Abstract

The cleaning workforce management system (CWMS) is an app-based solution for a cleaning company. The App allows managers to schedule and roster staff, trace cleaners' attendance by using QR codes (check-in/out) with location-based services, manage tasks by assigning tasks and tracking their status as well as reporting any issues for maintenance. This App has achieved all the key functions required by the client through the MVC system framework. In the future, this app can improve the experience of human-computer interaction, such as user interface and experience by linking to third-party applications to improve user experience in communication, notification, and map functions.

1. Introduction

The client's cleaning company has been dealing with issues managing tasks and staff, i.e., checking their status in an efficient way. To use to CWMS App, the clients will successfully improve work efficiency by reducing communication time and enhancing work progress updates which can ultimately result in financial increase in a long run. Traditionally, using pre-assigned task sheet is less flexibility. By communicating on the App creates more flexibility and less labour cost. At this stage, managers and employees can realise basic task and workforce management to improve their work efficiency.

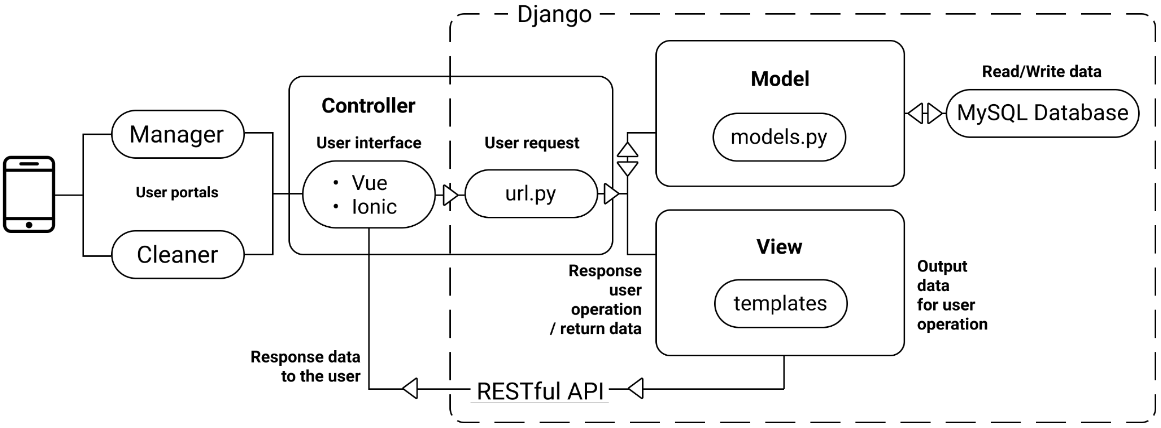
In this report, project aims, approaches, and results will be illustrated. In section 2, it will demonstrate the project aims by providing a workforce management solution to improve the efficiency for both managers and cleaners through the App, including the manifestation through QR code location-based service collaborating with the work-shift and task arrangement, and monitor specific missions, subtask and maintenance status. In section 3, it will show the process and approaches we use, including methodologies and software tools. In section 4, it will discuss the achievement of the project in terms of the implementation outcomes, testing feedback and the reflection from the development process. Last but not least, the outline of the project outcomes and the potential opportunities of the project will be expounded in conclusion.

2. Project Aims

The project aims to develop a workforce management App to let different users complete the tasks and send the data to each other by the MVC framework to improve their work efficiency. The realisation of the goal includes two aspects: staff and task management. The project intends to use MVC to read the data input by the user and store it into the MySQL database and output the data to the front-end user interface to accomplish the functions of the App.

3. Approaches

3.1. Software Architecture - MVC framework



The key concept of the program is based on the MVC framework, which is an architectural pattern that separates applications into Model, View, and Controller and each architectural component are built to address an application development (Principe & Yoon, 2015):

- The **Controller** handles the user interaction, informing the model and the view to change as appropriate. In the APP, users log in based on their roles to enter the corresponding program. The user can operate through the front-end user interface built by the Vue framework and send the URL request to the Model.
- The **Model** component stores data and its associated logic. It transfers data between controller components, retrieves user info from the database, and manipulates the data. In CWMS, the Model receives the request and decide whether to write the data to the SQL database by the Class type and read data from the database and return it to the View to the user.
- The **View** is created by the data collected from the model data to represent the data. A view requests the model to give information to present the output presentation to the user. In our project, it also represents the data from photos and text boxes by UI components. The front-end can receive the data returned from the View through the RESTful API interface and present it on the user interface for the user to conduct the next operation.

3.2.Main tasks

In general, we achieve the project goals in 2 milestones. In each milestone, we establish several key tasks and we split them into sub-tasks to implement. *(All task tickets refer to Appendix 1)*

Milestone 1

- User Analysis & User-Centred Design include user expectations analysis, case studies, usability heuristics evaluations, user flows analysis, App structure analysis and success measures.
- Prototyping includes Wireframing and High-fidelity prototyping.
- Software Structure Design based on MVC framework.
- Programming language research & study include React-Redux, Angular and Vue frameworks for the front-end and Python and Java for back-end development After comparison, we settled on Vue and python 3, then we studied by ourselves through online resources.
- Front-end interface development and Back-end structure development.
- Testing, Client meetings and Refinement.

Milestone 2

- Front-end Vue framework development.
- Back-end function development includes QR code location-based service and Maintenance.
- Testing, Client meetings and Refinement.
- New requirements include Dashboard and Sub-tasks functions.
- API connection with Restful API.
- User testing includes User Testing Checklist and Questionnaires.
- Testing, Client meeting and Refinement.

3.3.Tools and Technologies

App Design Tools (UX/UI)

- Mind mapping on Coggle *(Appendix 2)*
- Wireframing on Balsamiq
- High-fidelity prototyping on Figma

Every good App and website start with a strong structure (Fling, 2009). We use the Balsamiq wireframe (low-fidelity prototype) to generate digital sketches of our initial ideas and concepts for CWMS, promoting discussion and understanding before writing any code.

