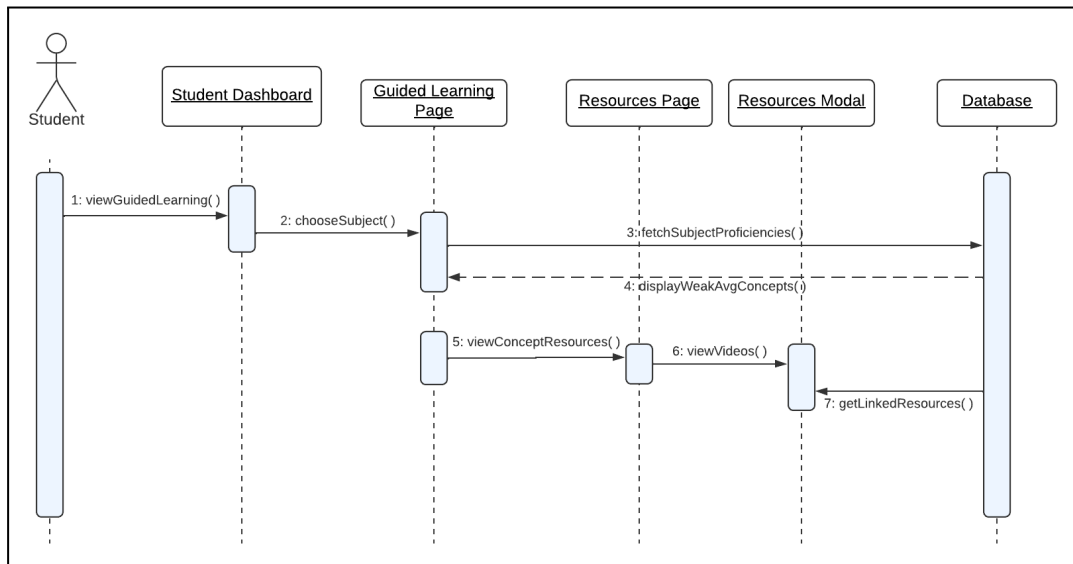
4.5. Dynamic View

4.5.1. Sequence Diagrams

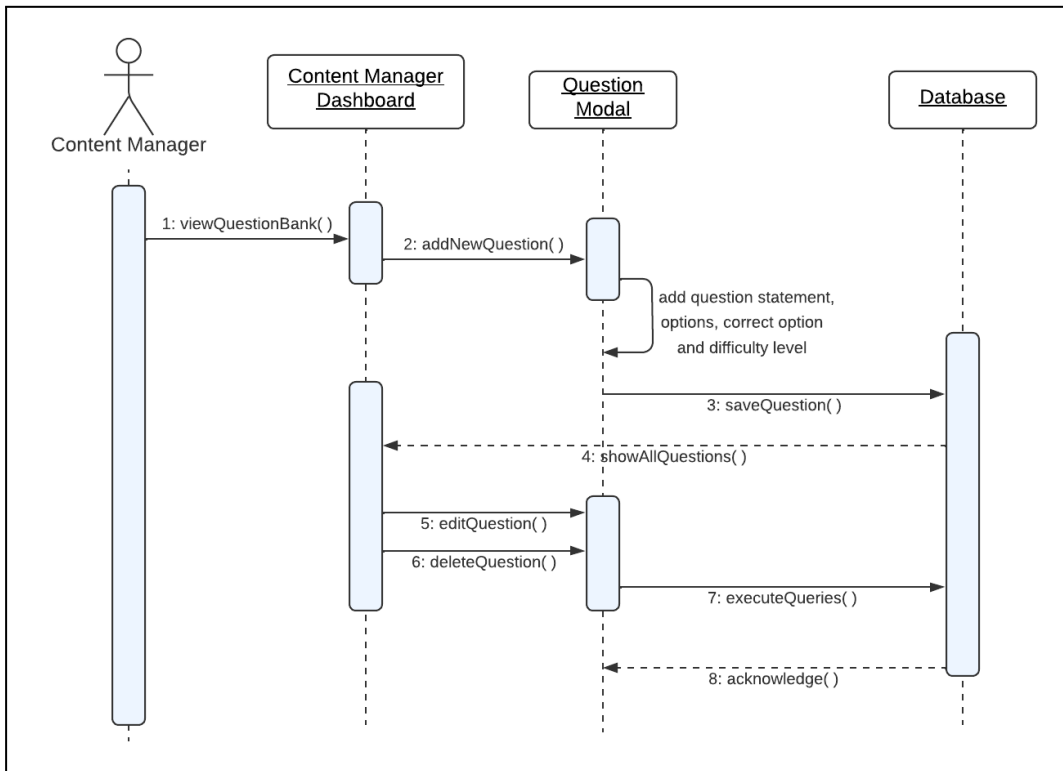
4.5.1.1. Take Test



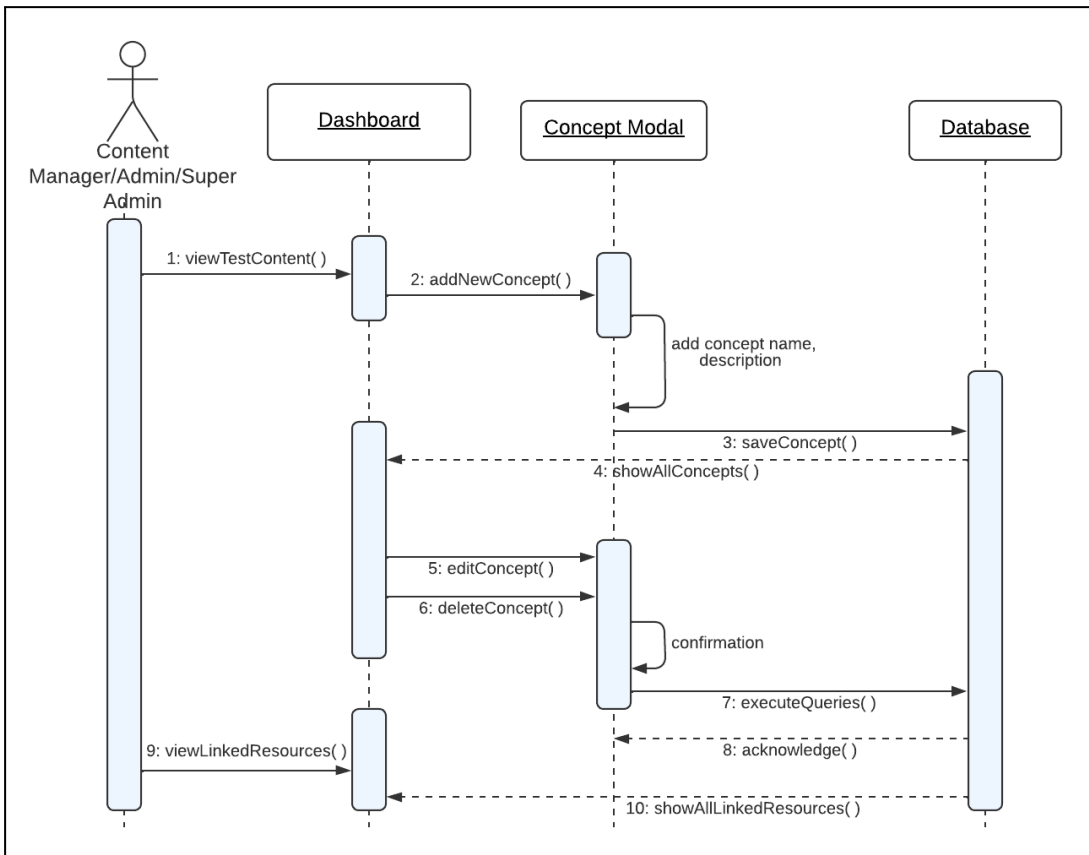
4.5.1.2. View Guided Learning & Resources



