EXHIBIT: A Tool for Digital Presentations of Cultural Heritage Artefacts

Aa'isha Dout
dtxaai001@myuct.ac.za
University of Cape Town

Ceara Mullins
mllcea001@myuct.ac.za
University of Cape Town

1 PROJECT DESCRIPTION

Digital archives— collections of multimedia objects — are ideal for preserving and valourising cultural heritage. Web portals allow these collections of cultural heritage artefacts, digitised in the form of images, videos, audio, etc., to be accessed worldwide where previously this would not have been physically possible. In a South African context, projects like the Bleek and Lloyd collection [2] and the 500 Year Archive [1] allow academics and the general public to browse artefacts related to the language, culture and history of indigenous people in Southern Africa.

Despite their value to the public, collections are not always easy to navigate [11] and there is a need for more complex, narrative-driven presentations of collection content than just the search-and-display features available for most archives. To generate new perspectives on cultural heritage and keep interest in the archives alive, digital presentation tools which allow users to create their own digital exhibitions of collection content have become crucial.

Existing exhibition tools are numerous and make use of varied technology. For example, 3D graphics technology has become popular amongst some galleries and museums as a way to create faithful replicas of physical exhibitions [5]. However, this is overkill for the student user group, which we have found gravitates to tools such as PowerPoint and Google Slides for their ease of use and collaborative features. There is a need for a tool simple enough for those with limited IT expertise to use that at the same time adheres to metadata standards for use by cultural heritage experts to display archive content. The solution is a direct-manipulation tool that outputs exhibitions of collection content in widely used and understood formats such as PDF files and HTML pages.

2 PROBLEM STATEMENT

Although they contain rich cultural heritage information, many archives do not provide an accessible way to interpret these contents, and often provide only searchable lists of contents as presentation to the public [26]. Digital exhibitions address this by allowing an exhibition creator to define a narrative or logical order to organise a collection of items. However, we have found that current software tools do not cater to wide user bases: tools which faithfully recreate physical exhibitions are difficult to use and overkill for most (high school) educational purposes. Tools popular in the school system, such as PowerPoint and Google Slides, offer useful collaborative features but are more suited to working with Web and user resources than collection items as they do not adhere to metadata standards and are generally not built with galleries, libraries, archives, and museums (the GLAM sector) in mind.

The aim of this project is to design and implement a software tool that can be used by those with limited or basic IT expertise to create exhibitions of cultural heritage artefacts from existing collections and relevant Web materials.

2.1 Requirements

Potential users of the exhibition tool are researchers, historians, students and interested members of the general public. An initial requirements-gathering stage revealed the following basic requirements.

The system must allow users to create digital exhibitions of archive content that are interactive, modifiable and shareable. Users must be able to import materials from collections and external sources and arrange them in a logical fashion. In addition, the system will provide tools to modify or define themes, including layout, colour palette and font.

The system will allow exhibitions to be saved to a collection of user-created exhibitions and also published in the form of PDF files and HTML pages. Versions saved to the database can be modified.

Above all, the system functionality must be independent of a single archive, and allow the client to implement it in the context of new archives and collections without creating a new solution.

2.2 Project Scope

The scope of the project does not include digitisation and preservation of cultural heritage artefacts. Storage, management and information access services are also already implemented and the exhibition tool will be built on top of such archives.

2.3 Stakeholders

2.3.1 *Project Supervisor.* Hussein Suleman

Responsibility: Managing the team members and assessing the standard and quality of all project deliverables. Attending a weekly meeting with the team members to discuss the direction of the project and progress.

2.3.2 *Project Team.*

- Aa'isha Dout (Project Communicator)

  Responsibility: Must communicate with team members, supervisor and other stakeholders. Attends the weekly meeting with supervisor and team member.

- Ceara Mullins (Project Leader)

  Responsibility: Ensures the team members are adhering to the project schedule and meeting the appropriate deadlines. Attends the weekly meeting with supervisor and team member.

2.3.3 Clients and Users

. **Clients**

- Carolyn Hamilton, Rifqah Kahn, Debra Pryor, Grant McNulty, Benathi Marufu, Chloe Rush, Henry Fagan, Ettore Morelli (UCT APC) - Five Hundred Year Archive
- Pippa Skotnes, Niek de Greef (UCT CCA) - Bleek and Lloyd Collection
- Stefania Merlo (Wits), Justine Wintjes (KZNM), Anton Coetzee (Wits), Serena Coetzee (UP), Victoria Rautenbach (UP) - Metsemegologolo
- Azizo Dafonseca (Wits) - South African Rock Art Digital Archive

Users

- Museum Curators
- High school students
- Academic researchers
- Communities with interest in South African history

3 PROCEDURES AND METHODS

3.1 Development Features

EXHIBIT will have a layered architecture as displayed in Figure 1.