Ripping up code is very difficult but prototyping (Kiczales, 1997). I used Figma to create high fidelity prototypes to make our designs as close to the real user interface as possible. This process is effective at gathering real human performance data (such as user flow and functionality) and demonstrating actual products to customers. Hence, we used it to meet with the client before we could architect the code structure *(Appendix 3).*

Software Tools

The implementation of CWMS is mainly achieved by the user operation of the App, the front-end Vue framework, the back-end Django framework and the collaboration of the MySQL database.

- **Front-end:** HTML 5, CSS, Bootstrap, JavaScript, Vue, Node.js.
- **Back-end:** Python 3, JSON, Django, MySQL, Restful API.

On the user end, users submit and obtain data from different ports through different roles, and the front-end web pages interact through Vue. The front-end will send a get/post request, and the server will respond after receiving it, and send the data to the front-end page in JSON format. Vue can send a URL request to the back-end to request the data interface by Django.

The front-end and back-end use restful API for data interaction. If the backend can retrieve the data from the database, it will return the data in response. In the back-end, RESTful API, the data between Django (high-level Python Web framework) and the MySQL database is by ‘read & write’ to store and retrieve data.

Task management Tool

Our task management runs great on Agile. Notion provides an overview for all key tasks and we generated the sub-tasks to various sprints in every single ticket *(Appendix 1)*. Team members organise tickets into the date, responsibility, priority and releases, as well as monitor the workload and task assignments of the team.

3.4.Methodologies

3.4.1.User analysis & user-centred Design

- **Consider User Expectations**

  We want each step and interface to convey what it is supposed to convey what is necessary to maximize efficiency. To maximise usability, we validated the clarity of the UI by matching prominent elements to prominent compositions based on user habits as cues.

- **Evaluate Usability Heuristics**

  Heuristic assessment is a set of design principles based on conventions, standards, and best practices to improve the overall usability of a product, including error prevention, minimising the opportunity for users to make mistakes (Toptal, 2021). Our project used the UI components and UI Kits of Bootstrap, Vue to maximise our interface specification.

- **Analyse the User Flows**

  We count the number of interaction steps (each swipe and click to complete a function) and consider the impact of the number of steps and complexity of the task on the user.

- **Evaluate the App Structure**

  We took into account the pages of features about the ease to switch between different features.

- **Measure Success**

  For the phased development, we think repeatedly about the problem areas that will affect future design updates. The success of these updates can be measured by comparing changes in user data and running subsequent analysis.

3.4.2.Agile

We used Agile as an iterative approach to assist our project management and software development. It helped our team to deliver value to reach our goals faster and with fewer headaches. Instead of betting everything on a giant launch, we split the tasks into small and consumable sub-tasks. Every task requirement, overall plans, and periodical results are evaluated continuously to have a natural mechanism to respond to the changes and updates *(Appendix 1).*

4. Results

4.1.Project Achievement

The project achieves most functions to meet the user expectations in terms of the function realisation. Besides, the Agile project management runs well to iterate and improve the program during the development process. Details and examples show as below:

Implementation and work efficiency

- On the manager portal, the manager can create tasks and staff in all properties (name, number, time, address, etc) and see their status as ‘ready to work’ before assigning them. The data can write in the MySQL Database through the URL request and the models.py.

- The manager can also assign tasks to staff, the status of them can change to 'working'. On the staff portal, the staff can see the assign tasks on their list and check the task details. The View can show the response from the Model and output responding UI.
- The staff can scan QR code to confirm the task name, time, and address, and check-out afterwards. The Model can check the url.py from the Controller and make decision by models.py.
- The staff can create sub-tasks and check for complement; they can submit maintenance photos and text description to the manager. The data can successfully store in the MySQL local server before getting any read request.
- On the manager portal, the manager can check the status of the tasks according to the data responding from the Model; they can also check the subtasks status and maintenance details sent by the RESTful API.

4.2.Individual Achievement

As a scrum master in the Team, I am responsible for App design, development support, project management, and project agility:

- I have successfully supported back-end developers to develop the QR code and maintenance function; I test and provide evaluation feedback for the refinement. Also, I supported front-end developers in the front-end UI components and I developed and refined the interactive web pages.
- I have successfully analysed user requirements and generated wireframe and prototype for the user test, designed UI components and participated in system-architectural research and design.
- I have monitored project progress, provided timely feedback, addressed potential barriers to ensure the project is on track and meets deadlines with the expected outcomes.
- I host regular team meetings and facilitate communication and information exchange (such as project progress and roadblocks) between team members and clients.
- I advocate the use of continuous integration to reduce the risk, time and effort. I encourage developers to collaborate in real-time to reduce development time and improve product architecture and quality.
- I organise the project in the final delivery, including refining each function, demonstrating the process, setting up the scenario demonstration, designing the unit test, offline user test, questionnaire, user feedback, etc.

4.3.Testing effectiveness

4.3.1.Unit testing

During the period between week 4 and week 11, our major unit tests include four aspects: test strategy, test execution, test management and risk management. The test objectives aim to ensure the App features, functionalities, performance and usability can work effectively. Specific testing directions include the following:

- **Unit tests** include individual functions on both portals to ensure that each role can accomplish its work needs.
- **Validation tests** ensure the functions can post and get data from the database for the data transferring between two roles.

- **Compatibility tests** for the webpage responsibility and web pages jump in different environments (systems), such as Chrome, Firefox, IE browsers, macOS Big Sur, Linux Ubuntu and Windows 10.
- **Operation tests** for web page demonstration in different router links and form validation for users' input. For the front-end input verification, staff/task creation module tests read this data into the database and return result data to the web pages.
- **List module tests** for the server query and write data to web pages.
- **Check-in/out module** for behaviours emulation for functional validation.
- **Defect tests** help to reduce the program bugs, which requires developers to debug and handle errors to find out the programming flaw.
- **Data consistency tests** for the input data and variables bound.
- **SQL injection tests** for database security verification.
- **Cross-Origin issue tests** for different domain names, protocols and port numbers with the proxy server and CORS module to resolve.
- **Usability tests** contain the implementation of each function and user experience according to the user's usage requirements and relevant logic according to the scenarios.

