G5

PROJECT DOCUMENTATION

Master Test Plan

Aerial Imagery Initiative

Version	Date	Remarks	Author
1.0	3-5-2021	Initial Version	D. Sheehan

Index

Index 1

Management Summary 2
- Project Objective 2
- Test objective and assignment 2
- Short description of the test approach 3
- Results to be realised 3
- Qualitative objectives 3
- Go/no-go decisions 4

Introduction 4
- Project and project objective 4
- Objective of the Master Test Plan 4
- Involved in creating the Master Test Plan 5

2. Assignment Formulation 5
- **Client** 5
- **Supplier** 5
- **Assignment** 5
- Scope 5
- Within scope 5
- Out of scope 6
- Preconditions and assumptions 6
- Reviewers and acceptance criteria 6
- Reviewers 6
- Acceptance criteria 7

3. Documentation 7
- Basis for the master test plan 7

4. Test Strategy 8
- **Product risk analysis** 8
- **Test Strategy** 9

5. Approach 11
- Test levels 11
- Deliverables 12
- Go / No go 12

7. Infrastructure 13

Test environment 13

8. Management 14

Test process management 14

Test infrastructure management 15

Defects procedure 16

9. Test process risks and countermeasures 17

10. Planning 18

11. References 20

G5 2

Management Summary

Project Objective

To develop a machine learning platform that can aid in the rapid response or recovery from floods. The system will use the SageMarker Machine Learning tool on the Amazon Web Services cloud infrastructure in combination with other open source libraries such as OpenCV to process flood imagery which is provided in JPEG format to identify and mark flood extent.

The project is being developed with 3 main parties involved: The Department of Spatial Services NSW, Intellify and Charles Sturt University Group 5. As such the approach to the project and consequent testing is fluid in nature and will change over time.

Test objective and assignment

The objective of the testing this project is to ensure the standards described in the Initial Requirements Model document, and subsequent variations of this are present within the software solution. A test manager will be allocated from Group 5 within the project to oversee testing, defect resolution, and reporting and may recruit others within Group 5 to assist with testing as required.

Short description of the test approach

The test approach will be risk based, with high risk aspects of the project tested first in an attempt to identify large issues early in the testing and provide the best chance of issue resolution at lowest cost.

Testing will be conducted over a number of levels including Static Review, Unit Tests, Integration Tests, Functional Acceptance Tests, User Acceptance Tests and System Tests at several milestones during the project.

Results to be realised

The test aspect of the project will produce the following deliverables which will aid further improvement and decisions for the project:

- Master Test Plan.
- Specification of Test Environment.
- Detailed Test Plan for each Test Level.
- Test Results Report.
- Defect Report.

Qualitative objectives

Testing will be based on the following objectives:

- Identify and test large risks early to reduce cost of defects.
- Complete testing for each stage within the Unified Process phases as described within this document.
- Document and resolve defects as required prior to the next testing window where possible.
- Provide test documentation within the Unified Process phases as described within this document.

Testing will be conducted across the following systems:

- Amazon Sagemaker
    - Object Tagging and Identification.
    - Jupyter notebook Instances.
- Storage and transfer of data within Amazon Web Services, including S3 buckets.
- Processing of data using other tools and open source libraries such as OpenCV.
- Code written using the Python Language.
- Other code or devices invented for the project.

Go/no-go decisions

Based on any defects found and subsequent review by Group 5 and DCS a Go / No Go decision will be made. Only defects major in nature which are determined to be unresolvable may pose a risk to the project go-live.

1. Introduction

a. Project and project objective

To develop a machine learning platform that can aid in the rapid response or recovery from floods. Utilising aerial photography provided by DCS the solution should identify and mark flood peaks.

The system will use the Ground Truth data labeling services within the SageMarker Machine Learning tool on the Amazon Web Services cloud infrastructure to process flood imagery which is provided in JPEG format. In addition other open source utilities such as OpenCV may be utilised to identify and mark flood extent data.

Ground Truth has the ability to identify and label items based on a model or previous manual identification to train a model. Unidentified objects can optionally be flagged for manual identification to be added to the model.

While the environment will be created by Intellify, It is anticipated the project will include research of approaches and techniques, review of data, the loading of images and proof of concept of the model in the first instance. If time permits development of an API to access the model will be explored.

b. Objective of the Master Test Plan

The objective of the Master Test Plan (MTP) is to inform all who are involved about the approach, the activities, and the (end) products to be delivered for testing the project.

Based on this detailed test plans will be developed for each test level with reference to the details in this Master Test Plan.

c. Involved in creating the Master Test Plan

The Master Test Plan is a document which will be continually reviewed and updated through the course of the project as more information is provided.

