



## **Remote Assessment and Proctoring using Intelligent Devices (RAPID): Raspberry Pi 5 Handover Document**

Chun Jia Jun Louis, 2201618; Denzel Low E Loong, 2202347; Nadhirah Binti Ayub Khan, 2201689;  
Sia Hong Liang, 2201493; Wyvern Kiang Teng Wei, 2200577

Name of the Industry Supervisor: **Arthur Loo Wee Yeong**; Name of the Company: **Singapore Institute of Technology**

Name of the SIT Academic Supervisor(s): **Muhammed Fauzi Bin Abbas** and **Goh Wei Han**

<b>Table of Contents</b>	
<b>Raspberry Pi 5 Credentials</b>	<b>3</b>
<b>Docker Container Overview</b>	<b>3</b>
1. PHP Container (php_app)	3
2. Database Container (mongodb)	3
3. Nginx Container (nginx)	4
4. Shared Resources	4
<b>Files Locations</b>	<b>4</b>
<b>How to Run Docker Containers</b>	<b>6</b>
<b>How to Manage MongoDB</b>	<b>7</b>
<b>How to Establish a Domain</b>	<b>8</b>
To establish a domain for the RAPID project using Duck DNS:	8

# Raspberry Pi 5 Credentials

Username: rapiduser

Password: ITP@us3r

## Docker Container Overview

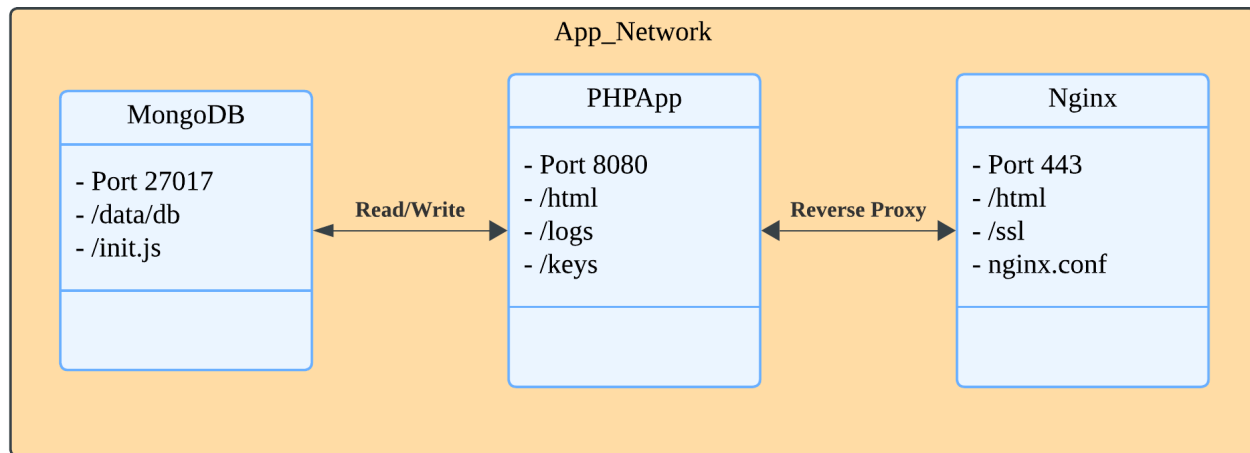


Fig 1. Brief Overview of Docker Containers

### 1. PHP Container (php\_app)

**Purpose:** Runs the PHP application.

**Build:** Custom image from a Dockerfile (`Dockerfile-php`).

**Mounted Volumes:**

`./html` → `/var/www/html`: Application files.

`./logs` → `/var/logs/myapp`: Application logs.

`./keys` → `/var/www/keys`: Application-specific keys.

`./ssl/openssl_myapp.cnf` → `/etc/ssl/openssl_myapp.cnf`: SSL configuration.

**Ports:** Exposes port 80 internally, mapped to host port 8080.

**Depends on:** db (ensures the database is ready first).

**Environment Variables:** Loaded from `.env` file.

**Network:** Part of the `App_Network`.

### 2. Database Container (mongodb)

**Purpose:** Provides a MongoDB database for the application.

**Image:** `mongo:latest` (official MongoDB image).

**Environment Variables:**

Credentials and database name loaded from `.env` file.

**Mounted Volumes:**

`./mongo-init.js` → `/docker-entrypoint-initdb.d/mongo-init.js` (read-only): Initialization script.