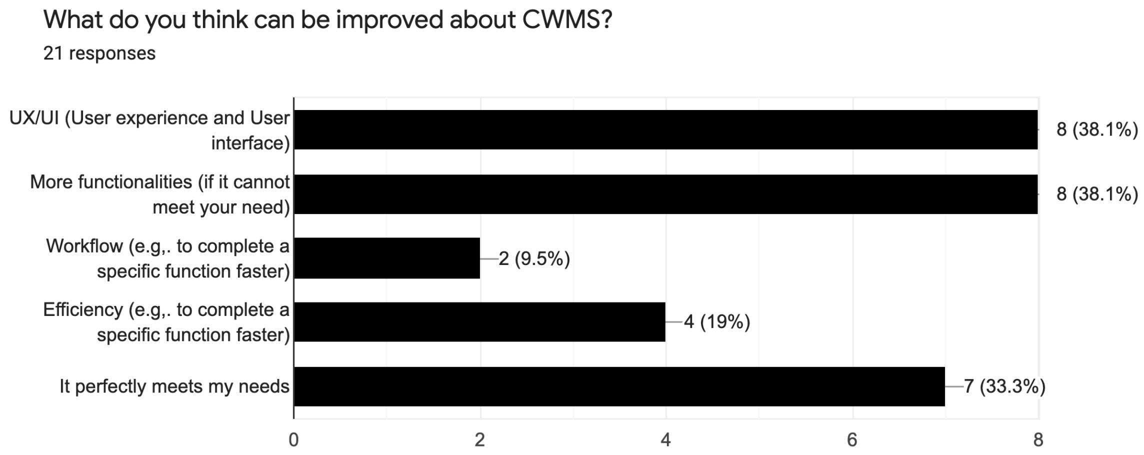
4.3.2.User Testing Checklist & Survey

User Testing Survey provides a checklist for users to physically test the App. It simulates the workplace environment based on the prepared scenarios, including two task chains, three addresses, three characters, and twenty-two actions. The survey has undergone four versions in total. In addition, we provide questionnaires for the participants for feedback to improve our project *(Appendix 5).*

4.3.3.Questionnaires

The questionnaire was created on Google Form. There are 21 participants and the responses are summarised as follow *(Appendix 7):*

- The participants include 38.1% MCI project students, 28.6% general students, 14.3% potential clients/users and supervisors. 52.5% of them has experience in using A task management App before (such as Tradify and Jira).
- On a scale of 1 to 5, 61.9% of respondents gave their overall experience as a manager using CWMS a score of 4; 52.4% of them rated 4 as a cleaner. 81% of users found the app easy to use. 52.4% of them will choose CWMS if they have the needs in the future.
- Although 33.3% of the participants think that the CWMS perfectly meets their needs, 38.1% of them think that the user experience of the App can be improved, and the same number expect that the App can have more functions.



- Some users expect the App to have a message bar and night mode for users to communicate.

4.3.4.Case studies

We have studied and analysed five success metrics, including Deputy, UKG Dimension, Tradify, Jobber and mySA GOV.

- **System structure and design**

  Their system design and architecture inspire us in roster and shifts planning, interface, activity management, task display and status display. Besides, we learned the design of SA Gov is in line with the habits of South Australians, who have become accustomed to the QR code scanning process during the COVID-19 epidemic.

- **Programming components**

  To improve the efficiency of the development process, reusable components can be adapted to respond to the needs of a particular project (Weide, 1991). We learned from those case studies; we use several libraries of functional components. Our developers can utilise our program components on multiple pages to managing our resources more effectively to speeds up the completion and maintain the code easier.

5. Conclusion

In conclusion, the project has sufficiently met the user needs, allowing managers and employees to implement basic tasks and workforce management. During the process, the project has performed unit tests, user testing and questionnaires to confirm a positive result of the product. Through the developing stages, I have learned the methods of project management, case research, code efficiency, software technologies, communication, unit and user testing. It improved my understanding and experience of software development. In the future, I look forward to deploying the App, which will improve the test efficiency and get more feedback data. In addition, I hope to develop more the functions of the App, such as third-party API connection, guiding maps, notifications, user interaction and other functions.

References

1. Deputy, 2021, Run efficient teams without the hours of admin https://www.google.com/search?q=Deputy&oq=Deputy&aqs=chrome..69i57j69i59j0i271j69i59j69i65.153j0j7&sourceid=chrome&ie=UTF-8

2. Fling, B., 2009. Mobile design and development: Practical concepts and techniques for creating mobile sites and Web apps. " O'Reilly Media, Inc.".

3. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.M. and Irwin, J., 1997, June. Aspect-oriented programming. In European conference on object-oriented programming (pp. 220-242). Springer, Berlin, Heidelberg.

4. mySA GOV, 2021, State Government of South Australia, https://apps.apple.com/au/app/mysa-gov/id1124301168

5. Principe, M., & Yoon, D. (2015). A web application using MVC framework. In Proceedings of the International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government (EEE) (p. 10). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

