|  |
| --- |
| ICT30017 - ICT PROJECT A |
| SPRINT 1 REPORT |
| PORTFOLIO TASK 4 |

|  |
| --- |
| SUBMITTED: 08/0/2024.  SUBMITTED: 19/04/2024 |

GROUP 2

**Table of Contents**

1. [Acknowledgement of Country 2](#_Toc141109977)
2. [Contributions 2](#_Toc141109977)

2.1. [Rubie Stannard 2](#_Toc141109978)

2.2. [Qiao Jun Chan 2](#_Toc141109979)

2.3. [Dylan Morrison 2](#_Toc141109978)

2.4. [Mingyuan Wang 2](#_Toc141109979)

2.5. [Henry Hua Rong Wang Hong 2](#_Toc141109979)

2.6. [Tan Dat Do 2](#_Toc141109979)

1. [Sprint 1 Report 3](#_Toc141109977)

3.1. [Sprint 1 Plan 3](#_Toc141109978)

3.2. [Progress Made 3](#_Toc141109979)

3.3. [Supervisor Feedback 4](#_Toc141109979)

3.4. [Critical Analysis 4](#_Toc141109978)

1. [Challenges 6](#_Toc141109977)
2. [Lessons Learned 7](#_Toc141109977)
3. [Next Sprint Plan 8](#_Toc141109977)
4. [Tracking and Monitoring 11](#_Toc141109977)
5. [Quality Plan 12](#_Toc141109977)

8.1. [Target Quality Expectations 12](#_Toc141109978)

8.2. [Steps to Assure Quality 12](#_Toc141109979)

1. **Acknowledgement of Country**

We acknowledge the traditional custodians of the land on which we gather, the Wurundjeri people of the Kulin Nation, and pay our respects to their elder’s past, present, and emerging. We recognize their continuing connection to the land and culture and honour the significant role they play in shaping the identity and history of this region.

1. **Contributions**

**2.1. Rubie Stannard: 103982732**

I contributed to this task by making the document, adding the cover page, table of contents, tables, headings, writing the acknowledgement of country, and completing the sprint 1 report and next sprint plan. Over the last 3 weeks, I organised the HTML pages so there’s not as many as originally and wrote all of the data for the profiles.

**2.2. Qiao Jun Chan: 103490604**

I have documented the tools that were utilized throughout the development process. Throughout sprint 1, while developing the management system I didn’t contribute much to the main coding stuff. However, I have suggested the template and colour theme in designing the looks of our website. In the next sprint, I have started working on the admin portal, ensuring all the features that have to be accessed by admin are included.

**2.3. Dylan Morrison: 101111673**

I contributed to this task by providing details for the challenges section, elaborating on some of the struggles we have faced as a group. Throughout sprint 1, while awaiting the chance to work on the JavaScript additions to our project, I did my best to help guide and critique my fellow group mates where I saw I could help, or when asked. Towards the end of sprint 1 I also began development of a roster/scheduling system to allow for all staff members to see their current week obligations in a read-only format, and for administration staff to edit and modify the schedules of other staff members.

**2.4. Mingyuan Wang: 104195667**

Throughout the sprint 1, I mostly take charge of the back-end section followed by the prototype that was created by Rubie. Also, I contributed to the report by describing challenges, quality expectations and steps testing for quality.

**2.5. Henry Hua Rong Wang Hong: 104792738**

Based on my previous teamwork experience, I completed the Lessons Learned. In Sprint1, my task was to work with Tan on HTML files and optimize the appearance of HTML pages using CSS files. I independently designed Facilities, Member profile list and other HTML pages, and designed CSS files for them. I modified the Inventory page according to Wang's requirements. I fixed a bug with the navigation bar. Based on my previous teamwork experience, I completed the Lessons Learned.

