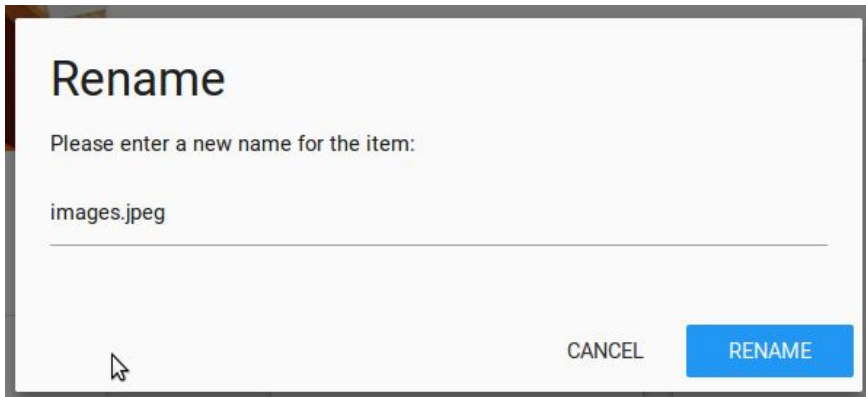


Sprint 1

1. File renaming

It was important to show the current filename when seeing the rename popup dialog. This value was inserted into the editable field for ease of use.

A screenshot of a 'Rename' dialog box. The title 'Rename' is at the top left. Below it, the text 'Please enter a new name for the item:' is displayed. Underneath, there is a text input field containing the text 'images.jpeg'. At the bottom right, there are two buttons: 'CANCEL' and 'RENAME'.

```
Template.ui_modals_renameFile.events({
  "click .mv-renameFile": (event, template) => {
    let newName = template.$(".mv-newFileName").val();
    Meteor.call("Files.methods.rename", {
      _id: template.data.file._id,
      newName: newName
    });
  }
});
```

```
export const renameFile = new ValidatedMethod({
  name: "Files.methods.rename",
  validate: new SimpleSchema({
    _id: {
      type: String
    },
    newName: {
      type: String
    }
  }).validator(),
  run: ({_id, newName}) => {
    Files.update(_id, {$set: {name: newName}});
  }
});
```

2. Project database schema

As Meteor projects are natively supported by the nosql database MongoDB, we decided to use it. Although this uses mostly boilerplate database structure for the libraries we used, it was also easily extendable for us to include custom information to support our application.

```
{
  "_id": "7AMdbr7ftHRGhDvri",
  "name": "images.jpeg",
  "extension": "jpeg",
  "path": "/home/teatimer/MediaVault/fileStore/7AMdbr7ftHRGhDvri.jpeg",
  "meta": {},
  "type": "image/jpeg",
  "size": 9705,
  "versions": {
    "original": {
      "path": "/home/teatimer/MediaVault/fileStore/7AMdbr7ftHRGhDvri.jpeg",
      "size": 9705,
      "type": "image/jpeg",
      "extension": "jpeg"
    }
  },
  "isVideo": false,
  "isAudio": false,
  "isImage": true,
  "isText": false,
  "isJSON": false,
  "_prefix": "ed17d58b5d798b333f85b167562a983d368a68a95aea078760f0c9aecc653af4",
  "_storagePath": "/home/teatimer/MediaVault/fileStore",
  "_downloadRoute": "/cdn/storage",
  "_collectionName": "files",
  "userId": "gAWrrY99BHvzaBNDP",
  "public": false,
  "previewURL": "/previews/7AMdbr7ftHRGhDvri_preview.png"
}
```

```
{
  "_id": "gAWrrY99BHvzaBNdp",
  "createdAt": ISODate("2017-06-02T01:26:43.625Z"),
  "services": {
    "google": {
      "accessToken": "ya29.GlxeBBvvXlymHP_jlkppwxLkg3v_Xioz5szsYZ8a9CuiZgjxnEtuINom9QbsTTtRMpAjSmqag",
      "idToken": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjMyZjMwYjM1Njc1NjNiYjUxYmY0ODRjODNiMDVjMjM1NGIyYTkwOT",
      "expiresAt": 1496454386428,
      "scope": ["https://www.googleapis.com/auth/userinfo.email", "https://www.googleapis.com/auth/u",
      "id": "108464623586973457967",
      "email": "ashleygeorgeclarke@gmail.com",
      "verified_email": true,
      "name": "Ash Clarke",
      "given_name": "Ash",
      "family_name": "Clarke",
      "picture": "https://lh6.googleusercontent.com/-dbJ0rfiSPsQ/AAAAAAAAAI/AAAAAAAAAR6U/dkE3k1SIthY",
      "locale": "en-GB",
      "gender": "male"
    },
    "resume": {
      "loginTokens": [{
        "when": ISODate("2017-06-02T01:26:43.643Z"),
        "hashedToken": "bquV2CwoAD9u8YJTOYnz66ne3LlMNHptQJVGuC9RgHc="
      }, {
        "when": ISODate("2017-06-03T00:46:28.533Z"),
        "hashedToken": "NqGFu57Tf+Xsm7J4ehXHiJUW/ZNFyecBTfUEgk52w90="
      }
    ]
  },
  "profile": {
    "name": "Ash Clarke"
  }
}
```