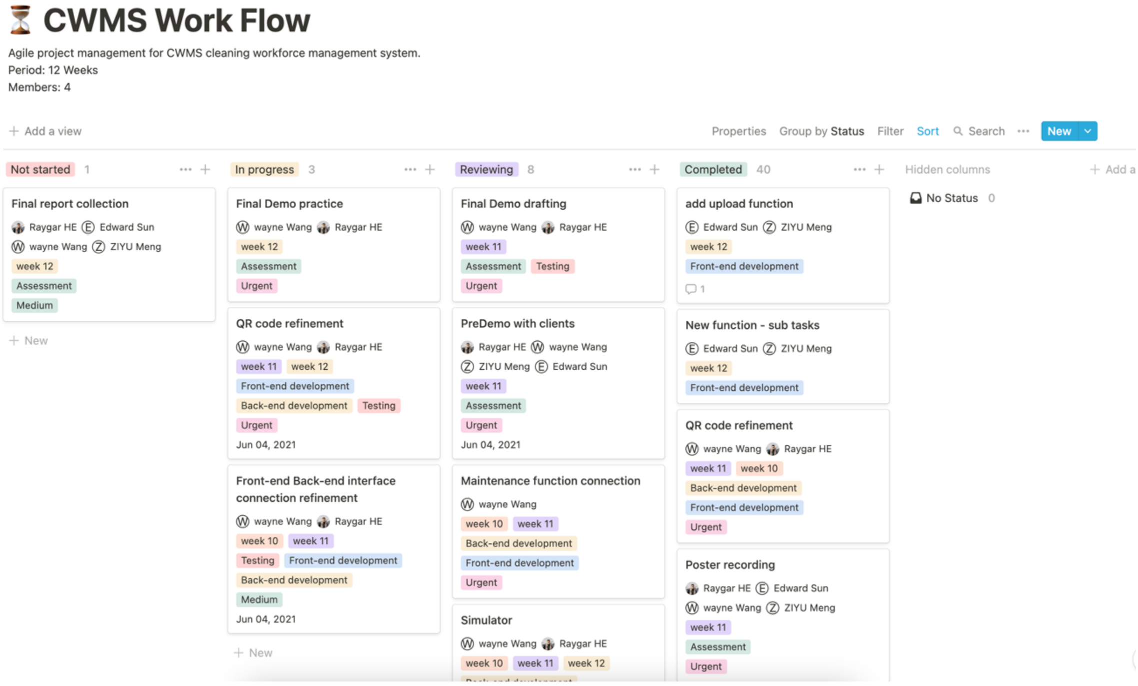
6. Toptal, 2021 Enhance User Flow – A Guide to UX Analysis, https://www.impactplus.com/blog/sans-serif-vs-serif-font-which-should-you-use-when

7. UKG Dimensions, 2021, New Dimensions in Work, https://apps.apple.com/au/app/ukg-dimensions/id1321967136

8. Weide, B.W., Ogden, W.F. and Zweben, S.H., 1991. Reusable software components. In Advances in computers (Vol. 33, pp. 1-65). Elsevier.

Appendix

1. Task management on Notion



⌛ CWMS Work Flow

Agile project management for CWMS cleaning workforce management system.
Period: 12 Weeks
Members: 4

+ Add a view — Properties | Group by Status | Filter | Sort | Search | ··· | New

Not started 1

- Final report collection — Raygar HE, Edward Sun, wayne Wang, ZIYU Meng — week 12 — Assessment — Medium

+ New

In progress 3

- Final Demo practice — wayne Wang, Raygar HE — week 12 — Assessment — Urgent
- QR code refinement — wayne Wang, Raygar HE — week 11, week 12 — Front-end development — Back-end development, Testing — Urgent — Jun 04, 2021
- Front-end Back-end interface connection refinement — wayne Wang, Raygar HE — week 10, week 11 — Testing, Front-end development — Back-end development — Medium — Jun 04, 2021

+ New

Reviewing 8

- Final Demo drafting — wayne Wang, Raygar HE — week 11 — Assessment, Testing — Urgent
- PreDemo with clients — Raygar HE, wayne Wang, ZIYU Meng, Edward Sun — week 11 — Assessment — Urgent — Jun 04, 2021
- Maintenance function connection — wayne Wang — week 10, week 11 — Back-end development — Front-end development — Urgent
- Simulator — wayne Wang, Raygar HE — week 10, week 11, week 12

Completed 40

- add upload function — Edward Sun, ZIYU Meng — week 12 — Front-end development — 1
- New function - sub tasks — Edward Sun, ZIYU Meng — week 12 — Front-end development
- QR code refinement — wayne Wang, Raygar HE — week 11, week 10 — Back-end development — Front-end development — Urgent
- Poster recording — Raygar HE, Edward Sun, wayne Wang, ZIYU Meng — week 11 — Assessment — Urgent