**2.6. Tan Dat Do: 103498255**

I contributed to the lessons learned part of this document. I have contributed to sprint 1 by creating a UI homepage and service page to show our customer. Furthermore, I was helping front-end development and analysis the problem in UI and optimize if there any problem in scaling in page.

1. **Sprint 1 Report**

**3.1. Sprint 1 Plan**

**Week 5 - 25/03/2024 to 07/04/2024**

|  |  |
| --- | --- |
| **Task** | **Status** |
| Create the website prototype | Complete |
| Create the pages ready for coding | Complete |

**Week 6 - 08/04/2024 to 14/04/2024**

|  |  |
| --- | --- |
| **Task** | **Status** |
| Add code to the HTML pages | In progress |
| Add code to the CSS pages | In progress |

**Week 7 - 15/04/2024 to 21/04/2024**

|  |  |
| --- | --- |
| **Task** | **Status** |
| Back-end PHP development | In progress |
| Write information for the databases | In progress |
| Add the JavaScript | Not started |

**3.2. Progress Made**

In the first sprint, the team didn’t make as much progress as planned, whether this is because we didn’t put the time and effort in, or because the tasks for the first sprint were too much. The tasks that were completed were the week 5 task 1 prototype creation, and the week 5 task 2 of creating the pages for the code.

A screenshot of a computer

Description automatically generated

Image 1: Week 5 task 1 and task 2 prototype and code page creation.

Week 6 tasks involved completing the HTML and CSS code, but this was not completed. The team started coding the HTML and CSS, hopefully ready to be completed in week 8.

A computer screen shot of a computer

Description automatically generated

*Image 2: Week 6 task 1 and task 2 HTML and CSS coding.*

Week 7 tasks involved the back-end development and creating data for the databases. The tasks were to code the PHP, which was started locally but not uploaded to the GitHub, write information for the databases such as profiles for members and staff, and code some JavaScript. The first two tasks were started, but not complete, and the third task wasn’t started.

A screenshot of a computer

Description automatically generated

*Image 3: Week 7 task 1 and task 2 PHP coding and profile data.*

**3.3. Supervisor Feedback**

Before we changed the task list, the feedback we received was not good. She wanted us to include dates, assigned members, and more detail specific to the task that needed to be complete. After the list changed to include information like this, Vasudha said that this task list was acceptable. She looked at the sprint 1 task list and was sceptical about the amount of progress that could be made. We assured her it could be done in the said time, proving her right that it was too much.

**3.4. Critical Analysis**

The Sprint 1 plan outlined comprehensive objectives and tasks to be completed over a three-week period. Tasks were defined, covering parts such as prototype creation, HTML and CSS coding, back-end PHP development, and database information.

While some progress was achieved, particularly in Week 5, the overall progress fell short of expectations. The team encountered challenges in completing Week 6 tasks related to HTML and CSS coding, with the work commencing but not reaching completion. Additionally, no progress was made on Week 7 tasks.

Vasudha expressed scepticism regarding the feasibility of achieving the outlined tasks within the allocated time frame. Despite assurances from the team, subsequent progress demonstrated that the workload was indeed ambitious, aligning with Vasudha's initial apprehensions.

This analysis reveals several insights into the team's performance during Sprint 1. While efforts were made to stick to the planned tasks, challenges emerged, potentially coming from either task complexity or resource constraints. The completion of Week 5 tasks demonstrated the team's capability to initiate tasks effectively. However, the inability to fully complete HTML and CSS coding tasks in Week 6 and the lack of progress in Week 7 highlight areas for improvement.

Moving forward, it’s important to conduct a detailed review of task allocation and resource management strategies. Additionally, proactive measures should be taken to address challenges promptly, ensuring that future sprint progress is smoother. Supervisor feedback on the team's performance provides valuable insights and guidance for course correction in subsequent sprints.

In conclusion, while Sprint 1 witnessed some achievements, it also revealed areas for improvement. By critically analysing progress and feedback, the team can leverage lessons learned to enhance performance and achieve greater success in future sprints.