3.1.1 Interface Layer. EXHIBIT will have a browser user interface (BUI) to allow users to access the services of the system.

3.1.2 Services Layer. The service layer connects the front-end interface to objects in the database that users will incorporate into exhibitions, be they collection content or user-uploaded content. Table 1 on page 3 contains more detailed descriptions of the client services to be incorporated and Figure 2 shows how these components will be integrated into the system.

3.1.3 Back-end Layer. This layer involves the storage and organisation of the digital objects, including collection content from an archive, user-uploaded content and the exhibitions created by EXHIBIT users.

3.2 Development Platform

EXHIBIT will be designed as a Web tool compatible with most current Web browsers, such as Chrome, Internet Explorer and Safari. The interface will adhere to responsive design principles to allow access via mobile devices. The user interface will be designed using HTML and Tailwind CSS, a front-end framework that allows for creation of responsive interfaces [3].

In terms of client services, exhibition editing, template creation, publishing and viewing will be self-implemented. Most notably, the

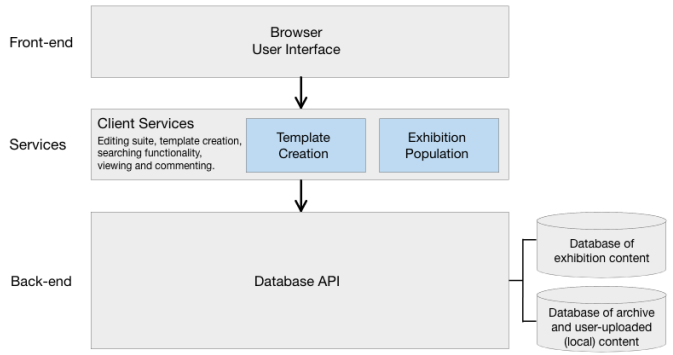

Figure 1: EXHIBIT architecture overview

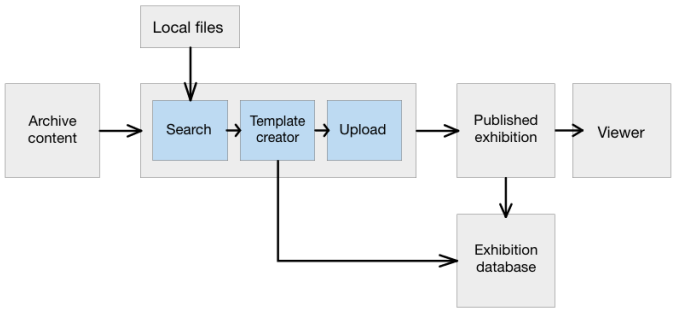

Figure 2: EXHIBIT system flow

drag-and-drop template creator will use JavaScript to allow manipulation of the HTML elements. Other services will be implemented using third-party software. Archive content search functionality will be basic and will likely be an implementation of a search feature developed by our supervisor in a related project. However, Solr [14] and Elasticsearch [8] are open source options being explored as an alternative service.

Exhibitions will be stored in a flat-file database with XML-formatted data. This is to bypass setup complexity of a database management system. The repository-layer functions and Web server functionality will be implemented using the Python programming language. The Python standard library provides modules for XML processing [13] and the lxml toolkit [27] is a Pythonic binding for C XML libaries that facilitates editing and saving. The Python CGI module[12] will be used on the server machine to generate and manage the website's content.

3.3 Implementation Strategies

Our project will follow an iterative design procedure to account for changing requirements. The Agile methodology [23] is ideal for the short project duration and our small team. This is because of the emphasis on team development and adapting to changing circumstances. Agile promotes working and communicating within development teams to help ensure a successful product is shippable at the end of the development cycle. Given the current lockdown

Table 1: Client Services

Service Name	Description
Search	Basic search functionality will be provided to allow users to select relevant items from the collection to use in their exhibitions.
Upload	Users will also be allowed to upload local files to be used in their presentations.
Template Creation	Users are presented with a drag-and-drop interface to create the template for their exhibition. The template includes layout and theme (colours and fonts). They may also skip this step and use default templates.
Exhibition Creation/ Instantiation	Users may populate the templates they created with content from the aforementioned sources and also incorporate external Web content through linked annotations. They may also return to saved exhibition templates and completed exhibitions to make edits.
Publishing	Users may publish and download their exhibitions in the form of a PDF file or HTML page.
Viewing	A carousel of published exhibitions (with their titles, descriptions and exhibition author/s displayed) will be displayed for the user to view exhibitions that interest them. Users can search for exhibitions by title and keywords.
Low priority services	
Access Privileges	Users may define viewing privileges for their exhibitions (public or private) and also access privileges to resources they upload themselves.
Comments	Users will be able to leave comments on exhibitions they have viewed, which will then appear with exhibition overview data in the viewing interface.