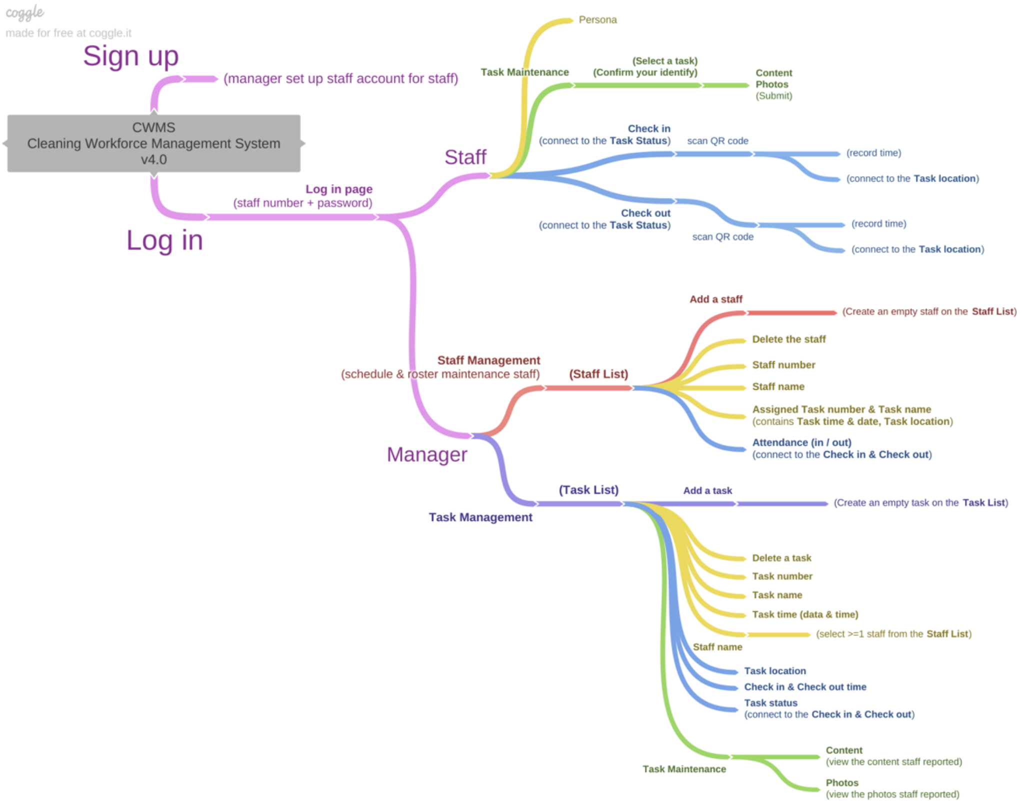
Hidden columns

- No Status 0

+ Add a

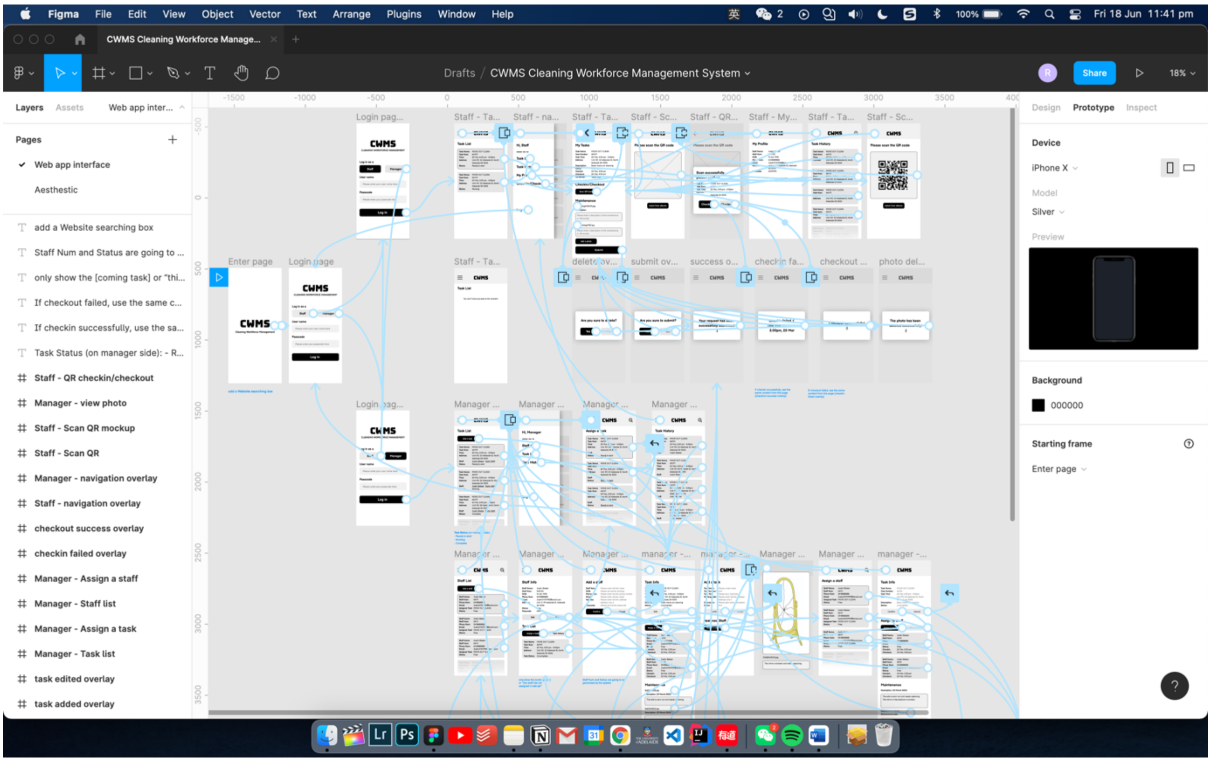
https://www.notion.so/raygar/015c23fdf9574552824da9724b82ef0a?v=08906e2dfa3a4abfbccdbd09e7fd0c2b

2. CWMS Mind Mapping on Coggle


coggle
made for free at coggle.it
Sign up
(manager set up staff account for staff)
CWMS
Cleaning Workforce Management System
v4.0
Log in
Log in page
(staff number + password)
Staff
Persona
Task Maintenance
(Select a task)
(Confirm your identify)
Content
Photos
(Submit)
Check in
(connect to the Task Status)
scan QR code
(record time)
(connect to the Task location)
Check out
(connect to the Task Status)
scan QR code
(record time)
(connect to the Task location)
Manager
Staff Management
(schedule & roster maintenance staff)
(Staff List)
Add a staff
(Create an empty staff on the Staff List)
Delete the staff
Staff number
Staff name
Assigned Task number & Task name
(contains Task time & date, Task location)
Attendance (in / out)
(connect to the Check in & Check out)
Task Management
(Task List)
Add a task
(Create an empty task on the Task List)
Delete a task
Task number
Task name
Task time (data & time)
(select >=1 staff from the Staff List)
Staff name
Task location
Check in & Check out time
Task status
(connect to the Check in & Check out)
Task Maintenance
Content
(view the content staff reported)
Photos
(view the photos staff reported)

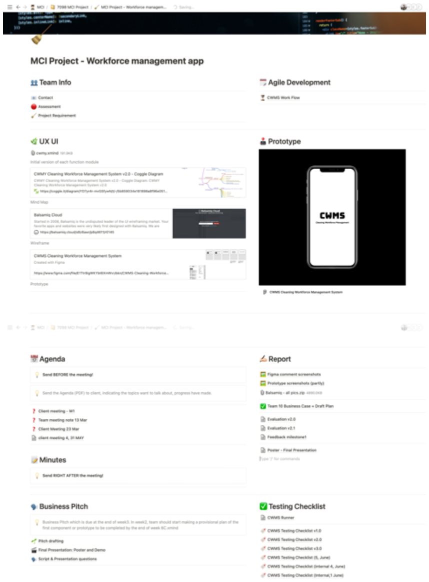
https://coggle.it/diagram/YEcpxqW-yh05c3ch/t/cwms-cleaning-workforce-management-system-v4-0/96887954814d4f00d0ae7629886895bff8141976c45e190b3c528dbd8b7a3352