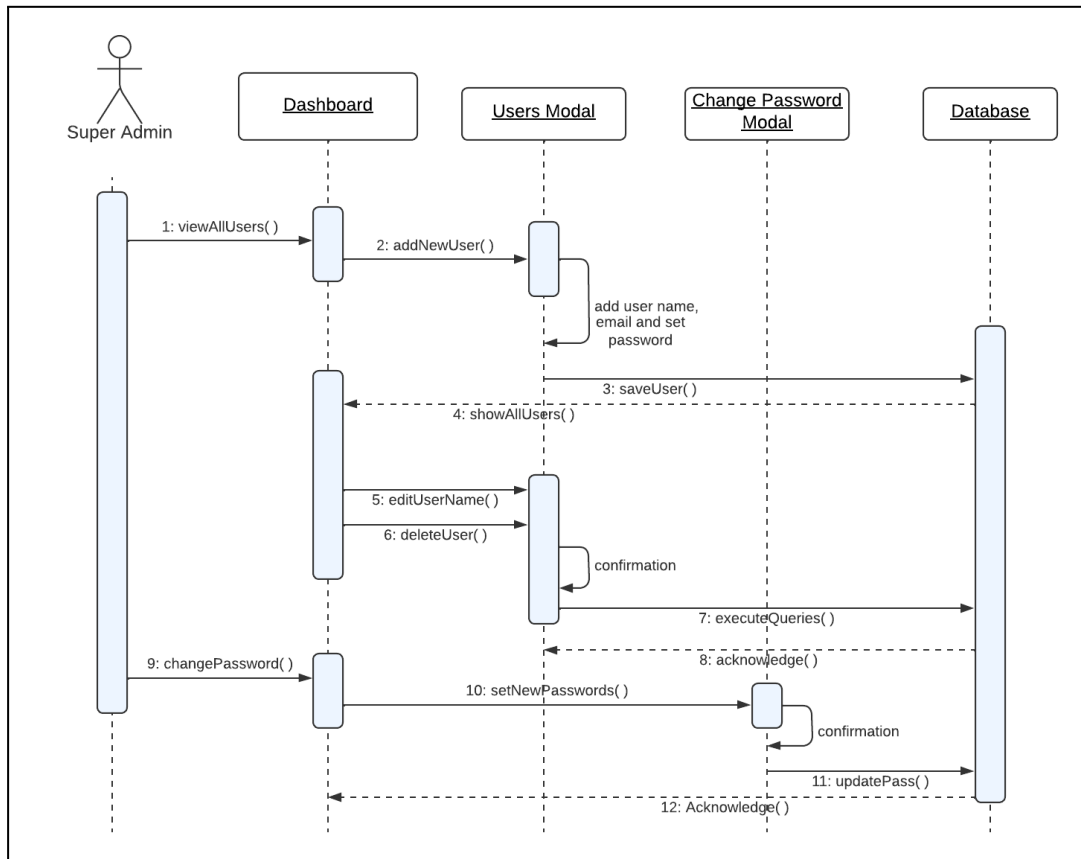
4.5.1.3. Manage Questions/Question Bank



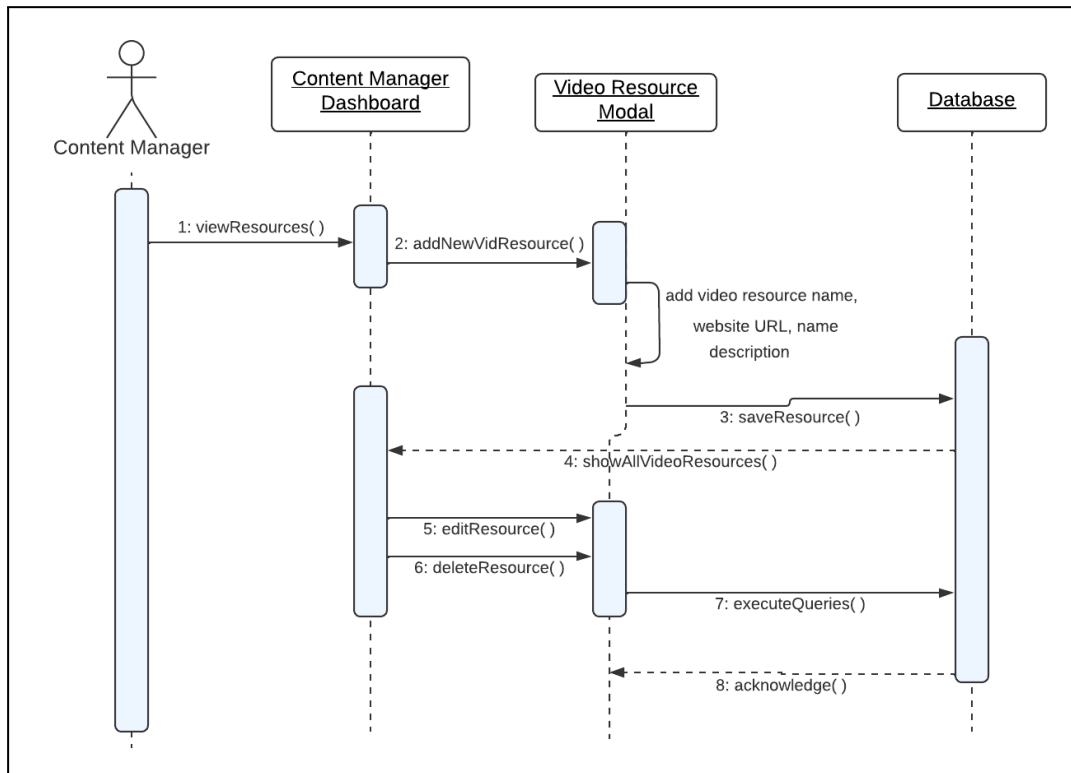
4.5.1.4. Manage Test Content



4.5.1.5. Manage Users

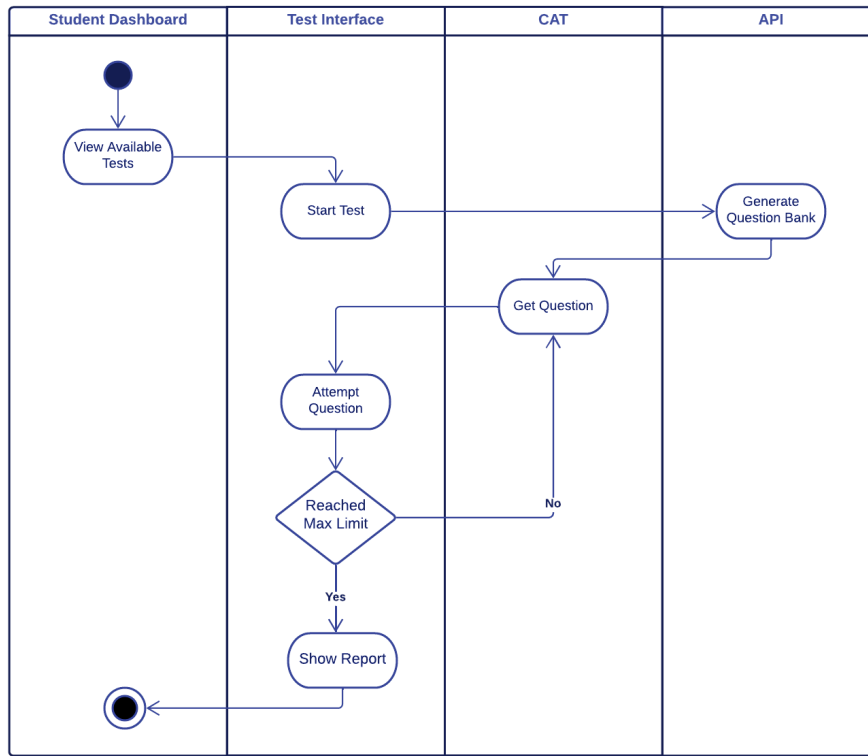


4.5.1.6. Manage Resources

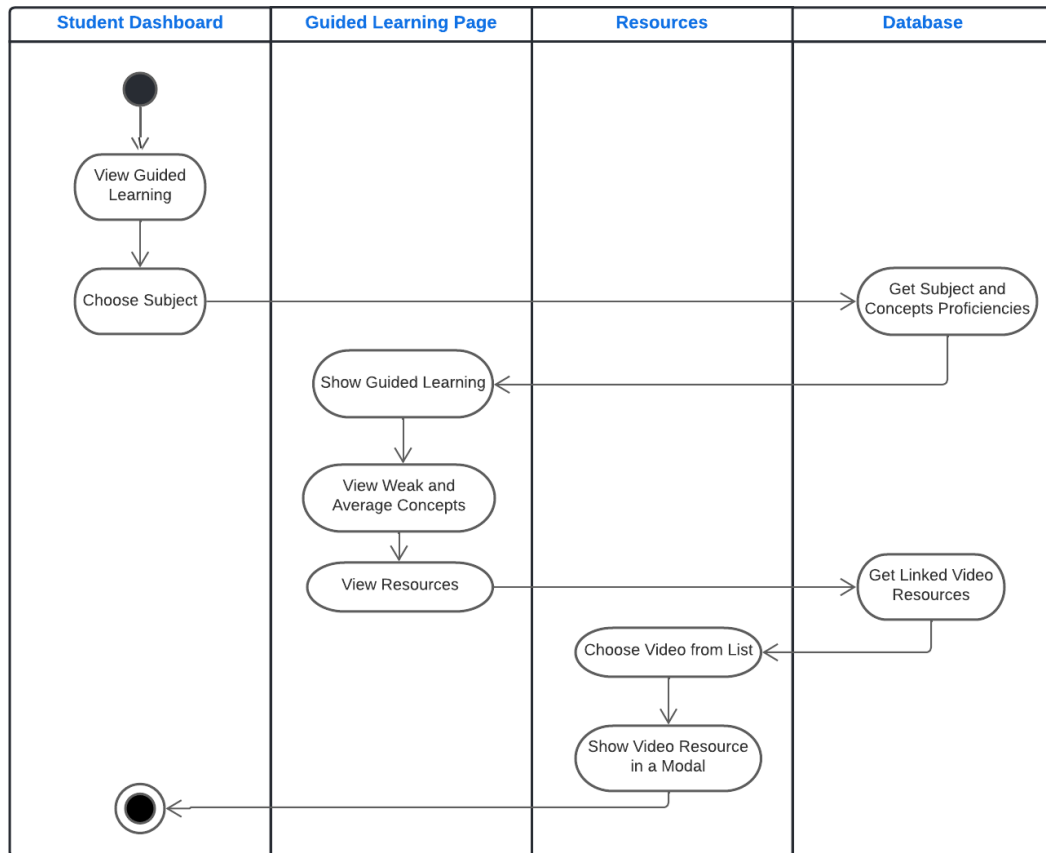


4.5.2. Activity Diagrams

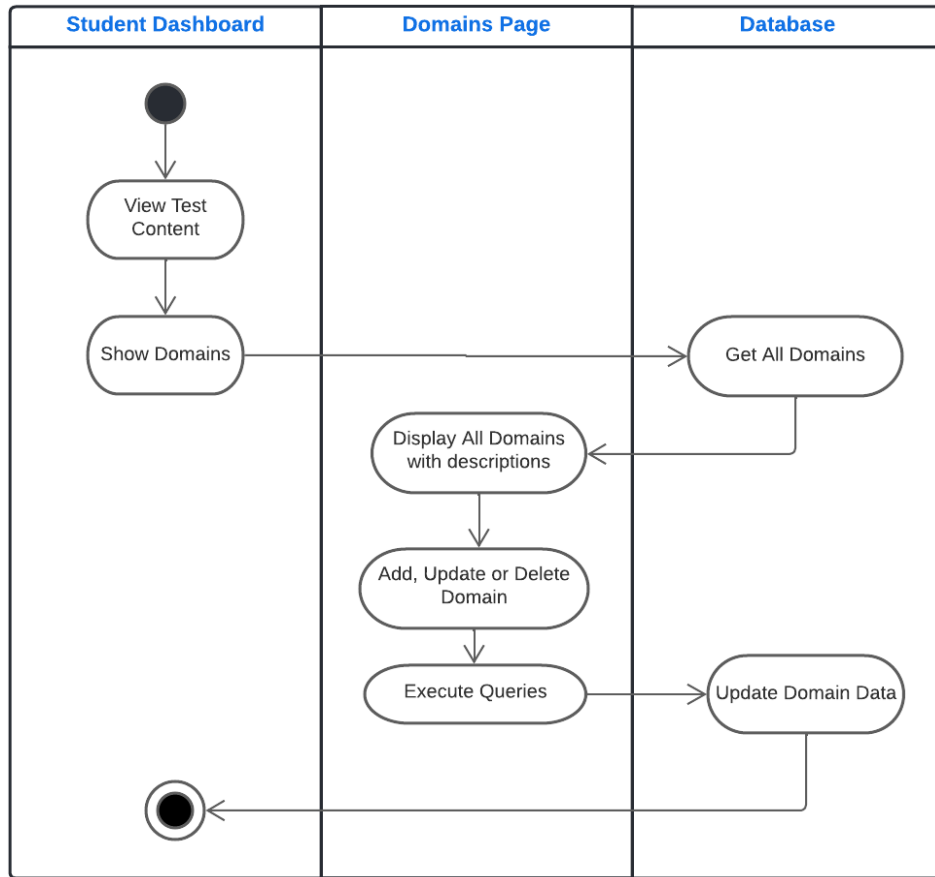
4.5.2.1. Take Test/View Detailed Report



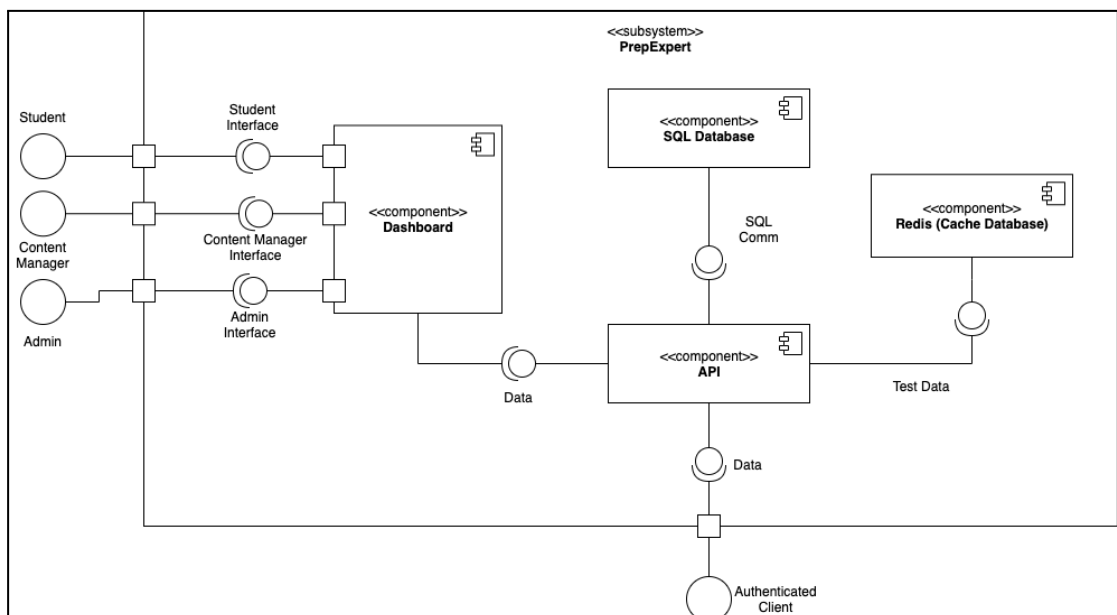
4.5.2.2. View Guided Learning



4.5.2.3. Manage Test Content



4.6. Component Design



4.6.1. Deployment View

- The user will use the latest Web Browser on their computer to access the software system which is hosted on a Web Server.
- The Web Browser hosts the webpages, stores and retrieves data from the Database Server which contains MariaDB database.
- On receiving user's requests, the Web Server serves the requests using a HTTP Secure (HTTPS) connection over default HTTPS Port 443.