1. **Challenges**

Through examining the challenges we faced during Sprint 1 of our project, we aim to identify areas for improvement and implement strategies to overcome obstacles, fostering a more efficient and cohesive project execution in subsequent sprints.

1. **Unclear infrastructure goals and implementation**

* Clarification was needed regarding the necessity of databases and their hosting and utilization specifics, leading to uncertainty about data management and the needed data.
* There was confusion around utilisation of GitHub and sharing the data for collaboration.

1. **Ambiguous requirements within the project**

* We saw confusion around what details were and weren’t necessary for patients, staff, and admins, as well as the specifics to each of those roles.
* Uncertainty persisted regarding whether any public facing web pages would exist, or if it was exclusively designed to be used by staff/admin.
* Excessive time spent on requirement discussions delayed actual project progression, impacting timelines.

1. **Communication issues**

* As there was so much discussion as a group and between individual members as to the specifics of our project and what features would best meet the requirements, miscommunications were common at the start of the sprint.
* While tasks were assigned to members, updates were inconsistent, leading to confusion as to how much work had been completed and where more resources were required.

1. **Scope creep challenges**

* The lack of clarity on project requirements made it challenging to identify and address scope creep, necessitating ongoing group discussions and delaying progress assessments.

1. **Lessons Learned**

Reflecting on the key insights gained from our collaborative efforts are integral components of project management. Some of the lessons the team learnt in the project thus far are:

1. Communication has been the only way we’ve been able to progress in this project, therefore clear communication has been an important factor for team cooperation. Throughout this project, our team has been realising the significance of clear communication in avoiding misunderstandings and delays, and helping the entire team stay in sync.
2. For our team especially, because of what we’re aiming to achieve, sharing the workload will help us be successful in the project. By sharing the roles and responsibility of tasks, and clarifying what each member is supposed to do, our team can do better work and get it do in a timely manner.
3. Regularly updating the team on the progress of the project has been helping our team solve issues, stay on track, and identify what tasks can be marked as complete, letting other members know that it doesn’t have to be worried about anymore.
4. Since some of what we aim to achieve with the code is complicated, the team has learnt to do independent research when confronted with technical challenges. When team members have done this research, we shard our findings with the others so they can be informed about what’s being done.

By distilling our experiences into valuable lessons and charting a course for improvement, we aim to foster continuous growth and success in our endeavours. Some recommendations and future plans for our teamwork and project are:

1. Establishing clear task deadlines to ensures timely completion by all team members. Making a clear task list with the specific steps that need to be done and deadlines that include dates and times helps control the task completion.
2. Our team’s communication was good, but it could’ve been better. The team knows we have to utilize communication software more to have the best outcome. This includes updating others on the status of the progress, providing members with feedback, and discussing ideas so everyone is on the same page.
3. There are members who don’t know what others know, or don’t know where to look for information. Encouraging members to share their knowledge will enhance overall technical proficiency and expedite problem-solving during collaborative efforts.
4. Documenting any project progress, decisions made, and critical matters will facilitate future reviews and learning endeavours, and ensure members stay up-to-date with the timeline of the project and their individual responsibilities.
5. **Next Sprint Plan**

**Week 8 - 22/04/2024 to 28/04/2024**

**Task 1: Add code to the HTML pages**

**Information:** The HTML pages have been merged so there are less pages. The team is still following the design of the prototype, coding navigation bars, input boxes, a footer, lists, and more. The software used for coding will be VSCode. The following are the new pages and needed code, and a list of pages finished in sprint 1:

* Create member - footer, input boxes, button to say add member
* Create staff - footer, input boxes, button to say add staff
* Facilities - footer, utility metrics, facility related content
* Home - footer, home page content
* Inventory - footer, medication numbers, food numbers, other inventory stuff
* Logon - No code needed. Finished in sprint 1
* Member profile 1-4 - footer, member information, images
* Member profile list - footer, buttons to go to the member profile
* Scheduling - footer, timetable schedule
* Services - footer, in home care and resident information, associated staff member, staff availability
* Staff profile 1-4 - footer, staff information, images
* Staff profile list - footer, buttons to go to the member profile

After the HTML code has been added to the pages the updated pages will be uploaded to GitHub for everyone to download.