and COVID-19 pandemic, the changing stages make adaptability important to the success of the project. Since usability of our tool is a key consideration, we will also adhere to user-centred design principles. We acknowledge that engaging with clients will be subject to the situation of the country and will rely on user personas when necessary. These benefits, along with others, such as frequent delivery and efficiency [28], will help ensure the smooth development of the project according to the schedule.

Along with the Agile methodology, we will be using the Scrum framework [25]. Scrum is a lightweight framework designed to be an iterative approach to developing whilst promoting the needs of the development team and fostering communication [19]. The Scrum framework will be managed with daily Scrum meetings via video conferencing software in order to discuss the progress of development, the responsibilities of the development team and to update the product backlog. We plan for our sprints to be around 1 week long, as this is the length of development time we have set aside for major features. This meeting is merely to assess the situation and how to proceed; it is a 10-15 minute meeting and is not used for solving development issues. Weekly meetings with the project supervisor will continue and ensure that stakeholders are updated on the project status. Scrum retrospectives [19] will help the development team to remain on schedule and guarantee that the quality of the code being developed is of a high enough standard. Continuous integration practices will be used, in keeping with Agile and Scrum principles, to support the development team and ensure that at the end of each iteration there is a "shippable" product.

Implementation will be done in 3 phases, with each addressing different sections of the software.

3.3.1 Implementation Phase 1. The first implementation phase consists of installing the system foundations and setting up the database, including basic database interaction operations. The requirements gathering phase will be conducted in parallel with this initial development phase. Requirements gathering will allow for user feedback about what this kind of a tool would require in order to be successful. Due to the limited schedule for development, the requirements gathering phase will need to be completed early in the project life cycle. First implementation of client services will also be done in this stage. Initial development will proceed while waiting for ethical clearance to conduct user testing. Once approved, usability testing will proceed. A write-up of system and user tests of the initial prototype will be produced at the end of this stage.

3.3.2 Implementation Phase 2. The second phase will address critical concerns from testing and feedback from the initial feasibility assessment. Low priority features – commenting and collaboration – will be implemented at this stage.

3.3.3 Implementation Phase 3. Refinements to the software will be made in this stage, including finalisation of the graphical user interface. We will begin work on the project website.

Refer to Figure 3 on page 8 for a timeline of these implementation stages.

3.4 Expected Challenges

We have not yet received archive content on which to build the tool. Although the EXHIBIT project is intended to function as part of any collection, we anticipate that interacting with this test data will inform how the project will be developed and correct misconceptions we may have had in the design stage.

Communication will be an issue in these times as physical meetings are not possible. It is likely that not everyone involved in the project will have reliable access to the Internet, which will be a concern during development for team communication and also testing, as users may not be able to test the system remotely and provide feedback.

A broad challenge of this project will be the EXHIBIT system fulfilling our value proposition, as initial contact with potential users has revealed that they currently use well-established and well-received tools — both those developed for this domain, as mentioned in the Related Work section, and makeshift tools like PowerPoint and Google slides — to create exhibitions, which we will likely struggle to improve upon given our skill level.

3.5 Evaluation

3.5.1 Software Testing. Besides unit testing, which will be done at all stages of the coding process, we will be conducting integration tests to ensure that the separately-developed parts of our system work in conjunction as expected. Once the prototype is completed, we will be conducting system performance tests such as load and stress testing, by simulating access to the hosted prototype.

3.5.2 User Testing. As stated previously, usability of the final product is a key consideration of our project. User testing with the target audiences is therefore essential. Although we cannot meet with evaluators physically, we plan to conduct surveys online using questions from established usability frameworks such as the System Usability Scale [6], Computer System Usability Questionnaire (CSUQ) [18], the Questionnaire for User Interaction Satisfaction (QUIS) [10] and custom Likert scales to answer usability questions more specific to EXHIBIT. The qualitative metrics to be assessed will include how easy the users found the system to use, how learnable the system is and how confident they felt using the system, etc. We will also review the exhibitions created during user testing and determine the average length of the exhibition, the depth of detail and range of content of exhibitions to get an idea of how comprehensively the tool is being used. There are some studies on existing exhibition tools that contain similar assessments [15] to which we can compare these quantitative results. We would also like to conduct interviews via video-conferencing tools to receive feedback, but we recognise that this may be difficult due to Internet access and connection variability.