3. File deletion

As shown below, the artefact demonstrates the

```
export const deleteFile = new ValidatedMethod({
  name: "Files.methods.delete",
  validate: new SimpleSchema({
    _id: {
      type: String
    }
  }).validator(),
  run: ({_id}) => {
    try {
      Files.remove(_id);
      Tags.remove({fileID: _id});
    } catch (ex){}
  }
});
```

```

Template.ui_modals_deleteFile.events({
  "click .deleteFile": function(event, template){

    callback = () => {
      that = this;
      return function() {
        toastr.success("Deleted "+that.file.name, "Deleted file.");
        Meteor.call("Files.methods.delete", {_id: that.file._id});
      }
    };

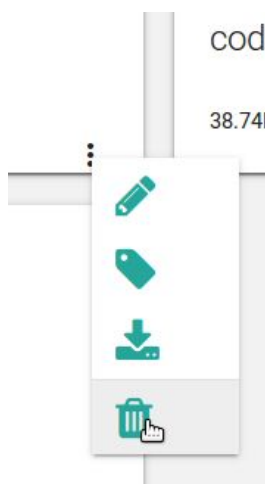
    setTimeout(callback(), 200);
  }
});

```

```

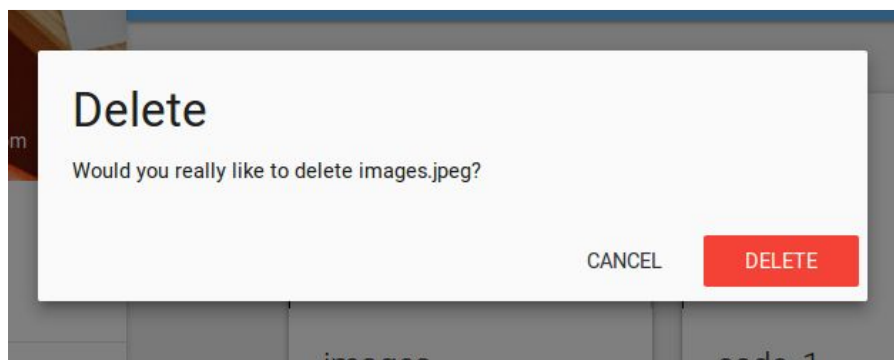
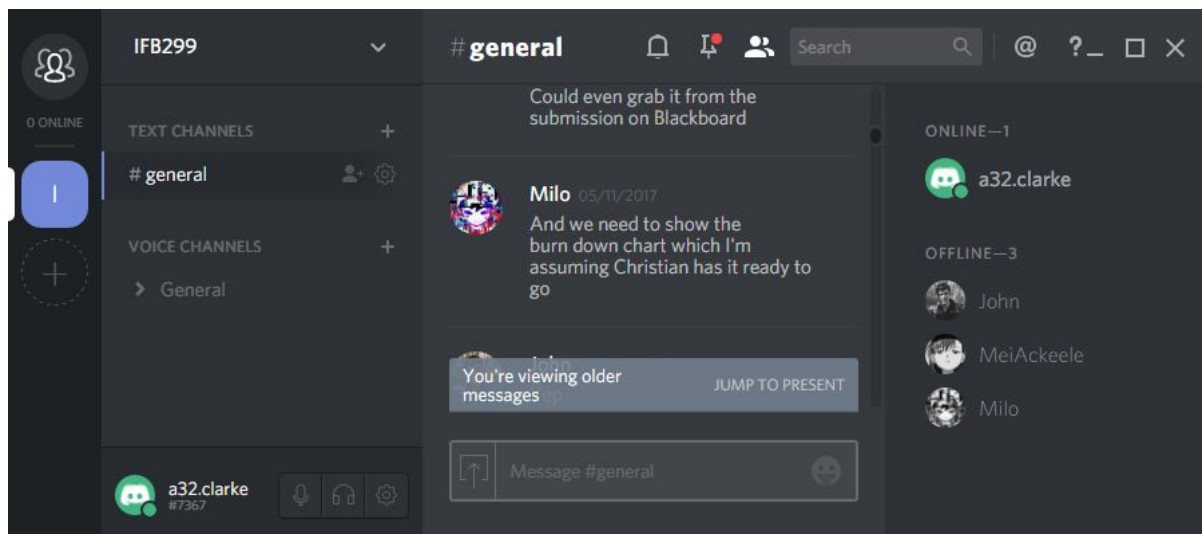
<template name="ui_modals_deleteFile">
  {{#ui_modals_modalTemplate modalID=modalID}}
    <div class="modal-content">
      <h4>Delete</h4>
      Would you really like to delete {{file.name}}?
    </div>
    <div class="modal-footer">
      <a href="#" class="deleteFile modal-action modal-close waves-effect waves-
      <a href="#" class="modal-action modal-close waves-effect waves-
    </div>
    {{/ui_modals_modalTemplate}}
  </template>

```



4. Setting up the Discord server for team communication

As we needed an effective method for team meeting and communication, we decided to use Discord. We used it to pin important information that the team needs for the project such as server information and links to supporting documentation.