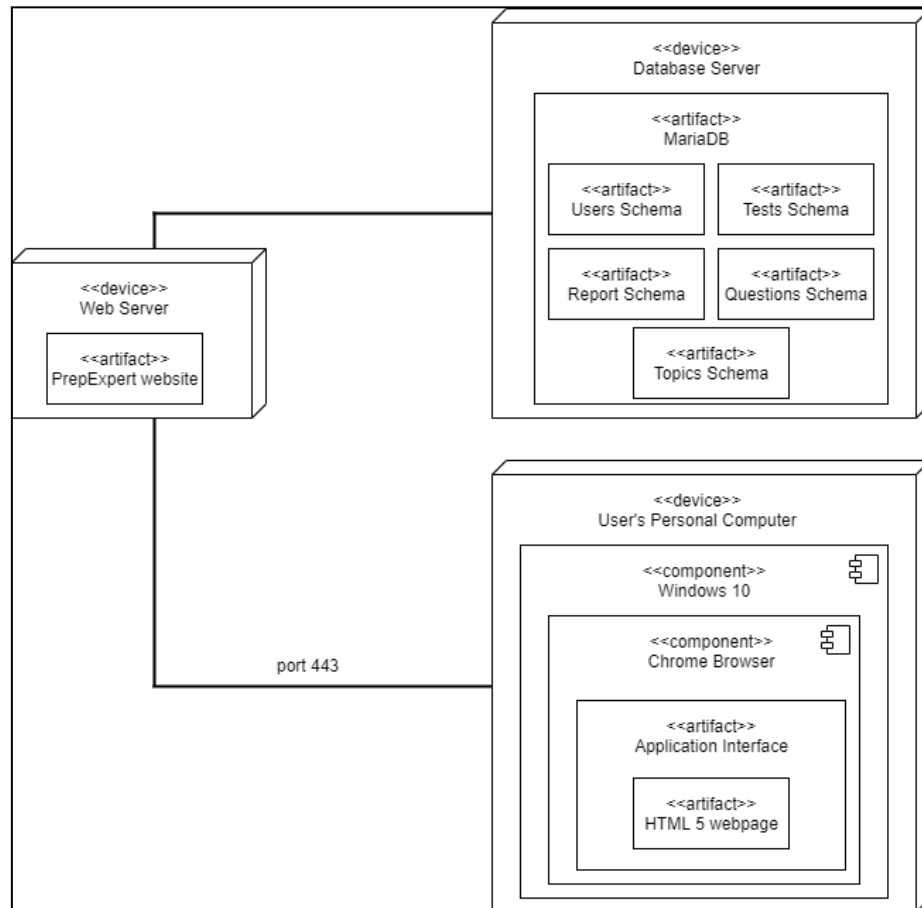


Figure 4.3: Deployment View

4.7. Data Models

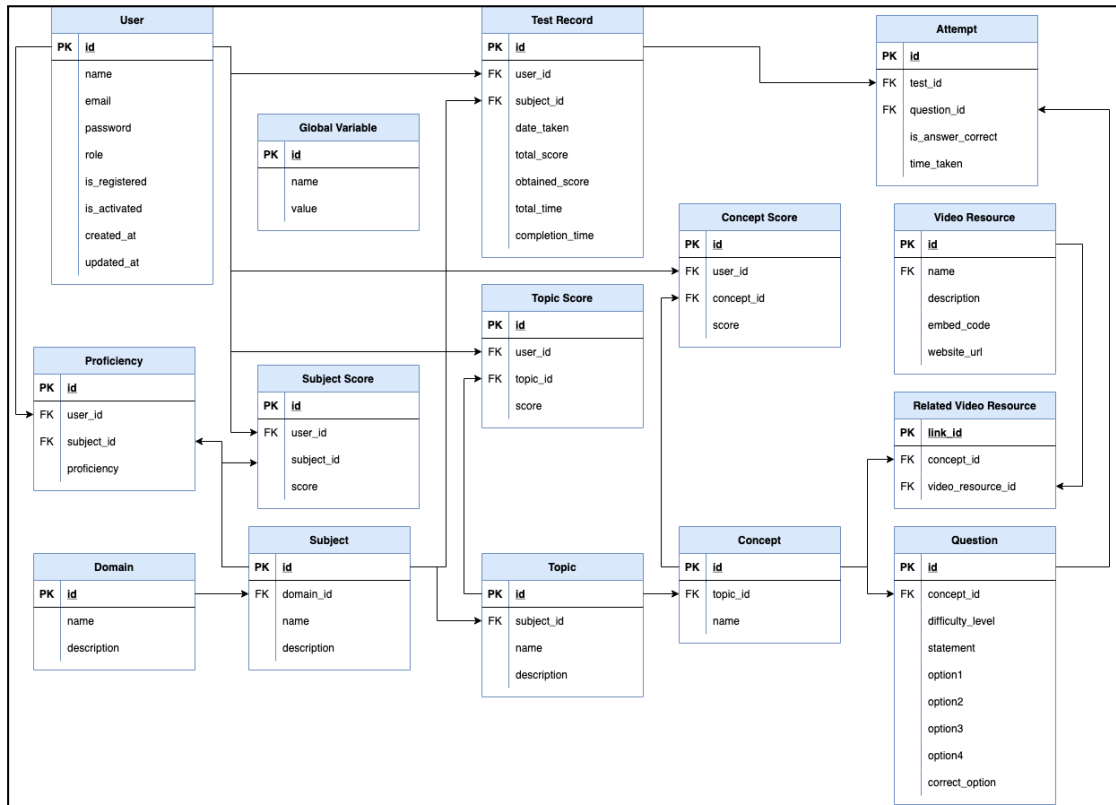
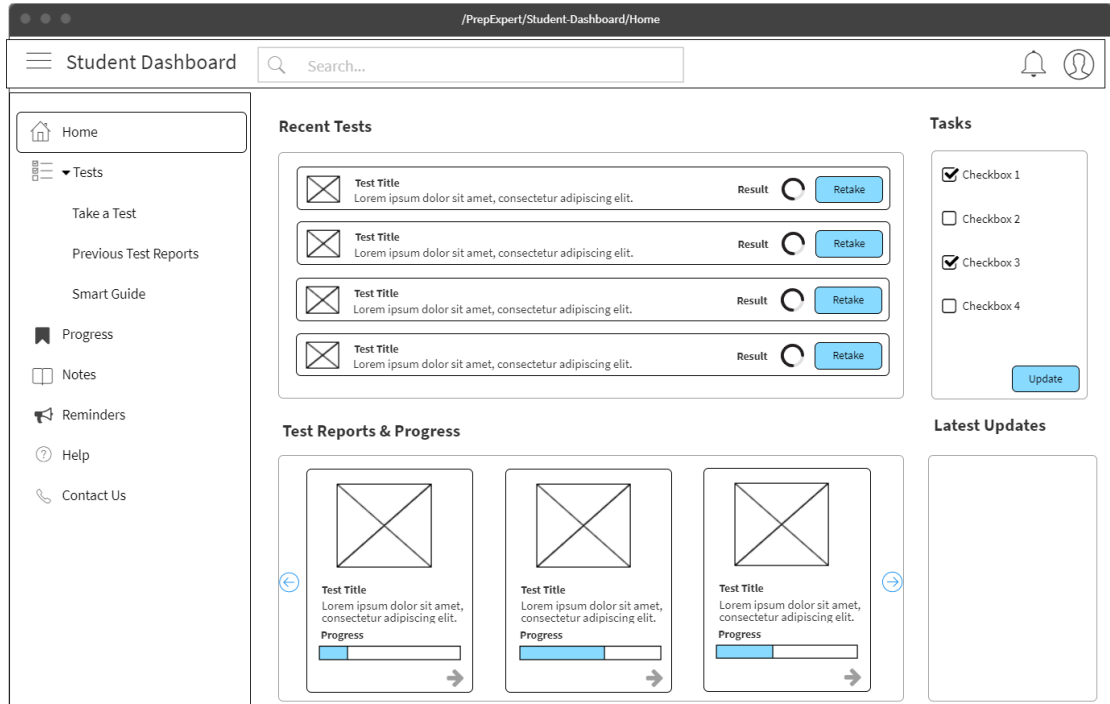


