

System Design Document

Steadfast Solutions

October 8, 2021

Contents

1	CRC Cards	3
1.1	App	3
1.2	SignUp	3
1.3	SignIn	3
1.4	SignInCallback	3
1.5	AuthProvider	3
1.6	AlertMessage	4
1.7	Navbar	4
1.8	AlertMessage	4
1.9	PillButton	4
1.10	Spacer	4
1.11	SubmitButton	5
1.12	TextInput	5
1.13	Joblisting	5
1.14	JobseekerProfile	5
1.15	User	5
1.16	SearchBar	6
1.17	Search	6
1.18	Resume API	6
1.19	Job Seeker	6
1.20	ElevatorPitch	6
1.21	Profile	7
1.22	Skills	7
1.23	Summary	7
1.24	WorkExperiences	7
1.25	Section	7
1.26	ProfilePage	7
1.27	Job Listing UI	8
1.28	Job	8
1.29	Job Listing API	8
1.30	Recruiter Profile	8
1.31	Application UI	8

1.32 Application Model	8
1.33 Applied Jobs UI	9
2 Software Architecture Diagram	10

1 CRC Cards

1.1 App

CRC card:

Parent class: N/A

Subclasses: SignInCallback, SignIn, SignUp, Search, Welcome

Collaborators: Navbar

Responsibilities:

Handles routing to the requisite subclasses, which receive routing information from App, and creates base pages.

1.2 SignUp

CRC card:

Parent class: App

Subclasses: N/A

Collaborators: AuthProvider, AlertMessage, PillButton, Spacer, SubmitButton, TextInput

Responsibilities:

Signs up users who provide their information.

1.3 SignIn

CRC card:

Parent class: App

Subclasses: N/A

Collaborators: AuthProvider, AlertMessage, PillButton, Spacer, SubmitButton, TextInput

Responsibilities:

Signs in users who bear correct credentials.

1.4 SignInCallback

CRC card:

Parent class: App

Subclasses: N/A

Collaborators: AuthProvider

Responsibilities:

Signs in users who are returning from an external authentication provider.

1.5 AuthProvider

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: React.Context

Responsibilities:

Provides authentication interface for other classes.

1.6 AlertMessage

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Provides alert messages for the UI.

1.7 Navbar

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: AuthProvider

Responsibilities:

Provides navigation bar component for UI.

1.8 AlertMessage

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Provides alert messages for the UI.

1.9 PillButton

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Provides a standard button for UI.

1.10 Spacer

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Provides a standard spacing utility for UI.

1.11 SubmitButton

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Renders a submission button with multiple states (active, pending, success).

1.12 TextInput

CRC card:

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Provides a styled text input for UI.

1.13 Joblisting

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Stores information data about individual job listings.

1.14 JobseekerProfile

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Stores information data about individual user profiles that are job seekers.

1.15 User

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Stores login and authentication details of all users.

1.16 SearchBar

Parent class: N/A

Subclasses: N/A

Collaborators: JobseekerProfile, AuthContext

Responsibilities:

1. Sends parameters to current page about location and keywords.
2. Send location and keyword parameters based on current user profile.

1.17 Search

Parent class: App

Subclasses: N/A

Collaborators: JobseekerProfile, AuthContext, SearchBar, Job-listing

Responsibilities:

1. View list from filtered search results of all job listings.

1.18 Resume API

Parent class: resumeController.js

Subclasses: N/A

Collaborators: JobseekerProfile

Responsibilities:

Allows users to upload a PDF resume in their profile.

1.19 Job Seeker

Parent class: N/A

Subclasses: N/A

Collaborators: N/A

Responsibilities:

An object representing a job seeker, which stores identifying information and job specific data like skills and education. It contains a field that ties itself to a User object.

1.20 ElevatorPitch

Parent Class: N/A

Collaborators: Section

Responsibilities:

When clicked, the user's elevator pitch will play in a popup.

1.21 Profile

Parent Class: N/A

Collaborators: Section

Responsibilities:

Contains contact information about the user like name, phone number, etc.

1.22 Skills

Parent Class: N/A

Collaborators: Section

Responsibilities:

Displays any skills the user wants to showcase. Users can also delete and add skills here.

1.23 Summary

Parent Class: N/A

Collaborators: Section

Responsibilities:

Quick summary of a user. Users can also update their profile summary as necessary.