Name	Function	Responsibility
Darren Sheehan	Group 5 Team	Write Initial Version of MTP
		Review MTP

2. Assignment Formulation

a. Client

The client for the project is the department of Spatial Services, a division of the NSW Department of Customer Service. Project details and requirements are provided by this division for which Dr David Tien is our liaison.

b. Supplier

As the supplier, Group 5 will formulate test cases to ensure techniques and approaches used are appropriate to meet requirements set by the Spatial Services Division.

c. Assignment

The management of the testing will be overseen by the allocated Test Manager. The Test Manager will invite other members of Group 5 to assist as required to conduct testing, assist with defect resolution and documentation.

d. Scope

i. Within scope

- Test the system against functional requirements set out by DCS.
- Testing within the following systems:
    - Amazon Sagemaker
        - Object Tagging and Identification
        - Jupyter notebook Instance

- Storage and transfer of data using Amazon S3 buckets.
- Processing of data using other tools and open source libraries such as OpenCV.
- Python Language Code
- Other implemented code such as Amazon Lambda.

ii. Out of scope

- Testing of front end services.
- Testing in an environment outside of Amazon Web Services.
- Testing after the end of Semester 2, 2021.

e. Preconditions and assumptions

- Amazon SageMaker using Jupyter Notebooks will be appropriate for a test environment.
- Appropriate images are loaded into the system.
- A machine learning model has been developed or in the process of development or a suitable image processing solution has been developed.
- Testing can be completed within time allocated in the Project Plan.
- Resources are available to complete testing when required.

f. Reviewers and acceptance criteria

i. Reviewers

The people listed here are reviewers of the test system and test results:

Name	Function
Darren Sheehan	Group 5 Team
Adam Blewitt	Group 5 Team
Andrew Smith	Group 5 Team
Patrick Funnell	Group 5 Team
Cameron Nyberg	Group 5 Team

ii. Acceptance criteria

To meet initial requirements provided by The Department of Spatial Services, the following objectives must be met.

Objectives	Acceptance Criteria
Ability to upload data to the service.	Achieved by successfully uploading images to the service and the images can be accessed by internal services.
Able to process aerial imagery.	Internal services can interpret and process details within images, proven by highlighting elements.
Able to locate flood affected areas.	Flood affected areas can be identified and highlighted, particularly flood extents.
Ability to output data in specified format.	Output file can be produced and placed in a storage environment for collection. (TBC)
Service works in AWS.	Objectives can be completed successfully in the AWS environment.

3. Documentation

a. Basis for the master test plan

The following documents are used as basis for this Master Test Plan

Document name	Date	Author
Project Plan	April 2021	Cameron Nyberg
Initial Requirements Model	April 2021	Andrew Smith
High Level Business Requirements	March 2021	Maria Jansen

4. Test Strategy

The Test Strategy for the project will be based upon a Risk Analysis to focus testing and ensure time allocated for testing is used efficiently. Large defects in the solution will aim to be found early as possible to provide time for a resolution with a minimum of disruption to the overall project.

a. Product risk analysis

Product risk analysis assists in determining test goals. Risks are categorised dependent on the chance of failure and extent of damage they may cause. Risk class A is the highest with risk class C being the lowest.

Risk table

Characteristic	Risk Class	Description
Functionality	A	Project is required to meet requirements and will be used in critical situations.
User Friendliness	C	Current scope of project indicates that a documented API module is all which is needed, no User Interface at this stage.
Performance	B	While speed of image processing is not critical, processing still needs to be completed within 4 hours. The processing may be batch driven, depending on the technique and EC2 Instance type.
Security	B	While security is paramount, the images and data for this project does not contain customer specific data, but does contain data which is publicly available (TBC).
Suitability	A	Project is required to meet requirements, scenarios will be provided by DCS to assess suitability of the product.

b. Test Strategy

Testing will focus on high risk aspects as early as possible in the project.

Characteristic	Risk Class	High Level Test Case Descriptions.	Test Level					
			SR	UT	IT	FAT	UAT	ST
Functionality	A	Does the project Plan Meet Requirements. Can images be uploaded and accessed by internal services? Can aerial imagery be processed and highlighted? Can Flood affected areas be identified and highlighted?. Does the model work across a number of scenarios / images? Can an output file be produced and placed in a storage environment for collection? Does the service work within Amazon web services?	***	**	**	**	**	**
User	C	Can a user use the	*					

Friendliness		solution successfully?						
Performance	B	Does the system only use a reasonable amount of resources (memory, compute, storage). Does the system work appropriately with chosen Architecture ie. instance type.	**		**	*	*	*
Security	B	Is access controlled by MFA? Does the project follow security best practice (OWASP)? Is the solution secured at all layers? Is data protected in Transit and rest?	***			**	*	*
Suitability	A	Does the project meet requirements & scenarios provided by DCS Is the project finished by the end of October 2021. .	***	**	**	**	**	**

Test Levels	**Test extensiveness**
SR - Static Review	* - Limited ** - Medium

UT - Unit tests IT - Integration Tests FAT - Functional Acceptance Test UAT - User acceptance test ST - System test	*** - High