Figure 4.4: Data Models

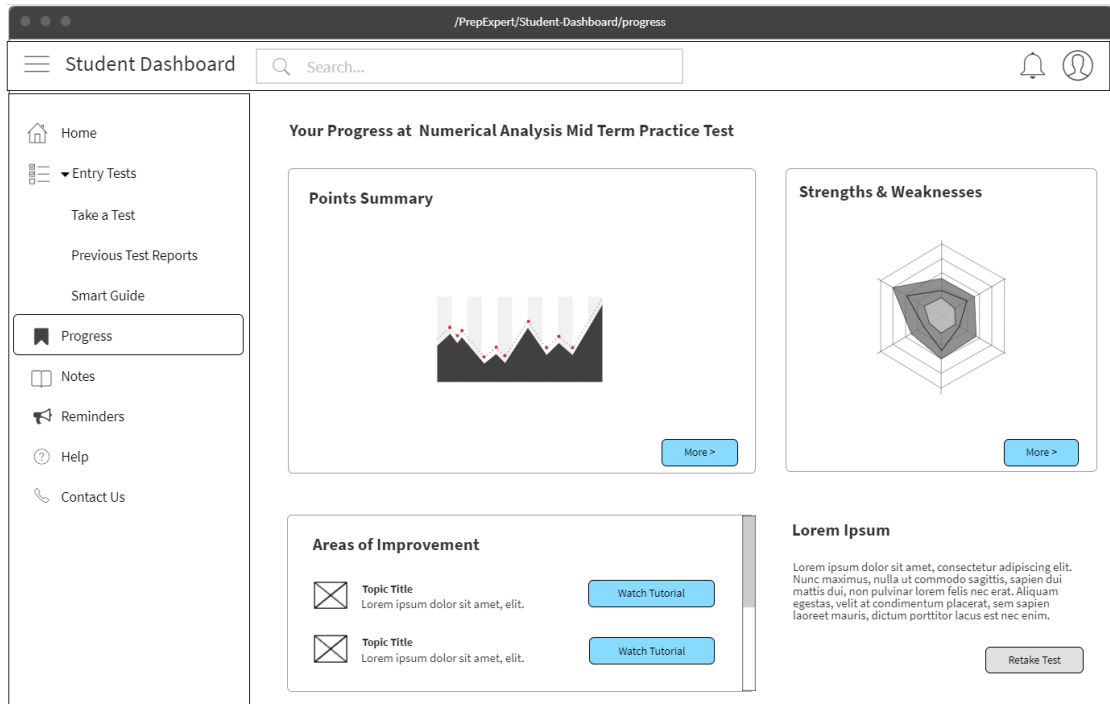
4.8. User Interface Design

4.8.1. Low Fidelity Prototypes:

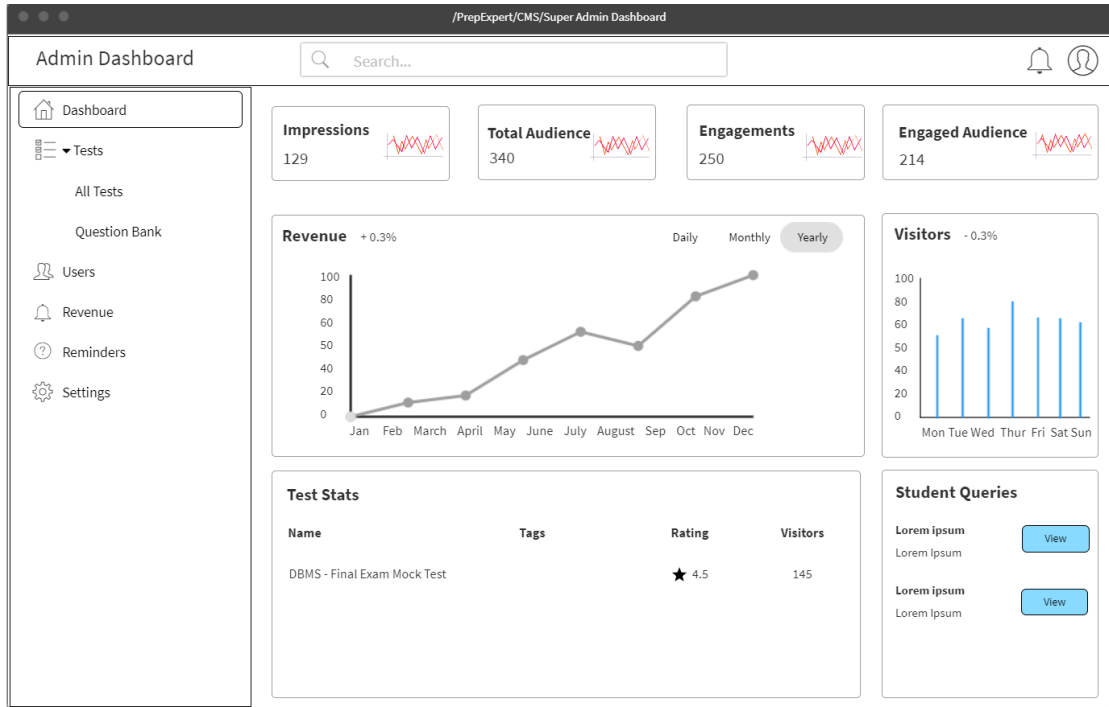
4.8.1.1. Student Dashboard - Homepage



4.8.1.2. Student Dashboard - Progress Page



4.8.1.3. Admin Dashboard - Homepage



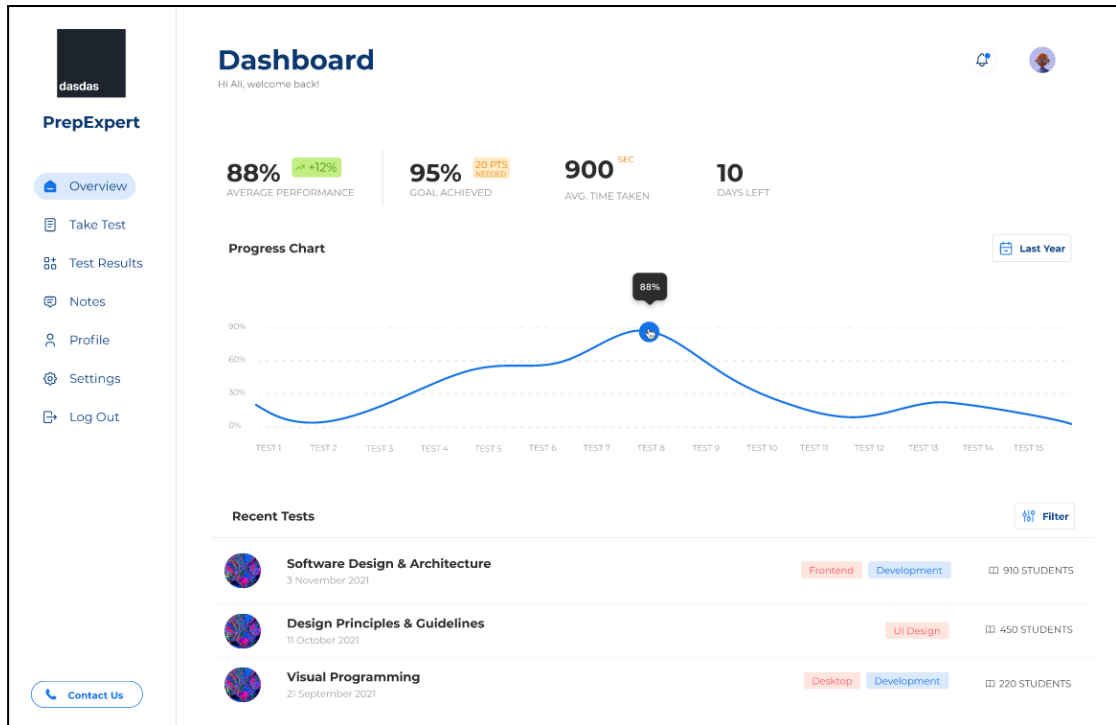
4.8.1.4. Content Manager - Question Bank

The Content Manager - Question Bank interface includes a sidebar with navigation links: Dashboard, Tests, All Tests, Question Bank, Users, Revenue, Reminders, and Settings. The main content area features a "Question Bank" section with filters for "DBMS" and "Final Mock Test", and an "Add New Question" button. Below is a table of questions:

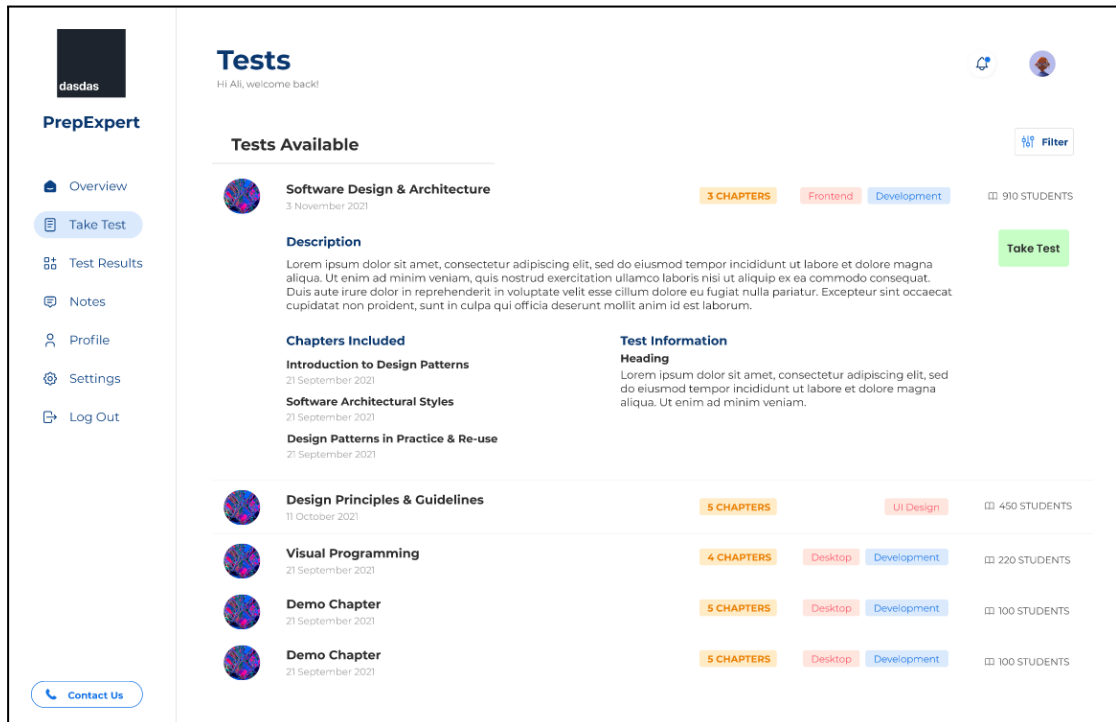
Sr No.	Topic	Question	Answer	Difficulty	Actions
1.	SQL Joins	Which join is equivalent to cartesian product?	Cross Join	Beginner	View Edit Delete
2.	SQL Joins	Which query retrieves rows from more than one table?	Join	Beginner	View Edit Delete
3.	SQL Joins	How many join types in join conditions?	5	Intermediate	View Edit Delete
4.	SQL Joins	Which is a join condition contains an equality operator?	equijoins	Advance	View Edit Delete
5.	Normalization	Functional dependency is a relationship between/among	Attributes	Advance	View Edit Delete
6.	Normalization	5NF is designed to cope with	Join dependency	Advance	View Edit Delete


4.8.2. High Fidelity Prototypes:

4.8.2.1. Student Dashboard - Homepage



4.8.2.2. Student Dashboard - Take Test





PrepExpert

- Overview
- Take Test
- Test Results
- Notes
- Profile
- Settings
- Log Out

[Contact Us](#)

Take Test

Test In Progress...

Software Design & Architecture
3 November 2021

30% Complete

Remaining Time
14:49 minutes

Question 11

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Specify the correct option

- Option A
- Option B
- Option C
- Option D

Question 12


Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Specify the correct option

- Option A
- Option B
- Option C
- Option D

End Test Next Page >

4.8.2.3. Content Manager - Add Questions



PrepExpert

- Overview
- Tests
- Add Questions
- Question Bank
- Notes
- Profile
- Settings
- Log Out

Add Questions

Question:

Add your question here

Beginner Intermediate Advance

Options:

			Correct Answer
1	<input type="text" value="Option 1"/>	🗑️	❌
2	<input type="text" value="Option 2"/>	🗑️	❌
3	<input type="text" value="Option 3"/>	🗑️	✅
4	<input type="text" value="Option 4"/>	🗑️	❌

[+ Add Option](#)

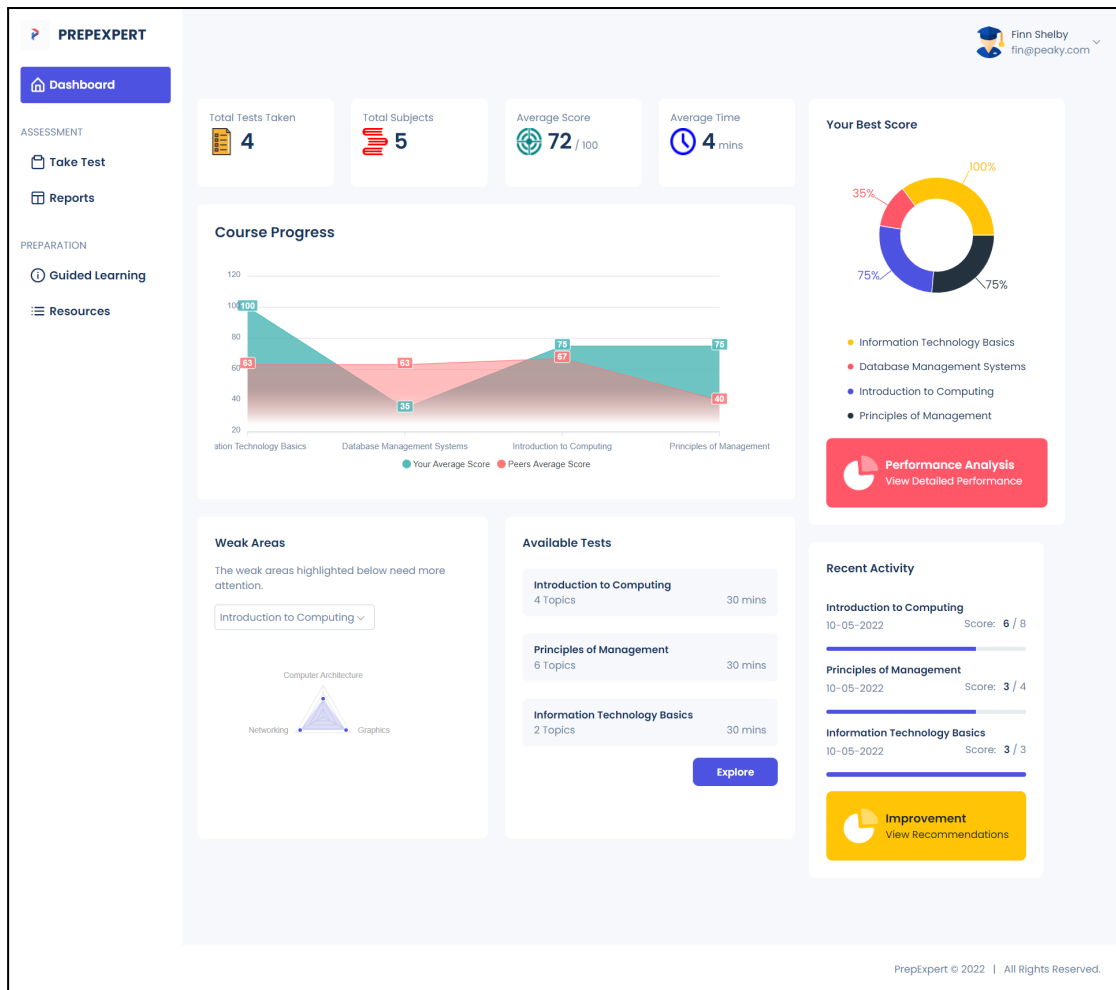
Mapped to:

Subject: Chapter: Topic:

Add Question Preview

4.8.3. User Interface:

4.8.3.1. Student Dashboard



4.8.3.2. Take Test

PREPEXPERT

Dashboard

ASSESSMENT

Take Test

Reports

PREPARATION

Guided Learning

Resources

Finn Shelby
fin@peaky.com

Available Tests

The list of available tests are shown here with details of the topics it's covering. Please choose a subject to start its test.

Introduction to Computing

Total Marks 20	Total Time 30 mins	Take Test
--------------------------	------------------------------	------------------

This test includes 4 topics:

Computer Architecture
Computer architecture is a specification detailing how a set of software and hardware technology standards interact to form a computer system or platform. In short, computer architecture refers to how a computer system is designed and what technologies it is compatible with.

Graphics
A graphic is an image or visual representation of an object. Therefore, computer graphics are simply images displayed on a computer screen. 2D graphics come in two flavors – raster and vector. Raster graphics are the most common and are used for digital photos, Web graphics, icons, and other types of images.

Networking
A computer network or data network is a telecommunications network which allows computers to exchange data. In computer networks, networked computing devices exchange data with each other using a data link. The connections between nodes are established using either cable media or wireless media.

Binary Numbers
The binary number system is a Base-2 numbering system which follows the same set of rules in mathematics as the commonly used decimal or base-10 number system. So instead of powers of ten, (10ⁿ) for example: 1, 10, 100, 1000 etc, binary numbers use powers of two, (2ⁿ) effectively doubling the value of each successive bit as it goes, for example: 1, 2, 4, 8, 16, 32 etc.

> Principles of Management

> Information Technology Basics

> Database Management Systems

PrepExpert © 2022 | All Rights Reserved.

PREPEXPERT

Dashboard

ASSESSMENT

Take Test

Reports

PREPARATION

Guided Learning

Resources

Finn Shelby
fin@peaky.com

TEST STARTED

Principles of Management

In this course, you will learn to recognize the characteristics of proper management by identifying what successful managers do and how they do it. Understanding how managers work is just as beneficial for the subordinate employee as it is for the manager. This course is designed to teach you the fundamentals of management as they are practiced today.

PROGRESS

TIME LEFT
19:56

End Test

Question 1

Question 1 / 4

In a large company, a strategy that sets the long term direction for the entire company is called ?

- Corporate
- Senior Level
- Functional
- Board

Confirm

PrepExpert © 2022 | All Rights Reserved.