**Dates and member:** This task will be started at the start of week 8, which is 22/04/2024, and needs to be finished by the end of week 8, which will be 28/04/2024. The member assigned to this task is Tan.

**Task 2: Add code to the CSS pages**

**Information:** To make all of the HTML code appear on the website, it needs to be styled with CSS. VSCode is the software used to code the CSS. The following CSS pages need to have the basic code styling:

* Profile list - buttons
* Style - input boxes, buttons
* Services - staff detail layout
* Inventory - table
* Scheduling - timetable, 2 sections for the admin schedules

After the CSS has been added, it will need to be uploaded to GitHub for everyone to download.

**Dates and member:** This task will be started at the start of week 8, which is 22/04/2024, and needs to be finished by the end of week 8, which will be 28/04/2024. The member assigned to this task is Henry.

**Task 3: Back-end PHP development**

**Information:** The website needs to use PHP to display and store the data. VSCode will be used to code the PHP. Not every page will need corresponding PHP code, but the following needs to be coded:

* Logon - store logon information
* Inventory - store and display amounts of medication and commodities
* Member and staff profiles - display member details
* Create a member - store member details

After the HTML code has been added to the pages the updated pages will be uploaded to GitHub for everyone to download.

**Dates and member:** This task will be started at the start of week 8, which is 22/04/2024, and needs to be finished by the end of week 8, which will be 28/04/2024. The member assigned to this task is Maxy.

**Task 4: Write information for the databases**

**Information:** Coming up with names and fake phone numbers or finding test data online will be needed in order to populate the website and test it. The following is needed for the different databases:

* Members - 4 sets of data for 4 different member profiles
* Staff - 4 sets of data for 2 different staff profiles and 2 different admin profiles
* Inventory - data for medication and food tracking
* Facilities - made up utility metrics

**Dates and member:** This task will be started at the start of week 8, which is 22/04/2024, and needs to be finished by the end of week 8, which will be 28/04/2024. The member assigned to this task is Qiao.

**Week 9 - 29/04/2024 to 05/05/2024**

**Task 1: Add the JavaScript**

**Information:** Some things for the front-end development need to have error messages as they involve data. JavaScript will be used to display these messages. If someone puts wrong information in, an error message saying that will be needed. The following need messages:

* Logon notification - message saying problem logging on if the logon isn’t right
* Inventory error notification - message saying there was duplicate data so the changes weren’t saved
* Create a member - message saying some fields are blank

The JavaScript code pages will need to be uploaded to the GitHub for everyone to download.

**Dates and member:** This task will be started at the start of week 9, which is 29/04/2024, and needs to be finished by the end of week 9, which will be 05/05/2024. The member assigned to this task is Dylan.

**Task 2: Add data to the databases**

**Information:** Now that the code is mainly finished, the website needs to be populated with information. This will come from the data for the databases that was collected in week 8. A member will go through the website and put information in the input fields. This will be done using WinSCP. The following notepads might need to be made to hold the data:

* Member profile
* Staff profile
* Facilities
* Inventory
* Create member

**Dates and member:** This task will be started at the start of week 9, which is 29/04/2024, and needs to be finished by the end of week 9, which will be 05/05/2024. The member assigned to this task is Rubie.

**Week 10 - 06/05/2024 to 12/05/2024**

**Task 1: Check for errors in the code**

**Information:** The PHP and JavaScript code, and possibly the databases, will have errors. These can be checked by running the code using WinSCP or online resources, such as PHP Code Checker. Once the errors have been fixed, the code will be uploaded to GitHub.