We will recruit participants via contacts provided by the institutions mentioned in 2.3.3 that are representative of the groups listed in the 'users' section of 2.3.3. We hope to recruit at least 3 participants per group for online surveys. Traditional usability studies regard 5 participants as the minimum number for legitimacy of results [24], and a minimum of 12 participants will give us a broader range of opinions and help us identify more issues with our system [17].

4 ETHICAL, PROFESSIONAL AND LEGAL ISSUES

Collection content will be provided by the UCT Archive & Public Culture (APC) research initiative. Terms of use will be communicated to the user. We anticipate some issues with the uploading of user content, as we cannot ensure that the multimedia objects they upload will be appropriate and of high quality. We therefore plan to incorporate prominent labels in exhibition templates that indicate the source of each item.

Prior to user testing, we will obtain ethical clearance from the UCT Human Research Ethics Committee. Aims of the study and project as a whole will be explained to all participants. Participants will be informed that they may withdraw from the study at any point and may remain anonymous in any publication of the research. Consent will be obtained before any recording of interviews.

The final product will be the intellectual property of the project team and the University of Cape Town. All software will be open source and available for download via the project website.

5 RELATED WORK

This section looks at software with similar functionality to the EXHIBIT project. An overview of the design decisions and services of the tools is explored for the development of the EXHIBIT tool.

5.1 Template Creation

Drag-and-drop template creation is common in the sphere of website building software. Tools such as bubble [7] and Carrd [9] allow users with no coding experience to control the user interface component of their sites via widgets and menus.

In the domain of digital exhibitions, VAES [29] is a digital exhibition creation tool that implements a direct-manipulation, drag-and-drop interface for users to create their own individual and unique exhibitions from uploaded content. The exhibitions are then converted into HTML viewing pages through the use of XML Extended Style Sheets (XSL) and Cascading Style Sheets (CSS).

EXHIBIT's template creation software would conform to these principles of interaction, providing a tool to place HTML components as desired. Users will also be able to change fonts and colours through menus.

5.2 Exhibition Population and Viewing

Exhibition population requires a considerable sorting and searching mechanism to manage the archive material. The MOVIO toolkit [22] implemented a dedicated semantic content management system (SCMS) for the purpose of organising and retrieving archival material. The SCMS is an abstraction of the database to allow users to interact with the material and incorporate content into their exhibitions [22]. MOVIO's CMS is also capable of importing local content which can be used to populate user exhibitions [21]. This tool is called the "module builder" which allows curators to import existing databases and other external material.

Both the MOVIO tool [20] and Omeka [16] have support for mobile viewing of exhibitions. The EXHIBIT project is intended to be widely accessible and viewing exhibitions on a mobile device significantly increases the accessibility of the tool to a wide variety of users.

Viewing formats such as HTML and PDF account for users that have limited IT expertise and users that would prefer a more interactive exhibition viewing experience. Interactive viewing tools, such as a carousel and linking, could enhance the experience of a digital exhibition by creating a progression through the exhibition. This logical progression is seen in the story-builder tool of the MOVIO toolkit [21].

6 ANTICIPATED OUTCOMES

6.1 System

The unique software features to be developed in this project are outlined below.

6.1.1 Template Creator. This drag-and-drop tool will allow users with no or limited knowledge of HTML and CSS to define their own template layouts and themes via dropdown menus and widgets. It

will also allow a more experienced user to edit the files themselves for more customisability and flexibility.

6.1.2 Exhibition Creator. This tool will allow users to upload cultural heritage artefacts, their own content and external Web materials to their template to create a digital exhibition. They can also define an order between exhibition items for viewers to navigate the exhibition. Users may publish their exhibitions in the form of HTML pages and/or PDF files.

6.1.3 Exhibition Viewer. This tool will allow users to explore exhibitions others have created. Users can view existing exhibitions by searching for titles or browsing keywords, and can leave feedback in the form of comments.

6.2 Impact

We hope to create a tool that will allow users of varying technical backgrounds to explore cultural heritage artefacts and collections. By allowing users to create their own narratives through digital presentations, we hope to stimulate interest in cultural heritage and make the field accessible to a wider audience.