4.8.3.3. Detailed Test Report

PREPEXPERT
Finn Shelby
fin@peaky.com

- Dashboard
- ASSESSMENT
- Take Test
- Reports
- PREPARATION
- Guided Learning
- Resources

Test Result

Topics Score

Topic	Your Score	Peers Average Score
Planning	50%	59%

Concepts Score

Concept	Your Score	Peers Average Score
Operational Planning	34%	-
Strategic Planning	100%	-
Tactical Planning	0%	-
Contingency Planning	100%	-

Need Improvement 😞

Test Score: 2 out of 4 Points

You scored better than **0%** of your peers.

Your Score: 50%

Peers Average: 59%

Completion Time: 2 minutes

Review Your Session

Have a look at your attempts to analyze your test performance. Not your best attempt? [Retake Test](#)

[Review Questions](#)

Guided Learning

Based on your weak areas we have some recommendations.

[Improve Score](#)

PrepExpert © 2022 | All Rights Reserved.

4.8.3.4. Test Records / History

PREPEXPERT
Finn Shelby
fin@peaky.com

- Dashboard
- ASSESSMENT
- Take Test
- Reports
- PREPARATION
- Guided Learning
- Resources

Reports

You can view your test history here.

SR.	SUBJECT	SCORE	DATE TAKEN	DETAILED REPORT
1	Database Management Systems	7 / 20	10-05-2022	View Report
2	Information Technology Basics	3 / 3	10-05-2022	View Report
3	Principles of Management	3 / 4	10-05-2022	View Report
4	Introduction to Computing	6 / 8	10-05-2022	View Report

Rows per page: 5 ▾ 1-4 of 4 < >

PrepExpert © 2022 | All Rights Reserved.

49

4.8.3.5. Guided Learning on Subjects that need attention

The screenshot shows the PREPEXPERT dashboard with a sidebar on the left containing navigation options: Dashboard, Take Test, Reports, Guided Learning (highlighted), and Resources. The main content area is titled 'Guided Learning' and includes an introductory paragraph: 'We are providing you with guided learning which has highlighted all the weak areas in the tests you have performed until now. Click in the subject to view the intelligent analysis along with video resources.' Below this, there are three tabs: 'Subjects', 'Average Performance', and 'Preparation Content'. The 'Subjects' tab is active, displaying 'Database Management Systems' with a description: 'Database Management Systems (DBMS) are software systems used to store, retrieve, and run queries on data. A DBMS serves as an interface between an end-user and a database, allowing users to create, read, update, and delete data in the database.' To the right of the description, there is a 'View Improvements' button. The footer of the page reads 'PrepExpert © 2022 | All Rights Reserved.'

4.8.3.6. Preparation Content on Weak Topics & Concepts

The screenshot shows the PREPEXPERT dashboard with the 'Preparation Content' section for 'Database Management Systems'. The sidebar on the left is the same as in the previous screenshot. The main content area is titled 'Database Management Systems' and includes a 'Retake Test' button. Below the title, there is a paragraph: 'Database Management Systems (DBMS) are software systems used to store, retrieve, and run queries on data. A DBMS serves as an interface between an end-user and a database, allowing users to create, read, update, and delete data in the database.' This is followed by another paragraph: 'Below are the weak and average topics in the current subject. You can refer to the given resources for each concept for better preparation.' A 'Topic 1' tab is active, showing 'Basic Database Concepts' with a description: 'In this topic we would discuss what is DBMS, DBMS components, advantages & disadvantages.' Below this, there are five sub-topics, each with a description, a performance indicator, and a 'Videos' button:

- What is Database?** (Average Performance) - Description: 'We consider a database to be a collection of related data and a database management system (DBMS) to be the software that manages and controls access to the database.'
- DBMS Components** (Poor Performance) - Description: 'The database management system can be divided into five major components, they are: Hardware; Software; Data; Procedures; Database Access Language.'
- Advantages of DBMS** (Poor Performance) - Description: 'Advantages include increased end-user productivity, recovery and backup, simplicity, faster data access, data security and integrity etc.'
- Disadvantages** (Poor Performance) - Description: 'Disadvantages include complexity, size, cost, performance and greater impact of failure etc.'
- Roles in Database Environment** (Poor Performance) - Description: 'These roles include data & database administrators, database designers, application developers, end-users & sophisticated users.'

 The footer of the page reads 'PrepExpert © 2022 | All Rights Reserved.'

4.8.3.7. View Resources

The screenshot shows the PREPEXPERT user interface. On the left is a navigation sidebar with sections: DASHBOARD, ASSESSMENT (Take Test, Reports), PREPARATION (Guided Learning, Resources), and ACCOUNT SETTINGS (Profile, Log Out). The main content area is titled 'Introduction to Computing' and displays a progress indicator of 7 Tests Taken and a 60% Average Score. Below this, there is a 'Retake Test' link and a 'Topic 1' section for 'Computer Architecture'. This section includes a definition of computer architecture and three sub-topics, each with a 'Videos' button: 1. Architectural Patterns, 2. System Design, and 3. Instruction Set Architecture.

4.8.3.8. Content Manager Dashboard

The screenshot shows the PREPEXPERT Content Manager Dashboard. The top right corner displays the user's name 'Thomas' and email 'thomas@shelby.itd'. The dashboard features a summary row with four cards: Total Domains (3), Total Subjects (5), Total Topics (14), and Total Concepts (28). Below this is a 'Manage Content' section with a 'Manage Content' button. The bottom section contains three resource management cards: 'Question Bank' (45 total questions) with a 'View Question Bank' button, 'Article Resources' (6 total article resources) with an 'Upcoming Feature' button, and 'Video Resources' (4 total video resources) with a 'Manage Video Resources' button. A footer at the bottom right reads 'PrepExpert © 2022 | All Rights Reserved.'

4.8.3.9. Content Manager - Manage Test Content

PREPEXPERT Thomas thomas@shelby.td

Dashboard

CONTENT MANAGEMENT

Test Content

Question Bank

Resources

Manage Domains

Search Domains [Add Domain](#)

SR. #	DOMAIN NAME	DOMAIN DESCRIPTION	ACTION	
1	Computer Science	Computer science is the study of computers and computing as ...	Edit	Delete View Subjects
2	Business Administration	In simple terms, business administration is the work of mana...	Edit	Delete View Subjects
3	Software Engineering	Software engineering is a detailed study of engineering to t...	Edit	Delete View Subjects

Rows per page: 5 1-3 of 3

PrepExpert © 2022 | All Rights Reserved.

PREPEXPERT Thomas thomas@shelby.td

Dashboard

CONTENT MANAGEMENT

Test Content

Question Bank

Resources

- Videos

[Computer Science](#) > [Introduction to Computing](#) > [Computer Architecture](#) > [Concepts](#)

Manage Concepts

Search Concepts [Add Concept](#)

SR. #	CONCEPT NAME	CONCEPT DESCRIPTION	ACTION			
1	Architectural Patterns	An architectural pattern is a general, reusable solution to ...	Edit	Delete	View Questions	Video Resources
2	System Design	System design is the process of defining the components, mod...	Edit	Delete	View Questions	Video Resources
3	Instruction Set Architecture	An Instruction Set Architecture (ISA) is part of the abstrac...	Edit	Delete	View Questions	Video Resources
4	Microservices	Microservices architecture is an approach in which a single ...	Edit	Delete	View Questions	Video Resources
5	Von Neumann Architecture	Von Neumann architecture is the design upon which many gener...	Edit	Delete	View Questions	Video Resources

Rows per page: 5 1-5 of 6

PrepExpert © 2022 | All Rights Reserved.

PREPEXPERT Thomas
thomas@shelby.ltd

[Dashboard](#)

CONTENT MANAGEMENT

- [Test Content](#)
- [Question Bank](#)
- Resources**
 - Videos

Architectural Patterns

List of all related video resources available.

[Link Video Resource](#)

SR. #	NAME	DESCRIPTION	EMBED CODE	WEBSITE URL	ACTION
1	Microservices Architectural Patterns	* Pattern description and basic concepts * Microservices implementation approaches * Pros and cons...	https://www.youtube.com/watch?v=8BPDv038oMI	https://www.youtube.com/watch?v=8BPDv038oMI	Unlink Resource
2	Normalization Video Tutorial	Database normalization is the process of structuring a database, usually a relational database....	https://youtu.be/lXycPq7MnwE	https://youtu.be/lXycPq7MnwE	Unlink Resource
3	Von Neumann Architecture	Von Neumann architecture is the design upon which many general purpose computers are based....	https://www.youtube.com/watch?v=Ml3-kVYLNr8	https://www.youtube.com/watch?v=Ml3-kVYLNr8	Unlink Resource

Rows per page: 5 1-3 of 3

PrepExpert © 2022 | All Rights Reserved.

4.8.3.10. Content Manager - Question Bank

PREPEXPERT Thomas
thomas@shelby.ltd

[Dashboard](#)

CONTENT MANAGEMENT

- [Test Content](#)
- Question Bank**
- [Resources](#)

Question Bank

Choose Domain, Subject, Topic and Concept to see related questions in the question bank.

[+ Add Question](#)

Select Domain:

Subject: Topic: Concept:

SR. #	STATEMENT	OPTION 1	OPTION 2	OPTION 3	OPTION 4	CORRECT OPTION	DIFFICULTY LEVEL	ACTION
1	Skype, BitTorrent, Facebook, Google Apps, SAP are:	Platform as a Service	Software as a Service	Infrastructure as a Service	None	Software as a Service	Medium	Edit Delete
2	Refactor is a tactic to achieve	Reduced Coupling	Increased Cohesion	Defer Binding	Reduced size of Module	Reduced Coupling	Medium	Edit Delete
3	In Service Oriented Architecture	SOAP is easier to use than REST	Services can be developed using different technologies	A system makes use of services provided by other system	A system registry stores end points of different services	SOAP is easier to use than REST	Difficult	Edit Delete
4	In a microkernel architecture adapter does not interact directly with	Microkernel	Client	External Server	Internal Server	Internal Server	Difficult	Edit Delete
5	Architectural pattern establishes the relationship between:	Architectural styles	Context, problem, solution	Architectural reference model, architectural pattern & reference architecture	all of these	all of these	Medium	Edit Delete

Rows per page: 5 1-5 of 45

PrepExpert © 2022 | All Rights Reserved.

4.8.3.11. Content Manager - Manage Resources

Manage Video Resources

List of all video resources available.

Search Video Resources | Add Video Resource +

SR.#	NAME	DESCRIPTION	EMBED_CODE	WEBSITE_URL	ACTION
1	Normalization Video Tutorial	Database normalization is the process of structuring a database, usually a relational database....	https://youtu.be/IXycPq7MnwE	https://youtu.be/IXycPq7MnwE	Edit Delete
2	Microservices Architectural Patterns	* Pattern description and basic concepts * Microservices implementation approaches * Pros and cons...	https://www.youtube.com/watch?v=8BPDv038oMI	https://www.youtube.com/watch?v=8BPDv038oMI	Edit Delete
3	Von Neumann Architecture	Von Neumann architecture is the design upon which many general purpose computers are based....	https://www.youtube.com/watch?v=Ml3-KVYLNt8	https://www.youtube.com/watch?v=Ml3-KVYLNt8	Edit Delete
4	Example Resource Edited	Example...	embed	example.com	Edit Delete

Rows per page: 5 ▾ 1-4 of 4 < >

PrepExpert © 2022 | All Rights Reserved.

4.8.3.12. Admin Dashboard

PREPEXPERT

Manal Talat
manal@gmail.com

Dashboard

USER MANAGEMENT

- All Users
- Test Records

CONTENT MANAGEMENT

- Test Content
- Question Bank
- Resources >

Total Domains **3** | Total Subjects **5** | Total Topics **14** | Total Concepts **28**

Manage Content

Admin can manage all the domains, subjects, topics, concepts and related questions. Choose a specific domain, subject, topic and concept to view its related questions.

[Manage Content](#)

Manage Students

Admin can manage all students registered and taking tests. Choose a student and also change its credentials; login information.

[Manage Students](#)

Manage Content Managers

Admin can manage the data of all the content managers. Choose a content manager and also manage its credentials.

[Manage Content Managers](#)

Total Questions **45** | Total Test Records **9** | Total Video Resources **4**

Question Bank

The question bank has all the questions of the related concepts needed.

[View Question Bank](#)

Test Records

View the test history of all the tests taken by registered students.