**Dates and member:** This task will be started at the start of week 10, which is 06/05/2024, and needs to be finished by the end of week 10, which will be 12/05/2024. The member assigned to this task is Rubie.

**Task 2: Review the website**

**Information:** The code should be complete by now, which means members will be able to navigate through the website and review it. If small changes need to be made, they will be made, but no big changes are to happen this late in the project.

**Dates and member:** This task will be started at the start after the errors are checked, which shouldn’t take the whole week. This task is to be completed by the end of week 10, which is 12/05/2024. The members assigned to this task are all of the team members.

1. **Tracking and Monitoring**

Tracking and monitoring project progress during the development of the aged care management system involves a strategic blend of tools and measures. Agile project management principles underpin our approach, fostering close collaboration among stakeholders, including developers, project managers, clients, and end-users.

Regular team tasks such as daily stand-ups, sprint reviews, and retrospectives ensure alignment and effective contribution from all team members. Tools such as Jira and Trello are important throughout the project lifecycle. Trello's visual task management system, represented by boards and cards, aligns with our team's preference for ease of use and task visualization. Meanwhile, Jira is used for its robust bug tracking and story management capabilities in an agile environment.

In the software development stage, GitHub serves as a cornerstone, facilitating safe concurrent work by multiple team members. Its version control and backup functionalities ensure the integrity of project data, enabling seamless collaboration and experimentation through branch creation.

Visual Studio Code (VS Code) enhances our development workflow significantly. Its compatible across multiple platforms and it supports JavaScript, Node.js, HTML, CSS natively with a vast ecosystem of extensions for other languages like C++, Python, and PHP. We can effortlessly switch between GitHub branches, create new ones, and merge them. Incorporating VS Code will significantly enhance coding efficiency, collaboration, and code quality management.

For communication and collaboration, MS Teams and Discord are the most convenient platforms. Instant messaging, file sharing, and real-time discussions on these platforms streamline workflows, reducing reliance on physical meetings and lengthy email threads. Features such as task assignments, status updates, and automated notifications boost team efficiency and productivity, particularly crucial in the context of remote work trends. With the rise of remote work, the importance of effective collaboration and documentation tools are more important than ever.

1. **Quality Plan**

In this quality plan, we outline our approach to maintaining high standards throughout the second sprint of our project. With a focus on detailed quality expectations and rigorous testing procedures, we aim to deliver products that meet or exceed stakeholder requirements and expectations.

**8.1. Target Quality Expectations**

1. Website UX evaluation: We aim for evaluation based on criteria like ease of navigation, clarity of information presentation, and user interaction intuitiveness. This will be assessed through end-user testing sessions where participants perform common tasks and provide feedback.
2. Responsive design implementation: We expect responsive design principles to dynamically adapt the layout and content to various screen sizes, ensuring optimal usability across devices.
3. Performance metrics: Our target is to maintain low performance metrics such as page load times, server response times, and resource utilization during HTTP requests and PHP file assessment.
4. Consistency with prototype design: We hope the web design will align with our prototype, defining design elements like colour schemes and UI components in a style guide for consistency.

**8.2. Steps to Assure Quality**

1. Code review: We will implement code reviews to evaluate the quality, readability, and maintainability of the HTML, CSS, JavaScript, PHP, and SQL code. Any issues will be addressed, including adding readable annotations and ensuring compatibility with Google Chrome and server environments.
2. Validation testing: Validation testing on HTML pages will be performed to ensure compliance with W3C standards and identify any markup errors or syntax issues.
3. Functional testing: Interactive elements such as forms, buttons, and navigation menus will undergo functional testing to verify proper functionality.
4. Device and screen size testing: The website will be tested on various devices and screen sizes to validate responsiveness and ensure consistent presentation across different platforms.
5. Usability testing: Feedback will be gathered from 3-5 end-users through usability testing sessions to assess ease of use, intuitiveness, and overall user satisfaction.