3. **High-fidelity prototyping on Figma**


Figma File Edit View Object Vector Text Arrange Plugins Window Help
Fri 18 Jun 11:41 pm
CWMS Cleaning Workforce Manage...
Drafts / CWMS Cleaning Workforce Management System
Share
18%
Layers Assets Web app inter...
Pages
Web app interface
Aesthetic
add a Website searching box
Staff Num and Status are going to ...
only show the [coming task] or "thi...
If checkout failed, use the same c...
If checkin successfully, use the sa...
Task Status (on manager side): - R...
Staff - QR checkin/checkout
Manager - view photo
Staff - Scan QR mockup
Staff - Scan QR
Manager - navigation overlay
Staff - navigation overlay
checkout success overlay
checkin failed overlay
Manager - Assign a staff
Manager - Staff list
Manager - Assign a task
Manager - Task list
task edited overlay
task added overlay
Design Prototype Inspect
Device
iPhone X
Model
Silver
Preview
Background
000000
Starting frame
Enter page
CWMS

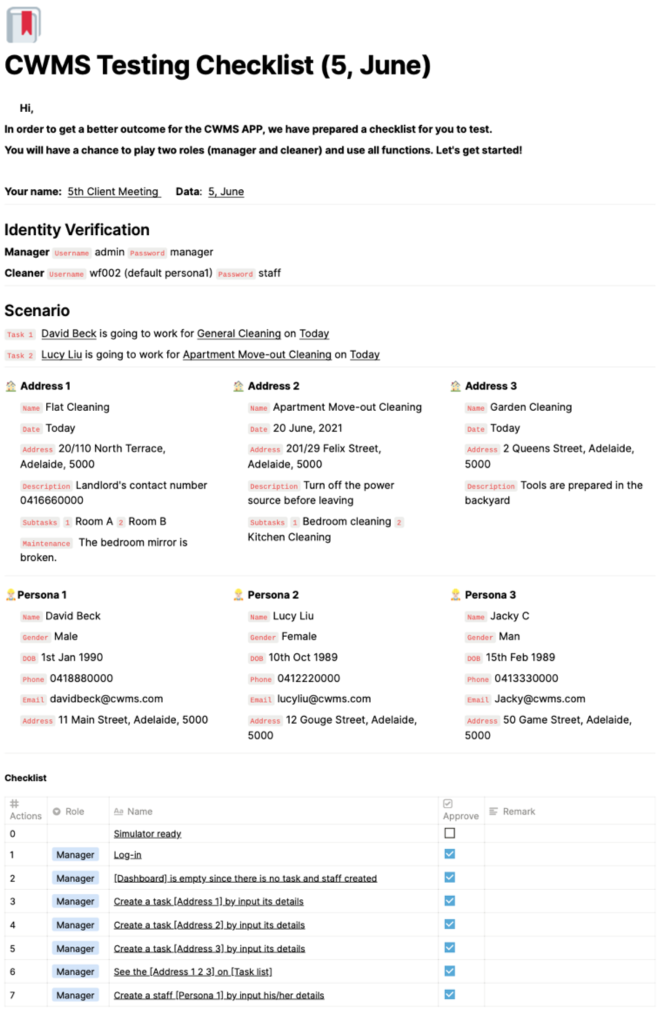
https://www.figma.com/proto/E1Ttr8igWKYbt6XmWvUbkn/CWMS-Cleaning-Workforce-Management-System?page-id=0%3A1&node-id=104%3A7&viewport=109%2C-751%2C0.37997907400131226&scaling=scale-down

4. **Content management on Notion**



https://www.notion.so/raygar/MCI-Project-Workforce-management-app-1e1e7ea016604b51ab389777a3fff8be

5. User Testing Checklist & Survey



CWMS Testing Checklist (5, June)

Hi,

In order to get a better outcome for the CWMS APP, we have prepared a checklist for you to test.

You will have a chance to play two roles (manager and cleaner) and use all functions. Let's get started!

Your name: 5th Client Meeting **Data:** 5, June

Identity Verification

Manager Username admin Password manager

Cleaner Username wf002 (default persona1) Password staff

Scenario

Task 1 David Beck is going to work for General Cleaning on Today

Task 2 Lucy Liu is going to work for Apartment Move-out Cleaning on Today

Address 1

Name Flat Cleaning

Date Today

Address 20/110 North Terrace, Adelaide, 5000

Description Landlord's contact number 0416660000

Subtasks 1 Room A 2 Room B

Maintenance The bedroom mirror is broken.

Address 2

Name Apartment Move-out Cleaning

Date 20 June, 2021

Address 201/29 Felix Street, Adelaide, 5000

Description Turn off the power source before leaving

Subtasks 1 Bedroom cleaning 2 Kitchen Cleaning

Address 3

Name Garden Cleaning

Date Today

Address 2 Queens Street, Adelaide, 5000

Description Tools are prepared in the backyard

Persona 1

Name David Beck

Gender Male

DOB 1st Jan 1990

Phone 0418880000

Email davidbeck@cwms.com

Address 11 Main Street, Adelaide, 5000

Persona 2

Name Lucy Liu

Gender Female

DOB 10th Oct 1989

Phone 0412220000

Email lucyliu@cwms.com

Address 12 Gouge Street, Adelaide, 5000

Persona 3

Name Jacky C

Gender Man

DOB 15th Feb 1989

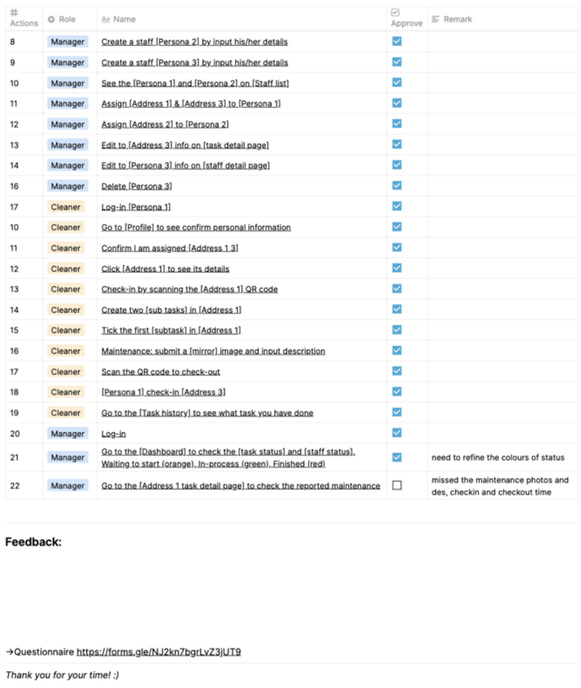
Phone 0413330000

Email Jacky@cwms.com

Address 50 Game Street, Adelaide, 5000

Checklist

# Actions	Role	Name	Approve	Remark
0		Simulator ready	☐	
1	Manager	Log-in	☑	
2	Manager	[Dashboard] is empty since there is no task and staff created	☑	
3	Manager	Create a task [Address 1] by input its details	☑	
4	Manager	Create a task [Address 2] by input its details	☑	
5	Manager	Create a task [Address 3] by input its details	☑	
6	Manager	See the [Address 1 2 3] on [Task list]	☑	
7	Manager	Create a staff [Persona 1] by input his/her details	☑	