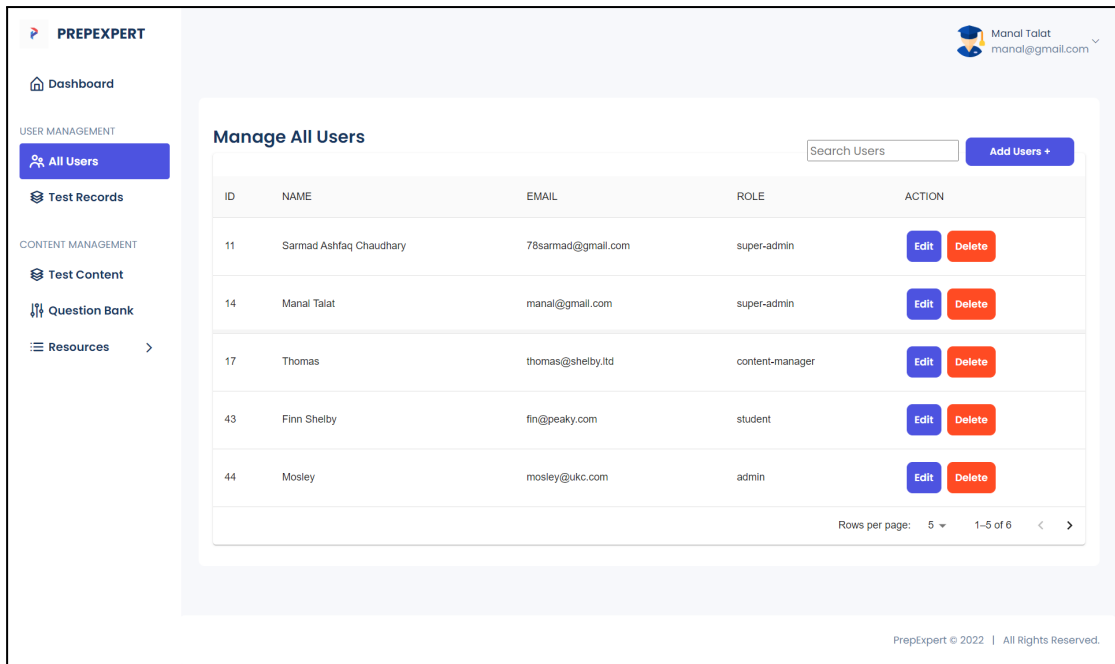
[View Test Records](#)

Video Resources

Add, update or delete related and relevant video resources.

[Manage Video Resources](#)

PrepExpert © 2022 | All Rights Reserved.



PREPEXPERT

Manal Talat
manal@gmail.com

Dashboard

USER MANAGEMENT

- All Users
- Test Records

CONTENT MANAGEMENT

- Test Content
- Question Bank
- Resources

Manage All Users

Search Users Add Users +

ID	NAME	EMAIL	ROLE	ACTION
11	Sarmad Ashfaq Chaudhary	78sarmad@gmail.com	super-admin	Edit Delete
14	Manal Talat	manal@gmail.com	super-admin	Edit Delete
17	Thomas	thomas@shelby.ltd	content-manager	Edit Delete
43	Finn Shelby	fin@peaky.com	student	Edit Delete
44	Mosley	mosley@ukc.com	admin	Edit Delete

Rows per page: 5 1-5 of 6

PrepExpert © 2022 | All Rights Reserved.

4.9. Conclusion

Hence, in this chapter we have discussed the architectural details and the logical structure of our system using domain models and class diagrams. Sequence and Activity diagrams show the dynamic view and how the system acts in various situations.

Chapter 5

System Implementation

For the implementation details of how we were able to successfully fulfil the user requirements, we will discuss the frontend and backend implementation and highlight key tools, technologies and libraries used.

Frontend is the **React.js** web application and the backend is in **Python** with **FastAPIs**. The key functionalities are discussed below:

5.1. Tools, Technologies and Libraries Used

Following are the key concepts in the backend implementation.

5.1.1. FastAPI

Our system uses fastAPI for making the endpoints for our web application. The fastAPI is a modern web framework for making APIs with Python. FastAPI has following benefits:

1. Faster and easier to code
2. Better security and authentication features
3. Uses **pydantic** for making schemas and detecting data types at runtime
4. Has **Swagger UI** for testing of API endpoints with extension **/docs** and for better understandability uses extension **/redoc**.
5. We have used **_security.OAuth2PasswordRequestForm** as a class dependency from the security module of fastAPI which generates tokens for user authentication. If the user does not exist or there are invalid credentials we have imported **HTTPException** from fastAPI to return the error status code 401 and for other error handling purposes as well.

5.1.2. passlib[bcrypt] hash

passlib is the password hashing library for Python. **passlib.hash** contains all the password hashing algorithms built into the library. We have used **BCrypt** modular format from that module. This is used in:

1. verifying the password while changing password
2. creating user

5.1.3. starlette Middleware

Starlette is a ASGI (Asynchronous Server Gateway Interface) framework for creating async web services in Python. Starlette provides several middleware classes but we are using **CORSMiddleware** for addition of CORS headers to the outgoing responses. This is done so that cross-origin requests from browsers are allowed.

5.1.4. NumPy for Scientific Computing

We have used two functions from NumPy library:

1. `numpy.amin()`: for calculating minimum of an array or minimum along axis
2. `numpy.amax()`: for calculating maximum of an array or maximum along axis
3. `numpy.empty()`: for creating an empty NumPy array by specifying its type and row and columns as shape.
4. `numpy.append()`: for adding value to the end of the array.

5.1.5. SQLAlchemy

To streamline the workflow and query the data more efficiently, **SQLAlchemy**(an open-source SQL toolkit) is used for object relational mapping(ORM). It helps in the communication between the Python programs and the databases.

1. Object relational mapping is a programming technique in which the database records are represented as objects and those are used to indirectly and effortlessly interact with the database.
2. SQLAlchemy uses reflection to automatically load tables from the database and builds the metadata.
3. To establish and maintain a session with the database we used **sessionmaker()** from SQLAlchemy orm. The session can be directly instantiated but to standardise all the configurations we use this method.
4. An SQL **engine** is created first which is bound to the session.
5. Next, we have used a factory function **declarative_base()** that returns a base/meta class from where the entities are inherited. The table and mapper are automatically generated after this.

5.1.6. Redis

1. Redis is an open source in-memory, key-value store that can be used as a cache and database.

2. It provides data structures such as strings, hashes, sets, sorted sets, bitmaps etc.
3. It has ultra-fast performance providing sub-milliseconds response times and latency.
4. It can also be used as a session store.
5. It is very helpful in our system for the real time analytics and calculations so that the users can have a seamless experience.
6. Keys are the unique identifiers and can be used to retrieve or set the values such as strings, hashes, sets etc.

The key value pairs saved in Redis are:

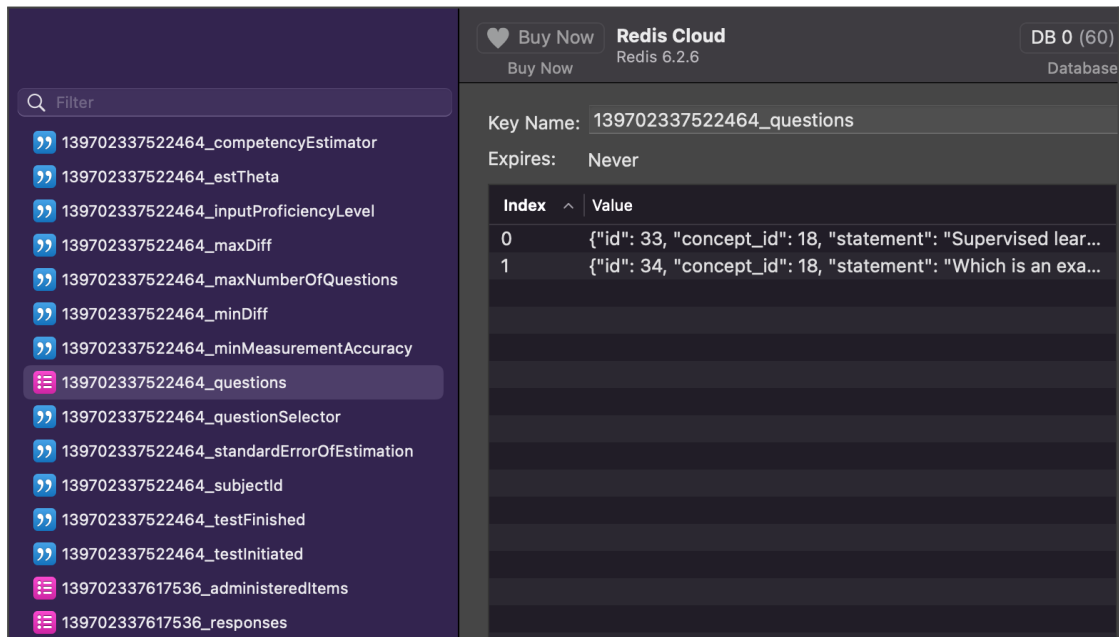
1. **subjectId:** returns current subject id
2. **testId:** returns current test id
3. **administeredItems:** question ids of the questions already asked
4. **Responses:** returns a record of previous answers in boolean
5. **responseVector:** returns questions with their answers
6. **maxNumberOfQuestions(maxItemStopper):** 20
7. **minMeasurementAccuracy(minErrorStopper)**
8. **inputProficiencyLevel**
9. **questionSelector**
10. **competencyEstimator**
11. **standardErrorOfEstimation**
12. **testInitialized:** returns boolean
13. **testFinished:** returns boolean
14. **estTheta:** returns current performance

Main functions of redis used:

1. **redis.set():** This function sets the string value against a key. If the key already holds a value then it is overwritten.
2. **redis.get():** This function returns the value of key, if the key doesn't exist it returns "nil" and this function only handles string values.
3. **redis.rpush():** The specified values are inserted at the tail of the list stored at key. If the key doesn't exist, first it creates an empty list before push operation. Through this function it is possible to push multiple elements using a single command.

4. **redis.del():** This function is used to remove the key followed by the name of the key you want to delete.
5. **redis.lrange():** This function is used to return the specific range of elements stored in the key. The offsets are given as “start” or “stop” as the range.
6. **redis.llen():** This function returns the length of the list stored at a key, if the list does not exist, it is considered an empty list and returns 0.

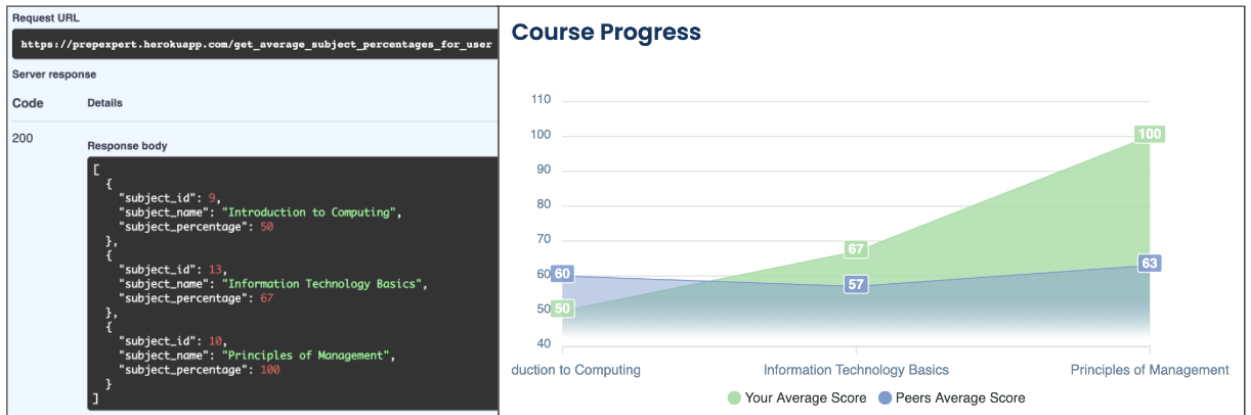
Caching the question data in redis:



5.1.7. Chart.js and ApexCharts.js

These are the two key **JavaScript** libraries used in the frontend implementation of our **React** web application. These are used for data visualisation purposes in the form of charts so that users can have a better understanding of their current stats. We have rendered these charts according to the user’s data dynamically and integrated with the APIs.

For example, the API **get_average_subject_percentage_for_user()** returns the percentage of all the subjects whose test the user has taken and we have rendered that data in the **Area Chart**.



5.2. Algorithms Used

5.2.1. catsim (CAT Simulation Python Package)[5]

catsim is a Python package for computerised adaptive testing(CAT) simulation. Item Response Theory(IRT) had mathematical models with functionalities in R language but they weren't present in Python before. This package provides the functionalities of CAT in Python which were made using scientific packages of **NumPy** and **SciPy**.