Sprint 2

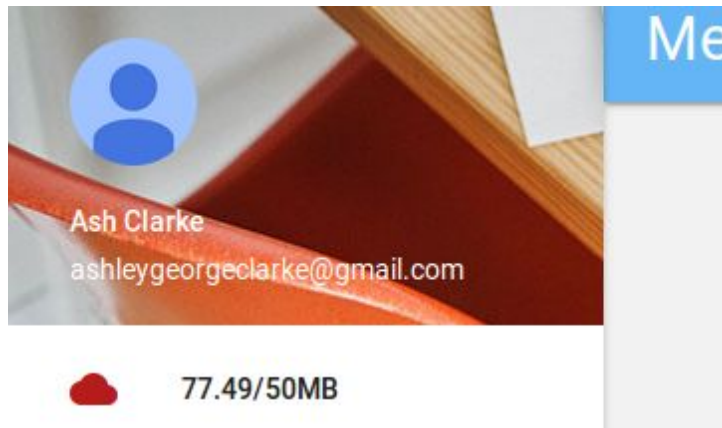
1. Hardcoded domain bug

A bug had appeared where some sections of the code had mistakenly been hardcoded using the production server domain ie. eante.com.au:3030. This meant that using the code on a different domain or working locally would not be able to run the install.

```
let previewName = fileObj._id+"_preview.png";  
let previewPath = "/var/www/eantecomau/previews/" + previewName;  
let previewURL = 'eante.com.au/previews/'+previewName;
```

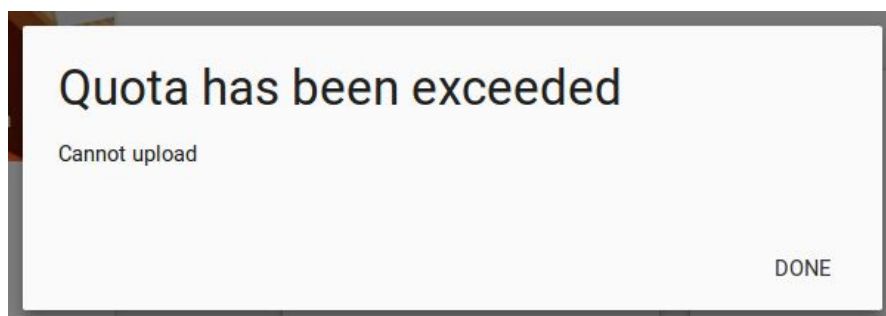
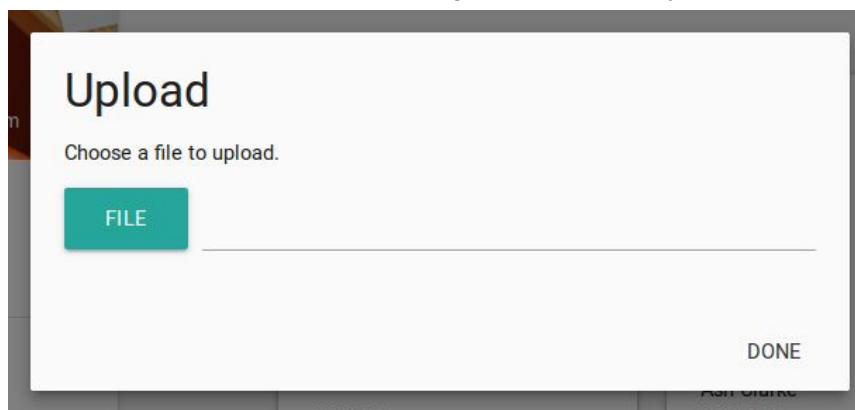
2. Download quota display

As we'd want a clear and visual representation of the current usage quota of the signed-in user, we display the current usage and the allowed quota. As shown below, the cloud turns red when the quota has been exceeded.



3. Exceeded quota upload prevention

Prevention of user uploading file if quota has been exceeded. This was deemed as a 'must have' in our user stories. As shown below, the different dialogues that are displayed when a user tries to upload a file depending on whether they have exceeded their quota or not.



4. Developer dependencies installation listings

A list of project dependencies was maintained between developers so that with each new installation either locally or on a server, we were able to install any newly required software easily. This maintained project efficiency especially when working collaboratively via git.


```
{  
  "name": "MediaVault",  
  "private": true,  
  "scripts": {  
    "start": "meteor run"  
  },  
  "dependencies": {  
    "babel-runtime": "^6.18.0",  
    "download-file": "^0.1.5",  
    "hammerjs": "^2.0.8",  
    "lodash": "^4.17.4",  
    "meteor-node-stubs": "~0.2.3",  
    "raf": "^3.3.0",  
    "rebound": "0.0.13",  
    "sister": "^3.0.0",  
    "vendor-prefix": "^0.1.0"  
  }  
}
```