

EECS3311 Group #2: Course Compass

System Design Documentation

Table of Contents

CRC Cards.....3

Software Architecture Diagram.....6

Class Name : CourseSearch.tsx	
Subclass: CoursePage.css, CourseSearch.css	
Responsibilities: <ul style="list-style-type: none"> • Gets all courses from the database • Returns courses matching users applied search filters • Displays contents of SearchFilters.tsx, CourseList.tsx and CourseDetails.tsx on one page 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva

Class Name : App.tsx	
Responsibilities: <ul style="list-style-type: none"> • Creates an App rendering the imported component from CourseSearch.tsx • Exports App 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva •

Class Name : main.tsx	
Subclass: index.css	
Responsibilities: <ul style="list-style-type: none"> • Renders and displays the React web application imported from App.tsx 	Collaborators: <ul style="list-style-type: none"> • Gurjap Hajra

Class Name : connection.js	
Responsibilities: <ul style="list-style-type: none"> • Connect the server to MongoDB database 	Collaborators: <ul style="list-style-type: none"> • Gurjap Hajra

Class Name : courses.js	
Responsibilities: <ul style="list-style-type: none"> • Manages the stored list of courses in the database • Can get a course, add/delete courses and update a course 	Collaborators: <ul style="list-style-type: none"> • Gurjap Hajra

Class Name : server.js	
Responsibilities: <ul style="list-style-type: none"> • Sets up the server • Directs server requests related to courses(e.g get course) to courses.js 	Collaborators: <ul style="list-style-type: none"> • Gurjap Hajra

Class Name : CourseDetails.tsx	
Subclass: CoursePage.css, CourseProfessor.css	
Responsibilities: <ul style="list-style-type: none"> • Displays a more detailed view about each course • Display which professors are teaching that course • Separates information about course and professors into different tabs 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva

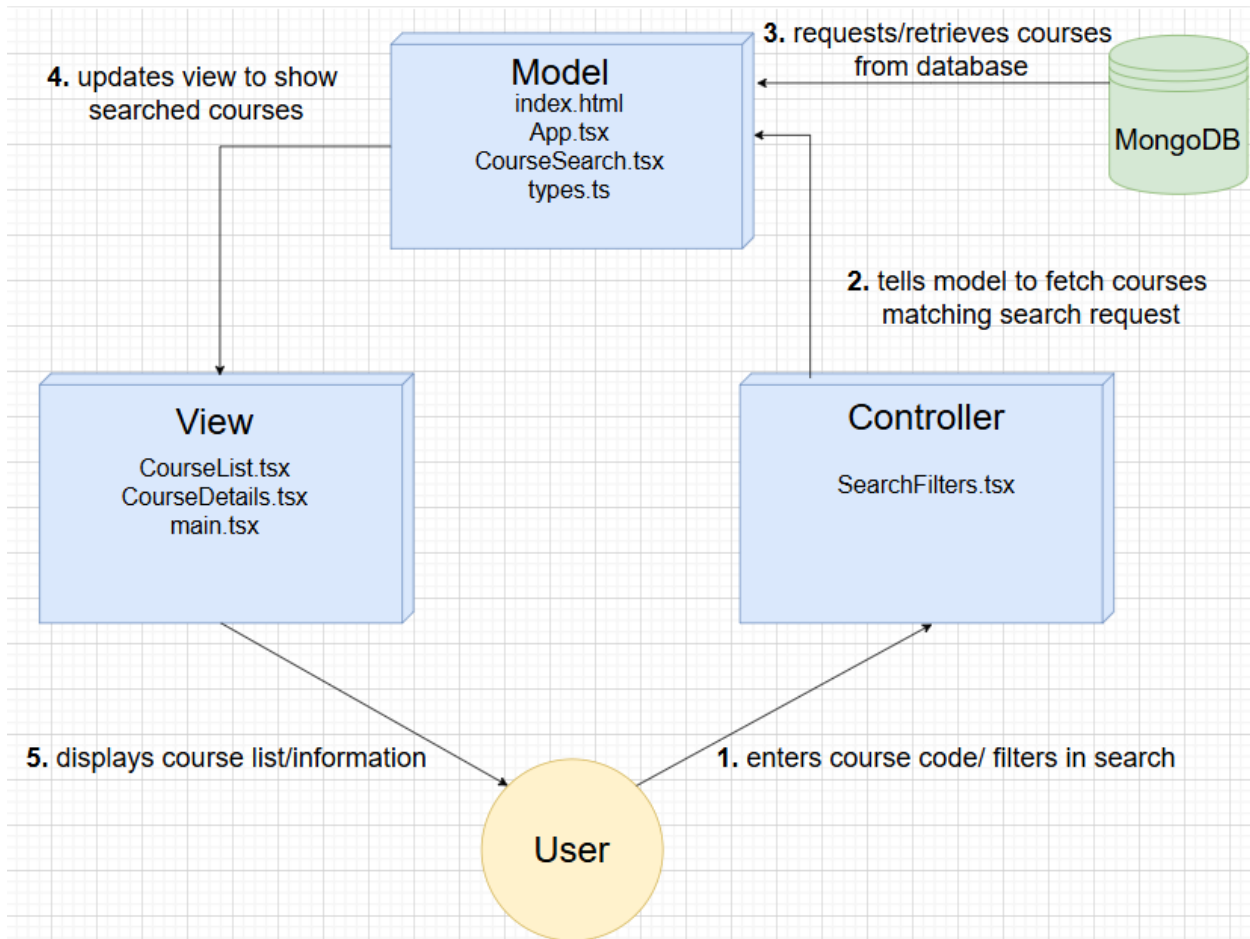
Class Name : CourseList.tsx	
Subclass: CourseList.css	
Responsibilities: <ul style="list-style-type: none"> • Displays a list of courses with each course shown in its own card 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva

Class Name : SearchFilters.tsx	
Subclass: CourseSearch.css	
Responsibilities: <ul style="list-style-type: none"> • Allows user to search for a course using an input field • Allows user to filter search results by course level, departments and its number of credits • Prompts user to apply their search filters with a button 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva

Class Name : types.ts	
Responsibilities: <ul style="list-style-type: none"> • Contains an interface for courses with attributes like course name, description, etc • Contains an interface for Professors with attributes like their teaching quality, difficulty, etc • Course interface contains Professor object as an attribute for professors teaching that specific course 	Collaborators: <ul style="list-style-type: none"> • Julia Da Silva

Class Name : index.html	
Responsibilities: <ul style="list-style-type: none"> acts as a container for the website and executes the script main.tsx 	Collaborators: <ul style="list-style-type: none"> Julia Da Silva Gurjap Hajra

Class Name : config.env	
Responsibilities: <ul style="list-style-type: none"> Holds connection string for the MongoDB database 	Collaborators: <ul style="list-style-type: none"> Gurjap Hajra



1. User can filter the list of courses shown by department, course level, and number of credits. On top of that they are also able to use the input field to search for courses by name or by its professors.

2. The controller sends the values from the search filters to the model.

3. The model will make a request to MongoDB to fetch all the courses from the database. From there it will filter out the courses corresponding to the received search filters.

4. The model will update the view so that it displays the correct courses to the user.

5. The user can scroll through the list of courses shown. When selecting a course the view updates to show more specific details about that course, including description, professors, etc.