6.3 Key success factors

The success of the project is based on the ability of users from varying backgrounds and technical expertise to create digital exhibitions of cultural heritage artefacts from existing collections and their own sources, easily. Software success factors to determine whether we have achieved this goal are outlined below.

Key success factors for the template creation tool:

- The ability to create an exhibition layout by using a drag-and-drop system to place components
- The ability to change colours and fonts via menus or editing of the CSS files

Key success factors for the exhibition creation tool:

- The ability to populate an exhibition with content from the archive or local files
- The ability to define an order in which a viewer can explore the exhibition content
- The ability to publish an exhibition in the form of an HTML page and/or PDF file

Key success factors for the exhibition viewer:

- The ability to browse all exhibitions created by users
- The ability to search for exhibitions by keyword

The success of the cultural heritage preservation aspect of this project will be judged based on continued use of the tool, i.e if users are creating, viewing and leaving comments on exhibitions after the project has concluded. We will measure this by monitoring site access and number of exhibitions published.

7 PROJECT PLAN

The following section details the project development plan. This includes identification of work allocation, potential risks, resources required, deliverables, milestones and work allocation.

7.1 Work Allocation

Table 2 summarises the work allocation plan. Tasks necessary for the implementation of other features will be addressed first. The table indicates at what stage features will be developed and which team member will be responsible for them. Broadly, Aa'isha will be concerned with exhibition template creation and editing and Ceara with exhibition implementation/instantiation, mainly importing content into the templates.

Table 2: Work Allocation

Iteration	Aa'isha	Ceara
1	-System foundations and database population - Upload feature: local content - Template editor	-System foundations and database population - Upload feature: archive content - Template population
2	Collaborative features	Commenting and annotations
3	Refine services	

7.2 Resources Required

The project requires various software tools to develop the system, users to test the system and content to populate the database. No special equipment is required beyond personal computers for development. Collection content will be made available by the previously mentioned stakeholders. User groups we have identified for testing the system include researchers, curators, high school students and the general public.

The website will be hosted using a free tier of Amazon Web Services (AWS) hosting [4]. This is to minimise site access issues during periods of loadshedding, as AWS has overseas servers [4]. UCT servers may be used as a backup.

7.3 Deliverables

- Initial Feasibility Demonstration
- First Implementation Write-up
- Final Prototype Write-up
- User Evaluation/Testing Report
- Final Complete Draft of Report
- Project Report Final Submission
- Project Poster
- Project Website

7.4 Milestones

The milestones listed can be located on the Gantt chart on page 8 and include official honours project deliverables and milestones we have set for ourselves.

- First prototype 31/7/20
- Initial Feasibility Demonstration 03/08/20
- Initial Testing 12/8/20
- Final Prototype 30/8/20
- Final Complete Draft of Report 11/9/20
- Project Report Final Submission 21/09/20
- Final Implementation 25/9/20

7.5 Risks

Table 3 on page 7 summarises the risks identified for this project, separated into categories and ranked according to their impact and probability of occurrence.

Table 3: Risk Matrix

Risk	Consequence	Probability	Impact	Mitigation	Monitoring	Management
Development						
Team member becomes unavailable	Delays and schedule adjustments and the remaining member's workload is increased	Medium	Critical	Clear division of work, clear documentation	Regular team meetings to assess how team members are coping with workload	Reduce scope if possible
Incorrect prioritisation of tasks	Not all features will be satisfactorily completed	Medium	Critical	Verify scope with supervisor and re-prioritise tasks if necessary	Regular check-ins with supervisor	Reduce scope if possible
Lack of access to the Internet due to loadshedding	Members may have issues accessing online resources during development and users may have issues accessing the system for testing	High	Marginal	Team members will download critical resources, such as third-party software documentation, as they are used. They will also ensure that they have multiple data sources (e.g. mobile data in addition to WiFi). Usability tests will be scheduled with the loadshedding timetable in mind	Check loadshedding schedule regularly and update plans accordingly	Team members will switch to alternative data sources. The team will reschedule usability tests to times when users are not experiencing loadshedding
Scheduling/Time Management						
Deadlines not met	Project features not delivered in time	Medium	Critical	Timelines have leeway to account for unforeseen issues	Project management tools to keep track of work rate	Adjust project schedule with remaining leeway days, assess project scope and prioritise essential features
Too much time spent deciding on third-party software to use for the system	Development time impacted	Low	Critical	Devise and adhere to timeline which has deadlines for these decisions. Have a default option if the team does not agree on software choices	Regular team meetings to assess if schedule is adhered to	Implement the default software option
Communication						
Team and client communication issues due to exclusively online contact	Issues with software: bugs, misunderstanding of requirements, etc.	Medium	Critical	Define a number of communication channels, include video-conferencing and less data-consuming options such as email and SMS, to communicate effectively if one breaks down. Make extensive use of Github for version control and software management	Check before meetings if everyone has access to the required platform	Change communication channels and if possible, reschedule meetings

7.6 Project Timeline

Figure 3 shows a timeline of the project, including the implementation phases, tasks, deliverables and deadlines.

