**Assignments (SE I/SE II):**

**Assignment 1**: Online test application

**Description:** An online skill test application which should provide candidates to register with basic information, choose skill test level and start a single/multiple choice answer test with time limit.

**App details:**

1. Registration should be the first step to start the test.
2. Candidates can view exam information, agreement check and choose skill level (1, 2 and 3).
3. A start button should start the test with specific time.
4. Candidates can view one question per page and can navigate next/previous.
5. Configurable JSON files for different skill level questions and answer options
6. Question may be a single or multiple choice.
7. Timers should be displayed on screen.
8. Summary of question attempts should be visible on screen.
9. It should auto submit on finish of specified time.
10. On submission of the test, create a JSON file with candidate registration details and answers.
11. Ensure after the test submit candidates should not be able to navigate back to question pages.

**Assignment 2:** Champions dashboard

**Description:** Create an application where users can view/search list of champions on home page and provide actions on displayed result to add/remove champions into user’s watch list. A watch list page should display selected champions with full details. Application should interact with an API in order to provide list and search champions.

**API Details:**

1. Login to <https://pandascore.co> to get your access token for using API.
2. Endpoint - [https://api.pandascore.co/lol/champions?page[number]=2&page[size]=10&token=](https://api.pandascore.co/lol/champions?page%5bnumber%5d=2&page%5bsize%5d=10&token=)<your token> to get the list of champions.
3. Endpoint - [https://api.pandascore.co/lol/champions?search[name]=<search keyword>&token=](https://api.pandascore.co/lol/champions?search%5bname%5d=twi&token=puqITtWgHHS9L6eg1UH1ADPlTykKMGXfi-EPqTISzX0Eg1PM1Nw)<your token> to search for a champion.

**App details:**

1. A home page where user can view default list of champions in grid with pagination
2. Home page should also provide an autocomplete search box to find a matching list of champions.
3. The results grid column should be sortable asc/desc (without API call).
4. Provide action buttons in each row of the grid to add/remove champions to user’s watch list.
5. Display summary/count of added champions into watch list icon on the home page. (Can be on page left or top panel).
6. On click of the champion name, users can view the details of a champion `(modal/popup screen).
7. Users can navigate to the watch list page on the click of the home page’s watch list icon, where he/she can view all selected champions with details one by one with image (may use card).
8. Users should also be able to remove champions from the watch list page.

**Assignment 3**: Crypto currencies watch

**Description:** An application where users can view/analyze crypto currency market trends and history. Users can view all available crypto currencies and choose the currency of his/her choice to further check trends in detail through the time series chart.

**API Details:**

1. <https://docs.coinranking.com/public#introduction>
2. GET<https://api.coinranking.com/v1/public/coins>
3. GET<https://api.coinranking.com/v1/public/coin/:coin_id>
4. GET<https://api.coinranking.com/v1/public/coin/:coin_id/history/:timeframe>

**App details:**

1. A home page where users can view all available cryptocurrencies with pagination.
2. Home page crypto currencies list should include currency name, logo, market capital and %change in 24 hours.
3. Users can change the base currency of their choice to view prices on that currency.
4. On selection of any crypto currency user should navigate to that crypto currency details page.
5. Details page should display detailed information of that crypto currency with a time series chart (default 24 hours).
6. Users should be able to change chart plot time - 24h, 7d, 30d, 1y, 5y.
7. Chart must display price points with date time hovering across the chart.
8. Users should be able to change base currency from any screen of application.

**Assignment 4:** Find makeup product

**Description:** An application where users can view/search/filter makeup products. View product list with filter narrow down options like category, brand, price range and tags. View details of product with image and different color options available with all important details like price, brand, description etc.

**API Details:**

1. [http://makeup-api.herokuapp.com](http://makeup-api.herokuapp.com/)

**App details:**

1. A home page with autocomplete text search by product type, category and brand.
2. Manage a JSON file inside app for product type, category, brand and tags to feed autocomplete search suggestions and filter options.
3. A screen left panel to apply further filters on search results. Options include brand, tag, price range and rating range.
4. Display product summary (card view) as search results with basic product information with thumbnail image.
5. Search results should be displayed with per page limit and pagination.
6. Product selection should open product details in a new browser tab.
7. A Detailed product page should display all information including available color options of product with a website link to that product.
8. Maintain product search history by user. It should browse products search back.

**Assignments (SSE/ML):**

**Assignment 1:** Magic Cards dashboard

**Description:** Create an application where users can view/search list of cards

on the home page and provide actions to view details of each card. Application should interact with an API in order to provide list and search cards.

**API Details:**

1. Endpoint - https://api.magicthegathering.io/v1/cards to get the list of cards.
2. Endpoint - https://api.magicthegathering.io/v1/cards/:<id> to get detail for a card.

**App details:**

1. A home page where user can view default list of cards in grid with pagination
2. Home page top bar should provide a search box to find a matching list of cards from API.
3. Left panel of the Home page should also display filters to be applied on search cards (make API calls to apply filters).
   1. Colours – It should be multi select dropdown
   2. Supertypes – Dropdown with options
   3. Rarity – Dropdown with options
   4. Page size – Dropdown with option (10, 25, 50)
4. The results grid column should be sorted based on “Layout” field (flip, normal, split and others) and pagination.
5. Site top bar should provide theme toggle options (Light and Dark) – using React Context.
6. Site top bar should also provide dropdowns with different language options (English, French, Chinese Simplified) to be applied on site – using React Context.
7. Display card detail in separate page on click of any card name from grid.
8. Display card distinctly on the card detail page according to “Rarity” field value.
9. Navigating back from Card detail page to Home page, search results/filters should be retained.
10. Use SASS/responsive library (bootstrap or any other) – Site pages should run smoothly over Mobile and Desktop resolutions.
11. Use Redux – API call and state.