# Actions	Role	Name	Approve	Remark
8	Manager	Create a staff [Persona 2] by input his/her details	☑	
9	Manager	Create a staff [Persona 3] by input his/her details	☑	
10	Manager	See the [Persona 1] and [Persona 2] on [Staff list]	☑	
11	Manager	Assign [Address 1] & [Address 3] to [Persona 1]	☑	
12	Manager	Assign [Address 2] to [Persona 2]	☑	
13	Manager	Edit to [Address 3] info on [task detail page]	☑	
14	Manager	Edit to [Persona 3] info on [staff detail page]	☑	
16	Manager	Delete [Persona 3]	☑	
17	Cleaner	Log-in [Persona 1]	☑	
10	Cleaner	Go to [Profile] to see confirm personal information	☑	
11	Cleaner	Confirm I am assigned [Address 1 3]	☑	
12	Cleaner	Click [Address 1] to see its details	☑	
13	Cleaner	Check-in by scanning the [Address 1] QR code	☑	
14	Cleaner	Create two [sub tasks] in [Address 1]	☑	
15	Cleaner	Tick the first [subtask] in [Address 1]	☑	
16	Cleaner	Maintenance: submit a [mirror] image and input description	☑	
17	Cleaner	Scan the QR code to check-out	☑	
18	Cleaner	[Persona 1] check-in [Address 3]	☑	
19	Cleaner	Go to the [Task history] to see what task you have done	☑	
20	Manager	Log-in	☑	
21	Manager	Go to the [Dashboard] to check the [task status] and [staff status]. Waiting to start (orange), In-process (green), Finished (red)	☑	need to refine the colours of status
22	Manager	Go to the [Address 1 task detail page] to check the reported maintenance	☐	missed the maintenance photos and des, checkin and checkout time

Feedback:

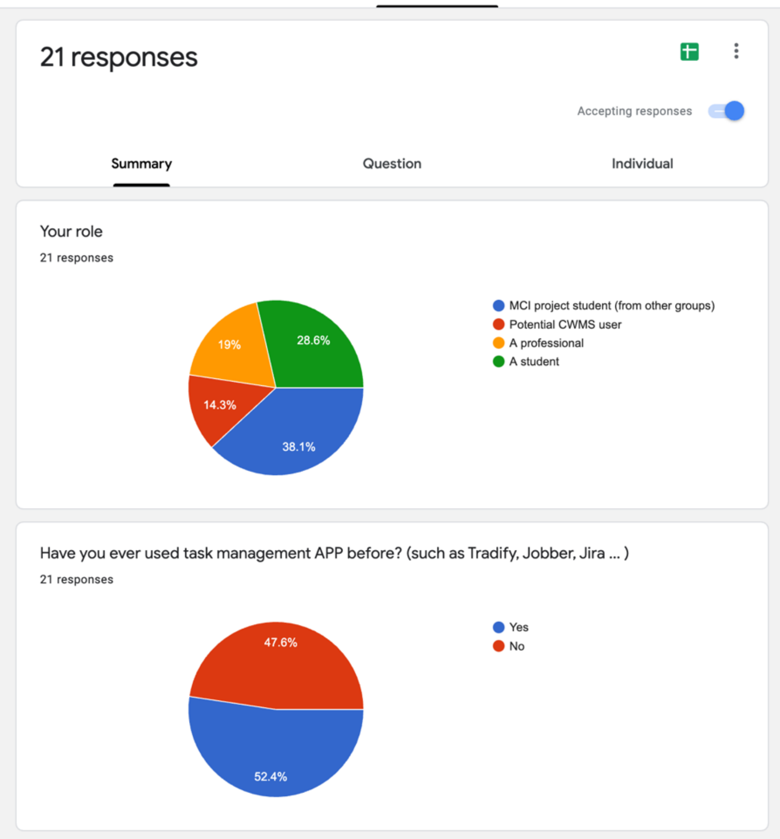
→Questionnaire https://forms.gle/NJ2kn7bgrLvZ3jUT9

Thank you for your time! :)

https://www.notion.so/raygar/CWMS-Testing-Checklist-v3-0-27c5f8ca6d09445890a0798eafd37986

6. **CWMS Questionnaires**
   https://forms.gle/NJ2kn7bgrLvZ3jUT9

7. **Questionnaires Response**



21 responses

Accepting responses

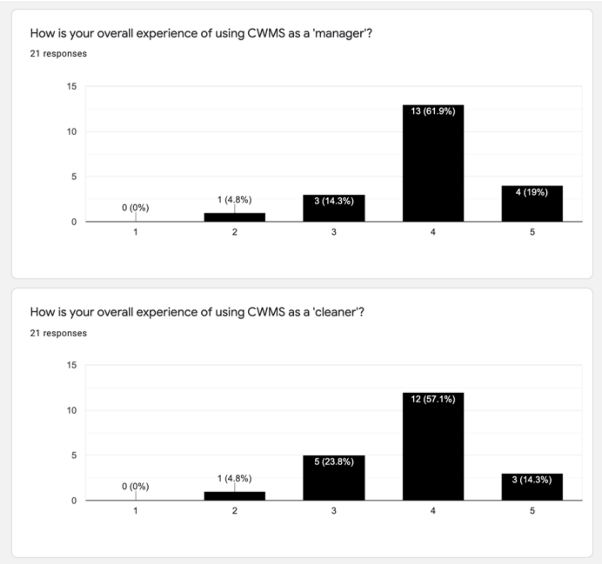
Summary | Question | Individual

Your role

21 responses

Have you ever used task management APP before? (such as Tradify, Jobber, Jira ...)