TASK TITLE	START DATE	DUE DATE	DURATION
Project Conception and Initiation			
Initial Project Proposal	12/5/20	1/6/20	20
Initial Project Proposal Due	2/6/20	2/6/20	N/A
Ethics Application Writing	2/6/20	5/6/20	4
Feedback on Project Proposal Due	18/6/20	18/6/20	N/A
Revisions to Project Proposal	18/6/20	28/6/20	11
Revised Project Proposal Due	29/6/20	29/6/20	N/A
Project Development			
Prototyping and Brainstorming	1/7/20	6/7/20	6
Final Paper (Background)	6/7/20	10/7/20	5
Final Paper (Design)	6/7/20	10/7/20	5
Implementation Phase 1	7/7/20	31/7/20	25
Uploading Test Data	7/7/20	10/7/20	4
Search Functionality (possibly use external)	13/7/20	17/7/20	5
Requirements Gathering	14/7/20	17/7/20	3
Template Creator	13/7/20	24/7/20	12
Uploading/Population Functionality	15/7/20	24/7/20	10
Exhbition Publishing	21/7/20	25/7/20	5
Viewing Interface	27/7/20	31/7/20	5
Initial Feasibility Demonstration	3/8/20	11/8/20	9
System Testing	12/8/20	14/8/20	3
Usability Testing	3/8/20	10/8/20	8
Project Paper (Draft 1)	11/8/20	17/8/20	7
Implementation Phase 2	18/8/20	2/9/20	15
Critical Changes	18/8/20	24/8/20	7
Commenting	25/8/20	27/8/20	3
Collaboration (editing permissions)	25/8/20	27/8/20	3
System Testing	28/8/20	30/8/20	3
Usability Testing	28/8/20	30/8/20	3
Implementation Phase 3	1/9/20	4/9/20	4
Project Paper (Draft 2)	1/9/20	10/9/20	10
Project Paper Draft Due	11/9/20	11/9/20	N/A
Project Paper Final Draft	14/9/20	20/9/20	7
Final Project Paper Due	21/9/20	21/9/20	N/A
Project Code Final Submission	25/9/20	25/9/20	N/A
Final Project Demonstration	5/10/20	9/10/20	5
Project Media			
Project Poster	1/10/20	11/10/20	11
Project Website	25/9/20	18/10/20	24
Project Poster Due	12/10/20	12/10/20	N/A
Project Website Due	19/10/20	19/10/20	N/A