The basic CAT lifecycle is given as below:

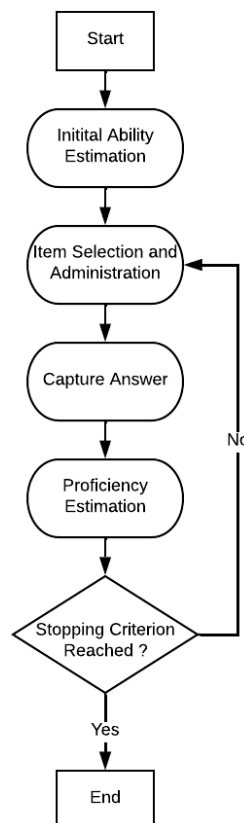


Figure 5.1: Basic CAT Life cycle

Four main steps in computerised adaptive testing(CAT) are:

1. Estimation of student's initial ability
2. Next item selected based on current ability estimated
3. Ability re estimation based on all responses/ proficiency estimation
4. Checking stopping criterion

5.2.1.1. Item Matrix

In **catsim** the items are represented as **numpy.ndarray** in which the items with their parameters are represented as rows and columns. The matrix consists of 4 columns showing the 4 parameters i.e. a,b,c and r. The item matrix is generated by a function **cat.generate_item_bank()**. As an example you can see,

$$\text{> generate_item_bank}(5, '3PL', \text{corr}=0.5)$$

(2.1)

Above function creates a 3PL simulation consisting of 5 items and they have a correlation of **0.5**. After the simulation, the fourth column denoting r is generated. This depicts the exposure rate and tells how many times an item has been used.

$$r_i = \frac{q_i}{N}$$

(2.2)

Where **N** is the total number of tests and **q_i** shows the number of times the item **i** is used.

5.2.1.2. Question Bank Generation

As we have generated our own item bank instead of random items as in simulation therefore, we have used our own question bank for this purpose. One of the main duties of the content manager is to manage the question bank. Each question in the question bank is mapped to the concept it belongs to in the software engineering domain. The content manager can add the question by specifying its:

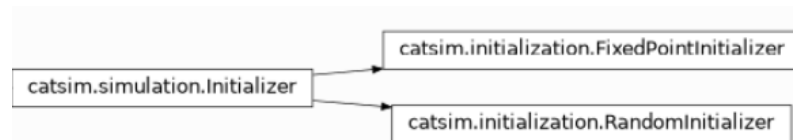
1. Statement
2. Options
3. Correct Option
4. Difficulty Level(1 for Easy, 3 for Medium, 5 for Hard)

5. The concept it belongs to

So, when the student takes the test, `generate_question_bank()` is called for that subject. Easy MCQs are chosen first to judge the current ability then gradually the difficulty level is varied according to his performance in the previous MCQs. Record of the responses and the administered items (questions already asked) is maintained.

5.2.1.3. Initialization

At the start of the test the ability θ is initialised. This can be of two types:



- FixedPointInitializer: fixed value with which the ability of all the students are initialised
- RandomInitializer: picks up any random value from normal distribution

In our system we have used FixedInitializer to initialise input Proficiency level as 0 for all students.

5.2.1.4. Selection

When the student's ability is estimated, the next item to be asked from the student is selected through the **Selector** methods from its base class. There are many item selection methods but we have used the one with maximum gain of information, that is, **MaxInfoSelector()**.

MaxInfoSelector() returns the first non-administered item which gives the maximum information. The parameters of this function include:

1. Estimated Theta: float containing current ability of the student
2. Index: current index of the student
3. Items: matrix containing item parameters
4. Administered Items: indexes of the items that were already administered

This method returns the index of the next item to be administered. So, the API we have made, which is **get_next_question()**, selects the next question based on the user's proficiency and which gives maximum information. The working can be more understood by the figure below:

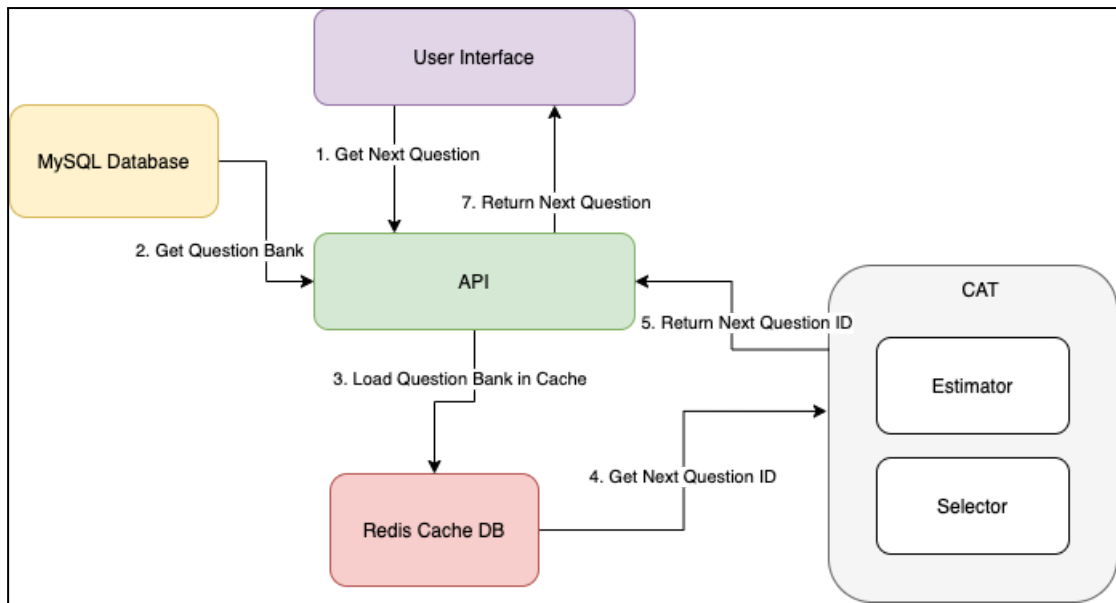


Figure 5.2: Get Next Question

5.2.1.5. Proficiency Estimation

The estimated ability of the student is calculated through the standard error of estimation. Whenever the student performs and whenever a student performs the mcq, after each attempt the ability is updated. To calculate the new value of the student's ability the **estimator.estimate()** function takes the dichotomous(binary) response vector along with its results, administered items and estimated ability as its parameters.

catsim.irt.see() calculates the standard error of estimation(SEE) of a test at a specific value and takes the student's ability value and the response vector as parameters . It mathematical function is:

$$SEE = \sqrt{\frac{1}{I(\theta)}}$$

(2.4)

So, therefore

Estimator	Differential Evolution Estimator (estimates user proficiency)
Selector	Max Info Selector (selects next question based on user proficiency)

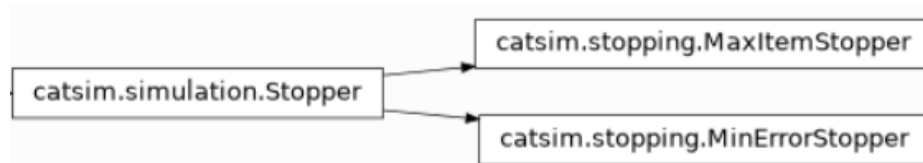
Table 5.1: Estimator Selector Review

5.2.1.6. Stopping Criterion

Two conditions must be met in order to stop the test:

1. When the limit of maximum number of questions is reached
2. Standard Error of Estimation(estimation of accuracy of predictions) reaches lower threshold

There are two methods defined to check these two conditions:



(2.4)

1. MaxItemStopper: which is the maximum number of questions and in our case it's **20**. It takes the administered items and current index of the student as parameters and returns a boolean whether the maximum question limit is reached or not.
2. MinErrorStopper: minimum measurement accuracy which is **0.8** in our case. In addition to this, it takes in ability value θ and administered items to return whether it has met the stopping criterion or not, in boolean.

5.3. Implementation Strategy`

The implementation strategy was as follows:

1. Requirements:
 - a. Requirements Elicitation
 - i. Gathering Stakeholder Requirements
 - ii. Requirements Prioritisation
 - iii. Competitive Analysis of the Available Systems
 - b. Requirements Analysis
 - i. Building User Stories
 - ii. Requirements Traceability Matrix
 - c. Software Requirements Specification
2. Design
 - a. Schema Design
 - b. Interface Design
 - i. Wireframes

- ii. Prototypes
 - c. Software Design Document
- 3. Development
 - a. Front End Development
 - i. Comparing Tools for Developing FrontEnd
 - ii. Dashboards Development in **React.js**
 - b. Back End Development
 - i. Database Setup
 - ii. API Development using **FastAPIs**
 - iii. Backend services in **Python**
- 4. Integration
 - a. Unit Testing
 - b. Module Integration
 - c. Integration Testing
- 5. Testing
 - a. Designing Test Suite
 - b. System Testing
- 6. Deployment
 - a. Deploy front end on netlify
 - b. Backend deployment at Heroku

The frontend and backend implementation was performed in parallel before integration and testing.

5.4. Conclusion

Therefore, in this chapter we have highlighted the key functionality of our project. We have discussed the important libraries used in the backend implementation and frontend implementation. The main algorithm **catsim** is discussed with the implementation strategies.

Chapter 6

System Testing & Evaluation

6.1. Test Strategy

So, in order to ensure the working system is performing according to the expected outputs, we have performed different testing techniques. The functional and non-functional requirements are given due importance to guarantee the quality of the system. We made sure there isn't any loophole remaining which disturbs the seamless user experience.

6.2. Component Testing

After the unit testing, component testing is performed. Various units making up the modules are tested independently with the real data instead of dummy data. To ensure each module of the system is working fine before integrating them together, we perform this type of testing.

6.3. Unit Testing

During the development, each unit that was developed by us (API endpoints, functions, classes) were tested. This was to ensure the design requirements are being fulfilled. This type of testing is performed at the lowest level. Therefore, we have tested minute functionalities like whether the student's ability is calculated correctly, whether the correct parameters are passed to the functions, whether the CRUD operations are working fine etc.

6.4. Integrated Testing

To check whether the independent components or units are performing properly together or not we perform integrated testing. As our backend was in Python with fastAPIs therefore, we had to integrate that with our frontend in React.js. After integration, the CRUD operation is verified to see if the integration was successful or not then the desired functionality is achieved.

We have used the APIs in numerous places like fetching user's data, test information, managing resources as well as for the taking test in which we have used the **catsim** algorithm.

6.5. System Testing

To ensure that the system fulfils all the requirements and to evaluate its performance we have done system testing. The validity of the requirements is tested and instead of going into the code or inner logic the overall functionality and behaviour of the system is considered which is black box testing.

6.6. Test Cases

6.6.1. Authentication

6.6.1.1. Create Account

Test Case ID:	TC-01
Scenario Description	The user can create an account on the system.
Input	The user clicks the “Create Account” button on the website.
Pre-condition	The user is not registered.
Expected Output	<ol style="list-style-type: none">1. The user opens the registration page.2. The user enters the required information and clicks the “Submit” button.3. System validates the user information.
Actual Output	The user is able to create account successfully after the validation of user information
Pass/Fail	Pass

6.6.1.2. Sign In (Multi-tenant)

Test Case ID:	TC-02
Scenario Description	The user can sign in to the system.
Input	The user clicks the “Sign In” button.
Pre-condition	The user must have a valid account on the system.
Expected Output	<ol style="list-style-type: none">1. The user opens the sign in page.2. The user enters their sign in credentials.3. System verifies the credentials from the database.
Actual Output	The user is able to sign in to the account successfully.

Pass/Fail	Pass
-----------	------

6.6.1.3. Forgot Password

Test Case ID:	TC-03
Scenario Description	The user can sign in to the system.
Input	The user clicks the “Sign In” button.
Pre-condition	The user must have a valid account on the system.
Expected Output	<ol style="list-style-type: none"> 1. The user opens the sign in page. 2. The user enters their sign in credentials. 3. System verifies the credentials from the database.
Actual Output	The user is able to sign in to the account successfully.
Pass/Fail	Pass

6.6.2. Profile Management

6.6.2.1. View Account Information

Test Case ID:	TC-04
Scenario Description	The user can view their account information.
Input	The user clicks on the “Profile” icon in the navigation menu.
Pre-condition	The user must be signed in to the system.
Expected Output	The user is able to see his account information
Actual Output	After clicking the “Profile” icon, the user is able to see his account information successfully.
Pass/Fail	Pass

6.6.2.2. Update Account Information

Test Case ID:	TC-05
Scenario Description	The user can update their account information.
Input	The user clicks on the “Update Profile” button on the Account Information page.
Pre-condition	The user must be signed in to the system.
Expected Output	<ol style="list-style-type: none">1. The user is navigated to the page with editable account information fields.2. The user updates the name and clicks the “Update” button.
Actual Output	After validating the entered account information, the system updates the account information successfully.
Pass/Fail	Pass

6.6.3. Test

6.6.3.1. Take Test

Test Case ID:	TC-06
Scenario Description	The student can take tests of the available subjects.
Input	The user clicks on the “Start” button under test information.
Pre-condition	The user must be signed in to the system.
Expected Output	The user is able to take the test.
Actual Output	<ol style="list-style-type: none">1. The user navigates to the Take Test page.2. System shows a list of all the tests (subject-wise) that are available to the user.3. The user clicks the “Start Test” button to start the test which navigates the user to the Test screen.4. On the test screen, the system shows test information, progress and time remaining to the user and starts the test.5. System generates each question using the CAT algorithm on the user's test answers and previous user proficiency in the subject (if any).6. Each question is offered on the basis of the user's performance, i.e. if the user performs poorly then he's offered maximum questions of “Easy” difficulty level.