21 responses

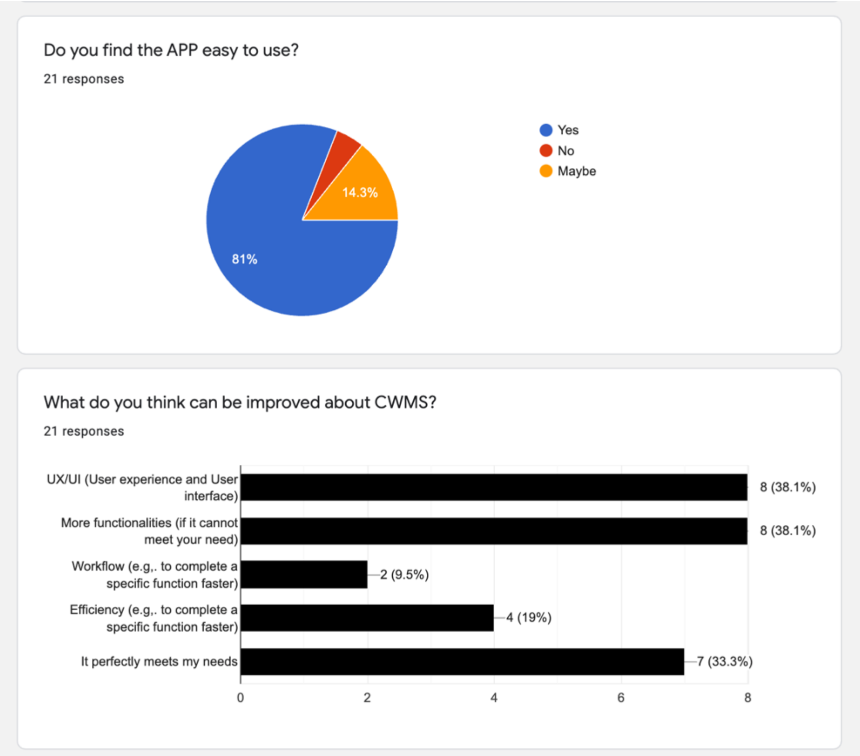


How is your overall experience of using CWMS as a 'manager'?

21 responses

How is your overall experience of using CWMS as a 'cleaner'?

21 responses



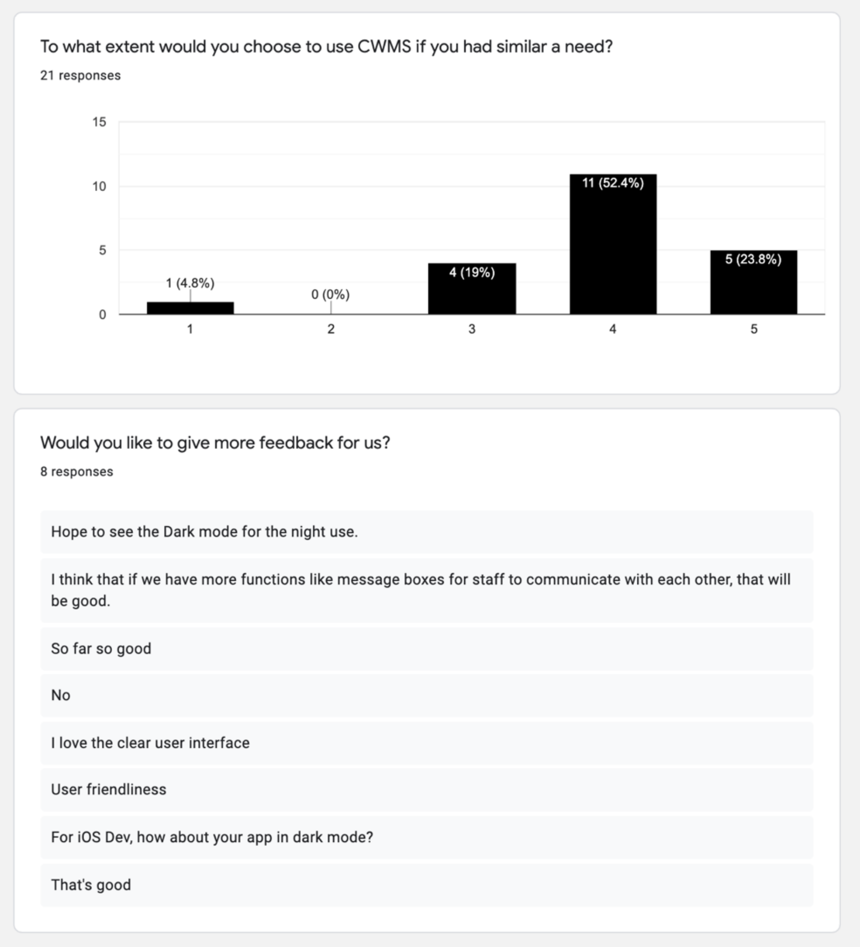
Do you find the APP easy to use?

21 responses

What do you think can be improved about CWMS?

21 responses

UX/UI (User experience and User interface)
8 (38.1%)
More functionalities (if it cannot meet your need)
8 (38.1%)
Workflow (e.g,. to complete a specific function faster)
2 (9.5%)
Efficiency (e.g,. to complete a specific function faster)
4 (19%)
It perfectly meets my needs
7 (33.3%)
0
2
4
6
8



To what extent would you choose to use CWMS if you had similar a need?

21 responses

Would you like to give more feedback for us?

8 responses

Hope to see the Dark mode for the night use.

I think that if we have more functions like message boxes for staff to communicate with each other, that will be good.

So far so good

No

I love the clear user interface

User friendliness

For iOS Dev, how about your app in dark mode?

That's good