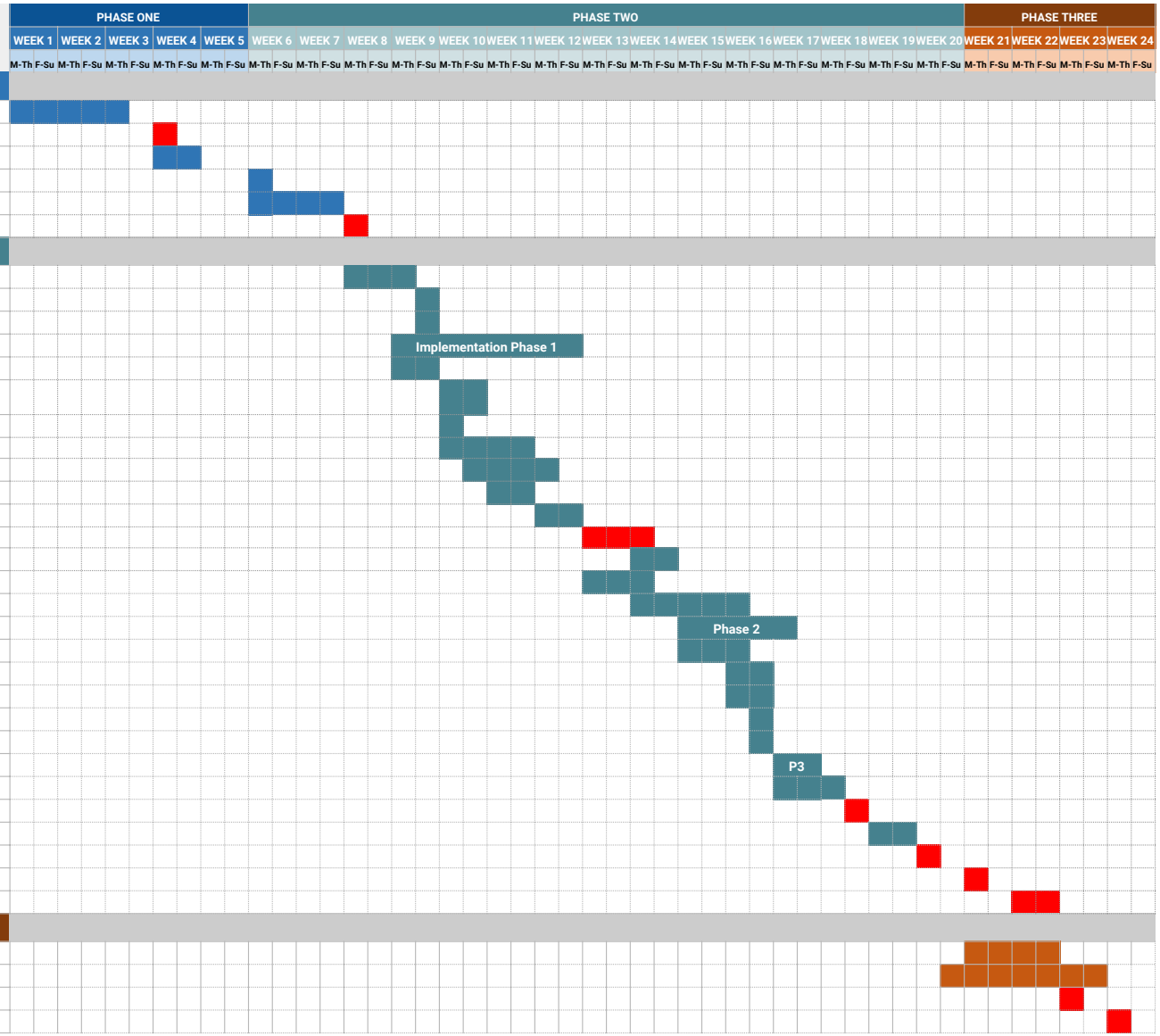

Figure 3: Project Gantt chart

REFERENCES

[1] [n.d.]. *The Five Hundred Year Archive.* Retrieved May 16, 2020 from http://www.apc.uct.ac.za/apc/research/projects/five-hundred-year-archive

[2] [n.d.]. *THE DIGITAL BLEEK AND LLOYD.* Retrieved May 16, 2020 from http://lloydbleekcollection.cs.uct.ac.za/

[3] 2020. *Tailwind CSS: A utility-first CSS framework for rapidly building custom designs.* Retrieved May 16, 2020 from https://tailwindcss.com/

[4] Inc. Amazon Web Services. 2020. *Web Hosting.* Retrieved June 02, 2020 from https://aws.amazon.com/websites/

[5] Daniel Biella, Wolfram Luther, and Nelson Baloian. 2010. Beyond the ARCO standard. In *Proceedings of the 16th International Conference on Virtual Systems and Multimedia*. IEEE, 184–191. https://ieeexplore.ieee.org/document/5665985

[6] John Brooke. 1986. System usability scale (SUS): A quick-and-dirty method of system evaluation user information. In *Usability Evaluation In Industry*. Taylor & Francis, Reading, UK, 1–7.

[7] Inc. Bubble Group. 2020. *Getting started.* Retrieved May 20, 2020 from https://bubble.io/documentation

[8] Elasticsearch B.V. 2020. *Elastic Stack and Product Documentation.* Retrieved May 16, 2020 from https://www.elastic.co/guide/index.html

[9] Carrd. 2020. *Documentation: General.* Retrieved May 20, 2020 from https://carrd.co/docs/general

[10] John P Chin, Virginia A Diehl, and Kent L Norman. 1988. Development of an instrument measuring user satisfaction of the human-computer interface. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 213–218.