`./media/rapiduser/Lexar/db_data` → `/data/db`: Persistent data storage.

**Ports:** Exposes MongoDB's default port (27017).

**Network:** Part of the [App\\_Network](#).

**Restart Policy:** always (ensures it restarts automatically).

### 3. Nginx Container (nginx)

**Purpose:** Acts as a reverse proxy and serves HTTPS requests.

**Image:** nginx:latest (official Nginx image).

**Mounted Volumes:**

[./html](#) → [/var/www/html](#): Serves the same files as the PHP container.

[./keys](#) → [/etc/nginx/ssl](#): SSL keys for HTTPS.

[./nginx.conf](#) → [/etc/nginx/conf.d/default.conf](#): Custom Nginx configuration.

**Ports:** Exposes port 443 for HTTPS.

**Depends on:** php (ensures the PHP container is ready first).

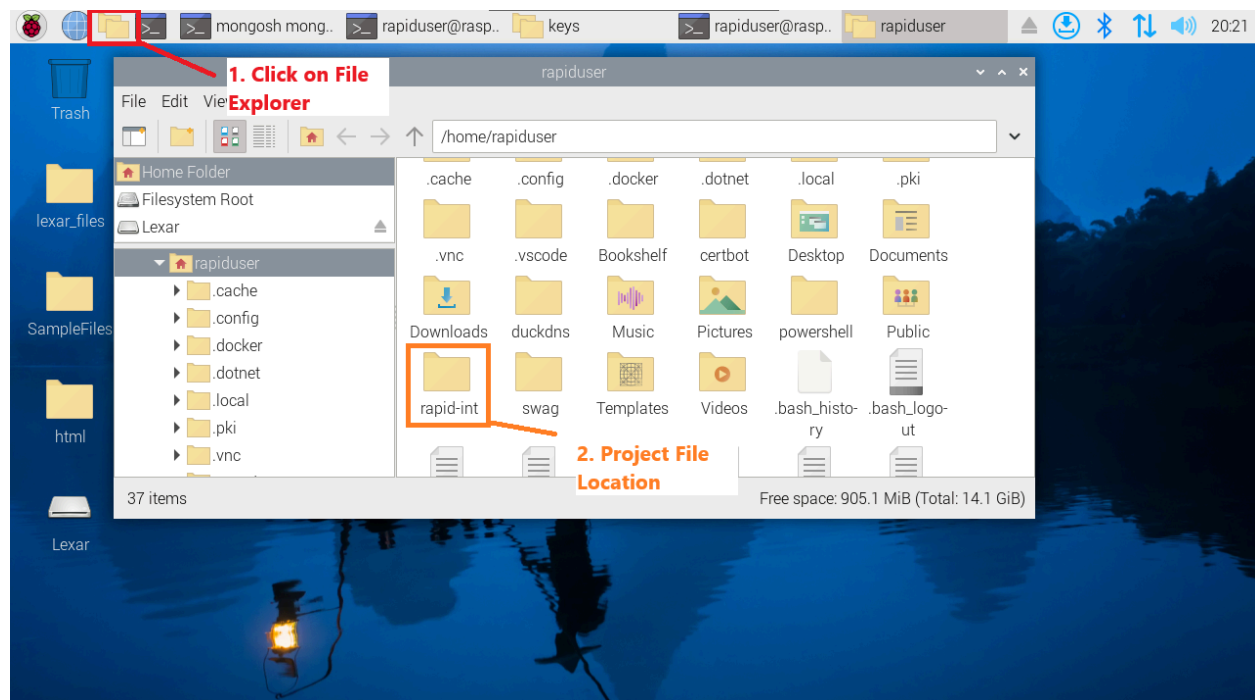
**Network:** Part of the [App\\_Network](#).