1.24 WorkExperiences

Parent Class: N/A

Collaborators: Section

Responsibilities:

Displays any work experiences the user wants to showcase. Users can also delete and add work experiences here.

1.25 Section

Parent Class: N/A

Collaborators: None

Responsibilities:

Section template for profile page.

1.26 ProfilePage

Parent Class: N/A

Collaborators: Profile, ElevatorPitch, Summary, Skills, WorkExperiences

Responsibilities:

Displays profile page.

1.27 Job Listing UI

Parent Class: Search

Collaborators: Joblisting, Job

Responsibilities:

Displays information about listings, displays a second card that shows more details depending on state.

1.28 Job

Parent Class: Job Listing UI

Collaborators: Joblisting

Responsibilities: Displays information about specific listing, change state in Parent when clicked on.

1.29 Job Listing API

Parent class: jobListingController.js

Subclasses: N/A

Collaborators: server.js

Responsibilities:

Allows recruiters to create job listings. Job details from create jobs form are parsed and passed into the various methods in the API.

1.30 Recruiter Profile

Parent class: RecruiterProfile.js

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Mongoose model to store recruiter profile information.

1.31 Application UI

Parent class: N/A

Subclasses: N/A

Collaborators: JobListing UI

Responsibilities:

Allow users to apply to jobs

1.32 Application Model

Parent class: Application.js

Subclasses: N/A

Collaborators: N/A

Responsibilities:

Mongoose model to store application information

1.33 Applied Jobs UI

Parent class: N/A

Subclasses: N/A

Collaborators: Job Listing UI

Responsibilities:

Display to the user what jobs they have already applied to

2 Software Architecture Diagram

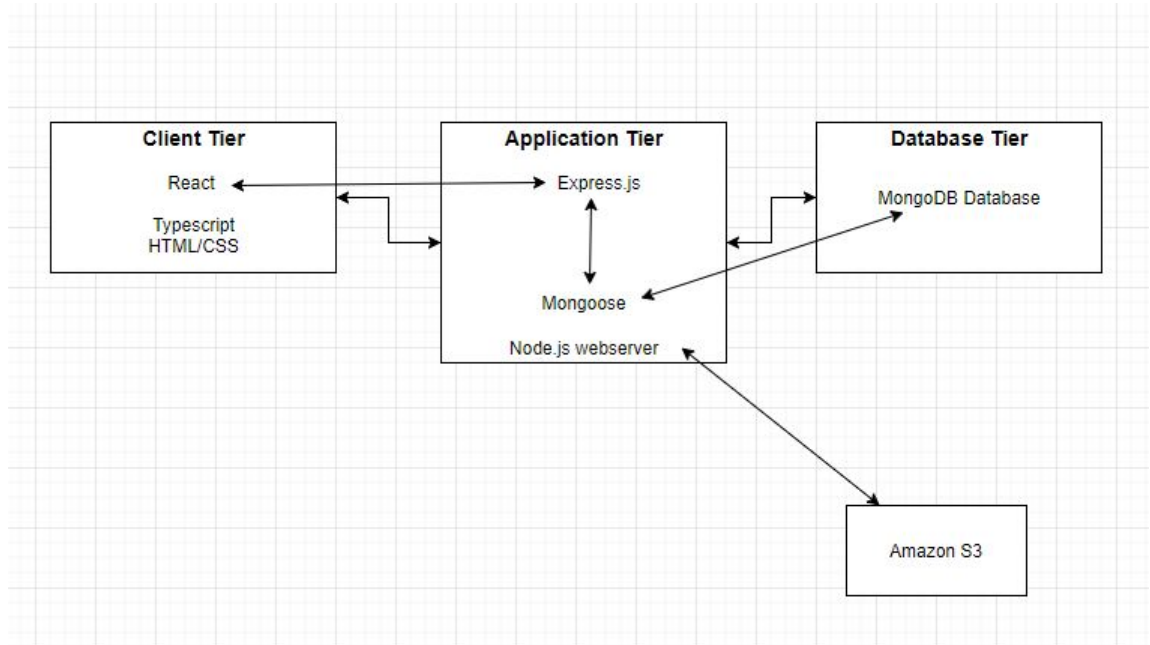


Figure 1: The EasyApply web app uses a 3-tier architecture. The tiers are Client, Server, and Database
Reference: <https://levelup.gitconnected.com/a-complete-guide-build-a-scalable-3-tier-architecture-with-mern-stack-es6-ca129d7df805>