	<ol style="list-style-type: none"> 7. Users cannot skip any questions but can discard or end the test. 8. On completing the test, the system saves the test progress to the database and shows the Test Result to the user.
Pass/Fail	Pass

6.6.3.2. View Detailed Performance Report

Test Case ID:	TC-07
Scenario Description	The user can view their detailed performance report.
Input	The user clicks on the “Profile” icon in the navigation menu.
Pre-condition	The user must be signed in to the system.
Expected Output	The user is able to see the reports of all the tests that he has taken up till now.
Actual Output	<ol style="list-style-type: none"> 1. The user completes a test and is navigated by the system to the “Reports” page. 2. The user selects the test. 3. Detailed Performance Report page shows: <ol style="list-style-type: none"> a. User’s test score, completion time, subject percentile. b. User’s performance topic-wise and concept-wise along with a comparison chart with the peers (other users performing the same test). c. Option to review the test session and see all the correct and incorrect attempts along with their correct answers. d. Option to view preparation content in order to improve their subject performance.
Pass/Fail	Pass

6.6.3.3. Access Guided Learning

Test Case ID:	TC-08
Scenario Description	The user can access the guided learning highlighting weak areas that need improvement.
Input	The user clicks on the “Guided Learning” in the detailed performance report or from the sidebar menu by selecting the subject.
Pre-condition	The user must be signed in to the system.

Expected Output	The user is able to see the highlighted weak areas of all subjects whose test he has taken so far.
Actual Output	<ol style="list-style-type: none"> 1. The user completes a test and views a Detailed Performance Report. 2. From the Detailed Performance Report, the user clicks the “Improve Score” button that takes the user to the Guided Learning page. 3. Guided Learning page highlights all the weak topics and concepts of the related subject and shows Resources to improve the test score.
Pass/Fail	Pass

6.6.3.4. View Resources/Preparation Content

Test Case ID:	TC-09
Scenario Description	The user can access the guided learning highlighting weak areas that need improvement.
Input	The user can view all the preparation content/resources.
Pre-condition	The user must be signed in to the system.
Expected Output	The user is able to see the highlighted weak areas of all subjects whose test he has taken so far.
Actual Output	<ol style="list-style-type: none"> 1. The user navigates to the Resources page from the navigation menu. 2. Resources page lists down all the subjects along with the options to view resources. 3. On clicking the “View Resources” button against any subject, the system shows Resources against all the subject topics and the included concepts. 4. Concepts have multiple resources, shown in the form of embed content, such as videos from YouTube, Vimeo, etc.
Pass/Fail	Pass

6.6.3.5. End Test

Test Case ID:	TC-10
Scenario Description	The user is able to end the test at any point.
Input	The user clicks on the “End Test” button shown at the top while he is taking the test.
Pre-condition	The user must be signed in to the system and must be taking a test.

Expected Output	The user is able to end the test after clicking on the button.
Actual Output	<ol style="list-style-type: none"> 1. The user clicks the “End Test” button while taking the test 2. An alert is shown to the user with a message that his current progress won’t be saved if he ends. 3. The user confirms and is navigated back to the available tests page without saving the current progress.
Pass/Fail	Pass

6.6.4. Administration

6.6.4.1. Manage Admins

Test Case ID:	TC-12
Scenario Description	Super admin can manage admin accounts and information.
Input	The user clicks on the Manage Admins from the menu.
Pre-condition	The user must be signed in to the system as Super Admin.
Expected Output	The super admin is able to add, update or delete admins.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the admin dashboard and clicks on Manage Admins from the menu. 2. System shows a list of admins that are registered. 3. The user clicks the “Add” button, fills in the admin details to add a new admin to the system. 4. The user clicks the “Edit” or “Delete” button against an entry to modify or remove that admin.
Pass/Fail	Pass

6.6.4.2. Manage Students and Content Managers

Test Case ID:	TC-13
Scenario Description	Super Admins & Admins can manage students and content manager’s accounts and information.
Input	The user clicks on the Manage Users from the menu.
Pre-condition	The user must be signed with an admin/super admin account..
Expected Output	The super admin/admin are able to add, delete or update students and content managers.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the admin dashboard and clicks on Manage Users from the menu. 2. System shows a list of users that are registered.

	<ol style="list-style-type: none"> 3. The user clicks the “Add” button, fills in user details to add a new user to the system. 4. The user clicks the “Edit” or “Delete” button against an entry to modify or remove that admin.
Pass/Fail	Pass

6.6.5. Test Management

6.6.5.1. Manage Domains

Test Case ID:	TC-15
Scenario Description	The user can manage the domain from where the test is taken from.
Input	The user clicks on the Manage Domains from the menu.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	The user is able to manage the domains for the test.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on Manage Domains under the Test Content in the menu. 2. System shows a list of Domains that are included in the test. 3. The user adds a new Domain by clicking the “Add” button on top of the list. 4. The user modifies the existing domains by clicking the “Edit” or “Delete” button against the domain entries.
Pass/Fail	Pass

6.6.5.2. Manage Subjects

Test Case ID:	TC-16
Scenario Description	The users can manage subjects in a domain.
Input	The user clicks on the “View Subjects” against each domain in the domain table.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the subject’s information in the database. In case of deleting a subject, the related topic and concepts are also deleted through object relational mapping.

Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on Domains and then “View Subjects”. 2. System shows a list of Subjects that are included in the domain. 3. The user adds a new Subject by clicking the “Add” button on top of the list and specifies the subject information. 4. The user modifies the existing subjects by clicking the “Edit” or “Delete” button against the subject entries.
Pass/Fail	Pass

6.6.5.3. Manage Topics

Test Case ID:	TC-17
Scenario Description	The user can manage topics in a subject.
Input	The user clicks on the “View Topics” against each subject in the Subjects table.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the topic’s information in the database. In case of deleting a topic, the related concept and questions are also deleted through object relational mapping.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and then “View Topics” on the Subjects table. 2. System shows a list of Topics that are included in that Subject. 3. The user adds a new Topic by clicking the “Add” button on top of the list and specifies the topic information. 4. The user modifies the existing topics by clicking the “Edit” or “Delete” button against the topic entries.
Pass/Fail	Pass

6.6.5.4. Manage Concepts

Test Case ID:	TC-18
Scenario Description	The user can manage concepts in a topic.
Input	The user clicks on the “View Concepts” against each topic in the Topics table.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the concept’s information in the database.

Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and then “View Concepts” on the Topics table. 2. System shows a list of Concepts that are included in that Topic. 3. The user adds a new Concept by clicking the “Add” button on top of the list and specifies the concept information. 4. The user modifies the existing concepts by clicking the “Edit” or “Delete” button against the concept entries.
Pass/Fail	Pass

6.6.5.5. Manage Questions

Test Case ID:	TC-19
Scenario Description	The user can manage questions in a concept.
Input	The user clicks on the View Questions option shown on the concepts page.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the questions in the database.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “View Questions” option in the concepts. 2. System shows a list of Questions that are included in the concept. 3. The user adds a new Question by clicking the “Add” button on top of the list and specifies the question details. 4. The user modifies the existing questions by clicking the “Edit” or “Delete” button against the question entries.
Pass/Fail	Pass

6.6.5.6. Manage Question Bank

Test Case ID:	TC-20
Scenario Description	The user can manage a question bank.
Input	The user clicks on the Question Bank from the sidebar menu.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the questions in the database.

Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “Question Bank” option in the sidebar menu. 2. System shows a list of all the questions in all the concepts. 3. The user adds a new Question by clicking the “Add” button on top of the list and specifies the question details along with the concept it belongs to. 4. The user modifies the existing questions by clicking the “Edit” or “Delete” button against the question entries.
Pass/Fail	Pass

6.6.5.7. Manage Linked Video Resources

Test Case ID:	TC-21
Scenario Description	The user can manage resources in a concept.
Input	The user clicks on the “View Resources” option from the concepts page.
Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the video resource in the database.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “View Resources” option in the concepts. 2. System shows a list of video resources that are linked to that concept. 3. The user adds a new video resource by clicking the “Link ” button on top of the list and specifies the video resource details. 4. The user modifies the existing video resource by clicking the “Edit” or “Delete” button against the video resource entries. 5. The user can unlink video resources in a concept by clicking on “Unlink Video Resource”. 6.
Pass/Fail	Pass

6.6.5.8. Manage Video Resources

Test Case ID:	TC-22
Scenario Description	The user can manage all video resources.
Input	The user clicks on the “Resources” option from the sidebar menu.

Pre-condition	The user must be signed as a Super Admin/Admin or Content Manager.
Expected Output	System updates the video resource in the database.
Actual Output	<ol style="list-style-type: none"> 1. The user opens the dashboard and clicks on the “Resources” option in the sidebar menu. 2. System shows a list of video resources that are available in general. 3. The user adds a new video resource by clicking the “Add” button on top of the list and specifies the video resource details. 4. The user modifies the existing video resource by clicking the “Edit” or “Delete” button against the video resource entries.
Pass/Fail	Pass

6.7. Results & Evaluation

Hence, through all the testing steps we have evaluated the system thoroughly and it is working according to the expected requirements.

6.8. Conclusion

Hence, the system testing and evaluation of our system is performed in order to determine if the various features of our system conform to the user requirements and satisfy all the guidelines and standards. The test cases try to cover all the aspects of the system to uncover any bugs/defects that were ignored so that the quality of the system is guaranteed.

Chapter 7

Conclusion

7.1. Contributions

Through this project we have successfully implemented the computerised adaptive testing system for the students enrolled in Software Engineering. Now, the students can easily prepare for their exams or quizzes through this system which maintains the record of each system and by giving personalised experience.

7.2. Reflections

While making this system, we knew that we had to do something about the current testing systems which offer no valuable insights rather simple questions with no actual learning value.

The strongest highlight of our project compared to the other systems is that it offers valuable data and analysis into the students' performance. Instead of preparing quizzes or for exams from the online available websites which provide a list of multiple choice questions, now students will focus on the preparation content before taking the test on our system. As we provide services to provide intelligent insights on the current knowledge level of the students therefore, the students can rely on our system for better preparation.

7.3. Future work

In the future, we have a set of functionality planned to further enhance the utility of our system. First up we are planning to add traffic insights, audience reach and user activity which shall help us to tailor our system best to the students' needs. In addition to that, we look forward to incorporating scoreboards and daily goals so that the students can assess their performance as well as track their progress against the time left for their exams.

REFERENCES

- [1] R. O. N. A. L. D. K. HAMBLETON and L. I. N. D. A. L. COOK, "Latent trait models and their use in the analysis of educational test data^{1,2,3}," *Journal of Educational Measurement*, vol. 14, no. 2, pp. 75–96, 1977.
- [2] F. Lord, "A Broad-Range Tailored Test of Verbal Ability", *Applied Psychological Measurement*, vol. 1, no. 1, pp. 95-100, 1977. Available: 10.1177/014662167700100115.
- [3] S. V. Jadhav, P. Jain, Y. Bhansali, S. Jain and D. Jain, "Adaptive and Automated Assessment System to Decide the Difficulty Level of Questions," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, pp. 1-4, doi: 10.1109/ICCUBEA.2018.8697433.
- [4] Xinming An Yiu-Fai Yung, "Item Response Theory: What It Is and How You Can Use the IRT Procedure to Apply It", 2014.
- [5]"Introduction", *Douglasrizzo.com.br*, 2022. [Online]. Available: <https://douglasrizzo.com.br/catsim/introduction.html>. [Accessed: 06- Jun- 2022].

APPENDIX A

Dictionary

DB	Database
SQL	Structured Query Language
CAT	Computerised Adaptive Learning
ReactJS	Frontend JavaScript Library
ASGI	Asynchronous Server Gateway Interface
Redis	Remote Dictionary Server
CATSIM	Computerised Adaptive Testing Simulator
ORM	Object Relational Mapping
IRT	Item Response Theory
SEE	Standard Error of Estimation