### 4. Shared Resources

**Volumes:** db\_data: Persistent storage for MongoDB (local driver).

**Network:** App\_Network: Shared bridge network connecting all containers.

## Files Locations

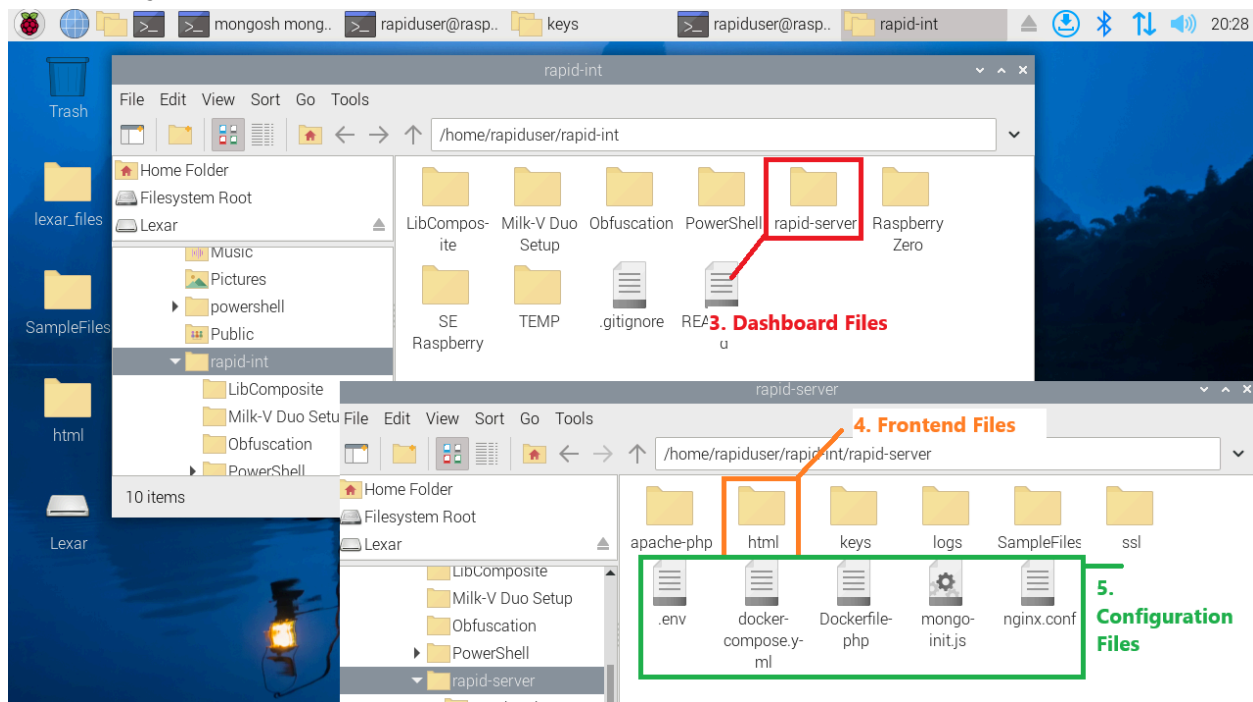


Screenshot 1. Navigating File Explorer to Rapid-int

#### 1. Open File Explorer

- a. To start navigating and managing the RAPID project, click on the file explorer on the Raspberry Pi 5 OS Navigation Bar.

## 2. Project File Location: Double-click on **Rapid-int** to access its contents.



Screenshot 2. Navigating Rapid-int to Rapid-server

## 3. Dashboard Files

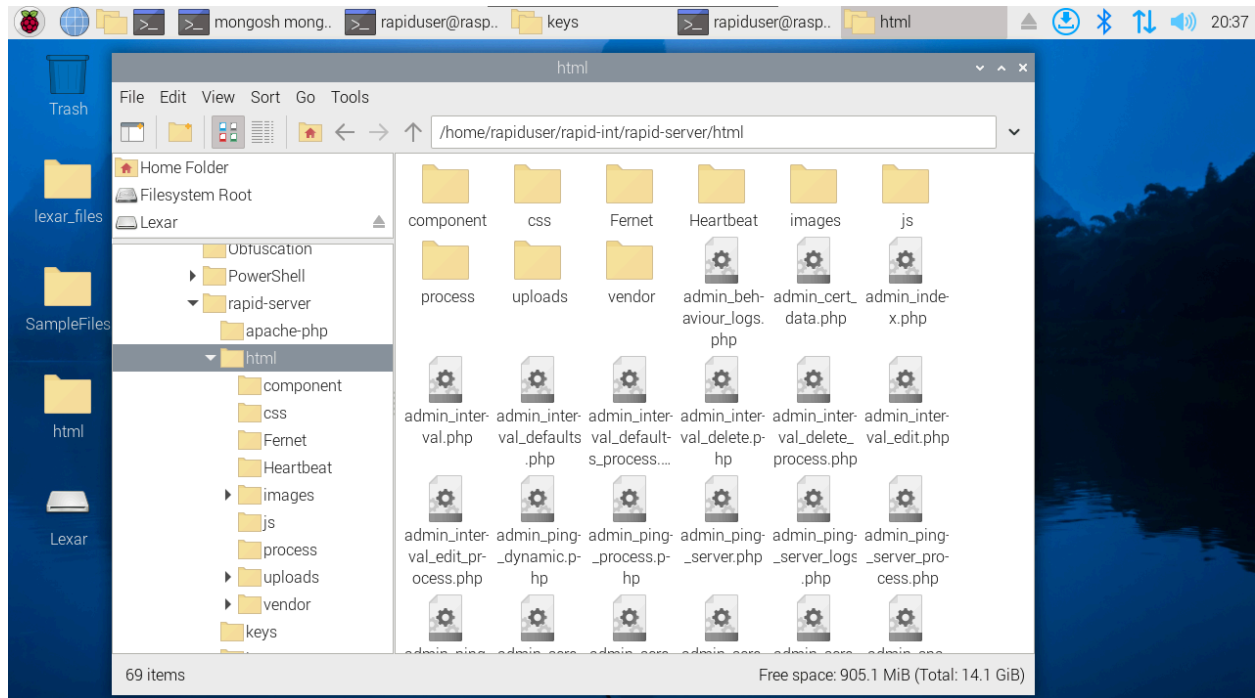
- a. Double-click on **Rapid-server** to access its contents.