[11] Gobinda Chowdhury. 2009. From digital libraries to digital preservation research: the importance of users and context. *Journal of Documentation* 66, 2 (July 2009), 207–223. https://tefkos.comminfo.rutgers.edu/Courses/Zadar/Readings/Chowdhury%20dig%20pres%20J%20Doc%202009.pdf

[12] Python Software Foundation. 2020. *cgi — Common Gateway Interface support.* Retrieved July 9, 2020 from https://docs.python.org/2/library/cgi.html

[13] Python Software Foundation. 2020. *XML Processing Modules.* Retrieved June 03, 2020 from https://docs.python.org/3/library/xml.html

[14] The Apache Software Foundation. 2020. *Apache Solr.* Retrieved May 16, 2020 from https://lucene.apache.org/solr/

[15] Paula Goodale, Mark Hall, Kate Fernie MDR, Phil Archer, Francesca Tavanti, and Paul Clough. 2011. D 1.1 User Requirements Analysis. *PATHS Project* (2011).

[16] Juliet Hardesty. 2014. Exhibiting library collections online: Omeka in context. *New Library World* 115, 3 (March 2014), 75–86. http://dx.doi.org/10.1108/NLW-01-2014-0013

[17] Ritch Macefield Janet Six. 2016. *How to Determine the Right Number of Participants for Usability Studies.* Retrieved May 12, 2020 from https://www.uxmatters.com/mt/archives/2016/01/how-to-determine-the-right-number-of-participants-for-usability-studies.php

[18] James R Lewis. 1995. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction* 7, 1 (1995), 57–78.

[19] Jim Medlock. 2017. *A Short Introduction to the Scrum Methodology.* Retrieved June 01, 2020 from https://medium.com/chingu/a-short-introduction-to-the-scrum-methodology-7a23431b9f17

[20] Sam Habibi Minelli, Iva Meštrović, Petra Milovac, Orsolya Veress, Donatas Snarskis, Jovita Vilimaitiene, Linnéa Karlberg Lundin, Karin Glasemann, Piot Kożurno, and Marek Wieczorek. 2015. Museums' Experiences in Creating Cultural Narrations Using the AthenaPlus Tool Called MOVIO. *Uncommon Culture* 6, 11 (June 2015), 67–87. http://uncommonculture.org/ojs/index.php/UC/article/viewFile/6086/4635

[21] Sam Habibi Minelli, Maria Teresa Natale, Barbara Dierickx, Paolo Ongaro, Daniele Ugoletti, Rubino Saccoccio, and Marc Aguilar Santiago. 2014. MOVIO: A semantic content management and valorisation approach for archives and cultural institutions. In *Arxius i Indústries Culturals. Girona, de l'11 al 15 d'octubre de 2014. 2a Conferencia Anual d'Arxius· 9a Conferencia Europea d'Arxius*. International Council on Archives. https://www.girona.cat/web/ica2014/ponents/textos/id234.pdf

[22] Sam Habibi Minelli, Maria Teresa Natale, Paolo Ongaro, Marzia Piccininno, Rubino Saccoccio, and Daniele Ugoletti. 2014. MOVIO: A toolkit for creating curated digital exhibitions. *Procedia Computer Science* 38 (January 2014), 28–33. https://www.sciencedirect.com/science/article/pii/S1877050914013660/pdf?md5=ef95993470fa0f6eb980ae581c5d5bce&pid=1-s2.0-S1877050914013660-main.pdf&_valck=1

[23] Dinnie Muslihat. 2020. *Agile Methodology: An Overview.* Retrieved May 22, 2020 from https://zenkit.com/en/blog/agile-methodology-an-overview/

[24] Jakob Nielsen. 2012. *How Many Test Users in a Usability Study?* Retrieved May 12, 2020 from https://www.nngroup.com/articles/how-many-test-users/#:~:text=If%20you%20want%20a%20single,using%20many%20more%20test%20participants.

[25] Scrum.org. 2020. *What is Scrum?* Retrieved June 01, 2020 from https://www.scrum.org/resources/what-is-scrum

[26] David Silver. 1997. Review: Interfacing American Culture: The Perils and Potentials of Virtual Exhibitions. *Bibliothek Forschung und Praxis* 49, 4 (1997), 825–850. https://muse.jhu.edu/article/175548

[27] Ian Bicking Stefan Behnel, Martijn Faassen. 2020. *lxml - XML and HTML with Python.* Retrieved June 02, 2020 from https://aws.amazon.com/websites/

[28] Jon Terry. 2019. *Benefits of Agile Development.* Retrieved May 22, 2020 from https://www.planview.com/resources/articles/benefits-of-agile-development/

[29] Ruan Yang, Chennupati K Ramaiah, and Schubert Foo. 2007. Virtual archival exhibition system: An authoring tool for developing web-based virtual exhibitions. In *International Conference on Dublin Core and Metadata Applications*. Dublin Core Metadata Initiative (DCMI), 96–105. http://dcpapers.dublincore.org/pubs/article/viewFile/870/866