

Welcome To Scapic

Scapic lets people create, share and explore immersive experiences. We're a platform for building Virtual/Augmented/ Mixed Reality content easily. You'll be at the center of the VR / AR ecosystem.

Create: Scapic's drag-and-drop editor makes it trivially simple to build immersive experiences from scratch in minutes. We call these experiences as Scapes

Share: Scapic is entirely built on the cloud. You can simply preview and share your Scapes as a URL. You can view a Scape right from your web browser and allow others to instantly jump into your experiences

Explore: Learn, suspend, tread, be. Be immersed in spheres of visuals. Move through content that surrounds you in the real world, or move through completely new virtual worlds

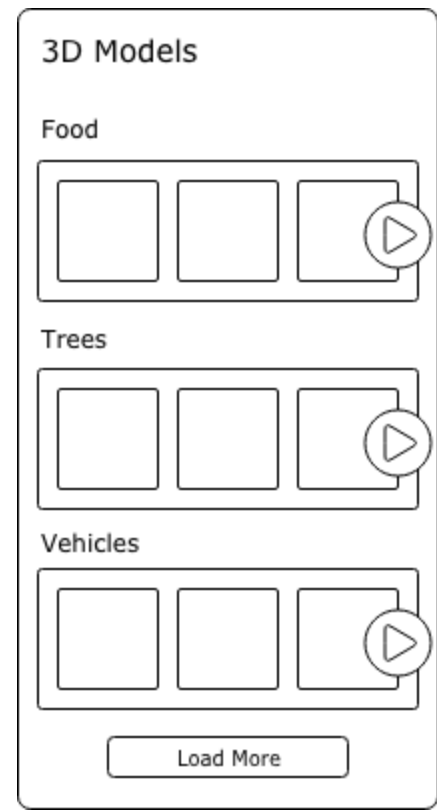
This means you can design an Immersive Experience (VR/AR/MR) in the same amount of time it takes to make a presentation in Powerpoint or pen a blog. People can now create experiences in hours – not days, weeks or months. We believe in the Open Internet, and envision a future where the web is an immersive one

As one of the fastest growing VR, AR startups in Asia, Scapic is a firm on a mission to build the immersive web of tomorrow. This mission runs through the heart of the organization—it's reflected in our work, our values and in our culture. At Scapic, we strive to create a supportive and inclusive environment where collaboration is encouraged, and learning is shared freely. We have the dynamism and short chain of command of a start-up, with a culture that is fast-paced and risk-taking. We achieve more as a team than alone and are always looking for great people who will strengthen our team and drive our mission forwards.

Code Sprint

Requirement:

- Build out a page for a 3D object store, where the user can see 3D models for objects across various categories.
- The store front page consists of models sorted by categories. As user scrolls down, more categories should be loaded. The user should be able to scroll horizontally in a collection and load more models of that category
- The number of models in a category, loaded at a time should be enough to fill the screen of that particular size
- Deploy your project so that we can try out your assignment and see it working. **Don't host the code on github** but instead create a .zip archive and send it across on the same thread you received this on.



Frontend Activity

- In order to build out the front-end for this project, using **React + Redux**.
- If you're applying for a backend role, you can use other frameworks to implement the front-end of this project.
- **Clicking on the card of a model should preview it in 3D.** Use A-Frame / Three.js to show / render the 3D model.
- Devise a pagination system which you feel provides the best UX, whether its auto-pagination or button based.
- If you haven't worked with Node / DB and or if you're applying only for a Frontend role only, you are free to load the required data as a JSON locally from a file. Do make sure that you still maintain a valid pagination scheme.

Backend Activity

- Make endpoints to support the frontend using **nodeJS**.
- Design the database models required for this application, along with relevant relations. Use any relational DB. (Psst... we like PostgreSQL)
- Implement a valid pagination scheme for the loading of more categories and models.
- Take into consideration the performance of the relations and queries you implement.
- **If you're applying for a Backend first role**, head over the next page for another fun activity.

The data for the models is given in the JSON at

<https://s3.ap-south-1.amazonaws.com/scapic-others/json/models.json>

Backend Activity – Additional Scope

Linked Accounts

Along with the store assignment, please complete the following if you are applying for a Backend First Role.

1. Create a login / signup page. If the user doesn't exist, they should be able to sign up and if they do they should be able to log in.
2. On the initial sign up, generate a unique user id and send it to the client.
3. The options provided to login / signup should be
 - Email and password
 - OAuth Google login
 - OAuth Facebook login
4. After signup / login the user should be redirected to a profile page.
5. On the profile page the user should see field such as email, the unique uid and 3 buttons:
 - Link account to google
 - Link account to Facebook
 - Delete account
6. The email should be editable and the user should be able to do the same
7. If the user has signed up using any one of the social oauth methods, this means that the user has already linked that particular account and accordingly the buttons should switch to "Unlink account from google / facebook".
8. If the user clicks on the "**Link account button**", the social profile associated with the button should be linked to the current account. This means that if the user would log in using their linked account, they should be redirected to the same account with the same email and same uid shown used on initial sign up.
9. If the user clicks on the "**Unlink account button**", the social account associated with the button should be unlinked and the next time user tries to login using the unlinked social account, a new account should be created instead.
10. Once an account is linked, that account can't be used to register or link to any other account
11. If the user clicks on delete account, a soft delete should be done on the DB.
12. The user should be able to create new account using a deleted account's email.

Prerequisites:

- Language: JavaScript (Node.js).
- Database: MongoDB.
- Web server Framework: Express.
- Frontend: Can use anything (Extra points to use ReactJs).

Points to keep in mind:

- Use JavaScript ES6 standard.
- Indent your code properly.
- Try following the DRY (Don't Repeat Yourself) principle with max code reuse.
- Handle all ERRORS properly, and throw exceptions when necessary.
- Use **Mongoose** ODM.
- A Good understanding of Promises is a must.

This is an open ended assignment meaning that you're free to add in your own magic to show us your understanding and creativity. We would love to see how you approach the UX for the project and do know that by building this, you are stepping into the future of the immersive web.