## 4. Frontend Files

- a. This folder contains all of the front-end dashboard functionality. Further details on this are in Screenshot 3.

## 5. Configuration Files

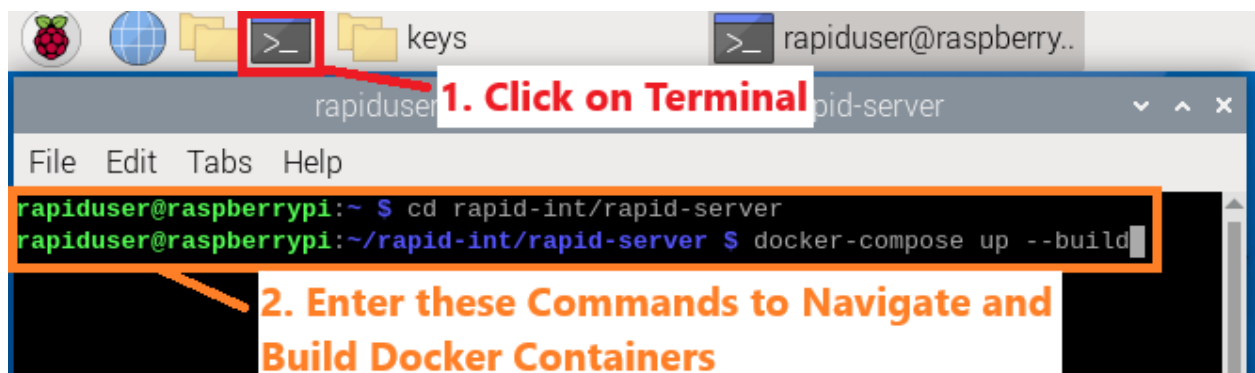
- a. **.env**: MongoDB Login Credentials
- b. **Docker-compose.yml**: Docker Containers Configurations
- c. **Dockerfile.php**: Docker Dependency Configurations
- d. **mongo-init.js**: MongoDB Collections Initialisation Configurations
- e. **Nginx.conf**: Reverse Proxy for Https Protocols to enable security measures



Screenshot 3. Html Dashboard Contents

- **components:** Shared UI components throughout all Pages(eg. Side Nav bar)
- **css:** Cascading Style Sheets Folder
- **images:** Not to be confused with Students' Snapshots and Screenshots, this folder contains images used on the Dashboard visuals
- **js:** Javascripts that contain Dynamic Database/Frontend UI retrieval and manipulation
- **process:** Contains all files that manipulate Database Data

## How to Run Docker Containers



Screenshot 4. Docker Compose Steps

1. Click on Terminal
2. Commands to Navigate and Build Docker Containers

- a. `cd rapid-int/rapid-server`: Navigate into Docker-compose.yml location
- b. `docker-compose up --build` ; **Important** MUST Include `--build` so that Docker will install the dependency needed into the respective containers.

```

rapiduser@raspberrypi:~/rapid-int/rapid-server
rapiduser@raspberrypi:~/rapid-int/rapid-server $ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
cf62ea39b875   nginx:latest   "/docker-entrypoint..." 22 seconds ago Up 19 seconds 80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp  nginx
3b7135750e57   rapid-server-php "docker-php-entrypoint..." 36 seconds ago Up 20 seconds 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp  php_app
a77d6f335e7b   mongo:latest   "docker-entrypoint.s..." 42 seconds ago Up 20 seconds 0.0.0.0:27017->27017/tcp, :::27017->27017/tcp  mongodb

```

**3. Run docker ps**  
**Ensure 3 Containers are Created**

Screenshot 5. Docker Compose Checker

3. Docker Container Verification: Run `docker ps` to ensure that nginx, php\_app, and mongodb containers are created.

## How to Manage MongoDB

```

rapiduser@raspberrypi:~/rapid-int/rapid-server
rapiduser@raspberrypi:~/rapid-int/rapid-server $ docker-compose exec db bash -c 'mongosh -u myuser -p mypassword --authenticationDatabase admin'
current MongoDB Log ID: 67498c251300b93a1e80b51a
Connecting to:
mongodb://<credentials>@127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&authSource=admin&appName=mongosh+2.2.6
Using MongoDB:
7.0.11
Using Mongosh:
2.2.6

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

-----
The server generated these startup warnings when booting
2024-11-22T21:31:24.836+08:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
-----

test> use rapid
switched to db rapid
rapid> show collections
behaviour_logs
cert_data
defaults
intervals
ping
Processes
proctoring
Screenshots
session_sequence
SessionInvigilators
Sessions
Snapshots
Students
Users
rapid>

```

**1. Enter these Commands to Navigate and Bash command into MongoDB Container**

**1a. Upon Successful Bash command into MongoDB Container, Success Return Message will be displayed**

**2. Switch to rapid database to access respective project collections**

**2a. Database Returns all the collections upon querying**

Screenshot 6. MongoDB Bash Command

1. Commands to Navigate and Bash command into MongoDB Container
  - a. `cd rapid-int/rapid-server`: Navigate into Rapid Project location
  - b. `docker-compose exec db bash -c 'mongosh -u myuser -p mypassword --authenticationDatabase admin'`: Bash command to manage MongoDB
- 1a. Return Message from Successful Bash: Contain MongoDB Version and Log ID

2. Switching to rapid database to access its collections
  - a. **use rapid**: Switch to rapid database to access its collections
  - b. **show collections**: Shows all collections in rapid Database
- 2a. Return Collections from querying: Containing all Collections from rapid database

Additional Documentation for bash commands:

<https://www.mongodb.com/docs/mongodb-shell/run-commands/>

## How to Establish a Domain

The screenshot shows the Duck DNS website interface. At the top, there's a navigation bar with links: spec, about, why, install, faqs, and logout. The user is logged in. The main header features a yellow duck logo and the text "Duck DNS". Below this, account details are displayed: account (redacted), type (free), token (redacted), token generated (1 month ago), and created date (20 Oct 2024, 07:26:11).

The "domains" section shows a list of domains. The first domain is "rapid-int". It has a "current ip" field (redacted) with an "update ip" button, an "ipv6" field (redacted) with an "update ipv6" button, and a "changed" status of "1 week ago" with a "delete domain" button. Above the domain list, there's a form to add a new domain: "http://" + "sub domain" + ".duckdns.org" + "add domain".

At the bottom, there's a "Donate" button, a Patreon logo, and links to "Terms of Use" and "Privacy Statement".

Screenshot 7. Free Domain Service Duck DNS

To establish a domain for the RAPID project using Duck DNS:

1. **Visit Duck DNS**
  - a. Navigate to the Duck DNS website (<https://www.duckdns.org>) and sign in using one of the available authentication options (Google, GitHub, etc.).
2. **Create a Subdomain**



- a. Once logged in, create a subdomain by entering the desired name (e.g., rapid-int) in the provided field.
- b. Click Add Domain to reserve it.

### **3. Note Your Token**

- a. After creating the subdomain, note down the token provided by Duck DNS. This token will be required to configure your domain's IP address updates.

### **4. Update IP**

- a. Go to what's my <https://whatismyipaddress.com/>
- b. Copy and paste the ipv4 into the current ip and click on update ip.

### **5. Test Your Domain**

- a. Access your domain (e.g., rapid-int.duckdns.org) in a browser to confirm it's properly resolving to your Raspberry Pi's IP address.