5. Approach

a. Test levels

The following test levels will be used for testing this project:

Test Level	Goal
Static Review (SR)	Review of documentation and techniques to ensure suitability for the requirements.
Unit Tests (UT)	Verify each piece of software using isolated tests.
Integration Tests (IT)	Testing parts of the system in combination once integrated.
Functional Acceptance Test (FAT)	Check that the system complies with requirements for functionality.
User Acceptance Test (UAT)	Ensure the solution can handle required tasks in actual scenarios. Includes tests which cannot be automated and require human review such as flood extent review.
System Test (ST)	Test all components as a whole to ensure.

b. Deliverables

Deliverables at each stage of the Unified Process Methodology:

Phase	Deliverables
Inception	Master Test Plan
Elaboration	Detail Specification of Test Environment Test plan for each test level
Construction	Test Results Defect Report
Transition	Release Notes

c. Go / No go

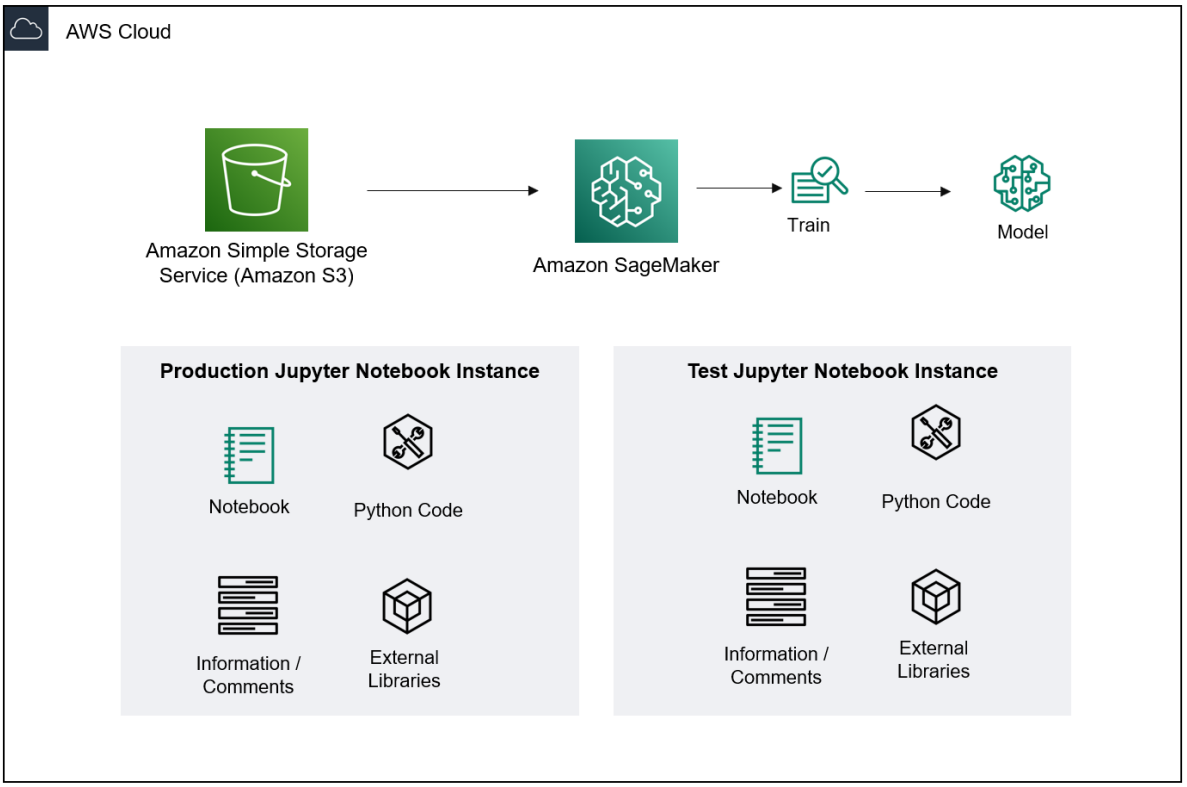
Go / No go of the project will be determined by the Defect Report. The Defect Report needs to contain no major defects which will prevent the system being functional at go - live. The determination of a no-go defect will be completed by Group 5 and will be flagged as such in the defect report.

7. Infrastructure

a. Test environment

The project will be created within the AWS Cloud environment utilising the Amazon Simple Storage Service (Amazon S3) to store images for processing. Images will be processed using python external libraries such as Open CV in combination with the Ground Truth tagging system within Amazon Sagemaker.

The models for Ground Truth will be trained manually to identify and tag bodies of water which will be passed to Open CV for feature extraction.



Jupyter Notebook Instances within the Amazon SageMaker will be used to develop and display the results of python coding in combination with comments and documentation for the code. The advantage of using Jupyter Notebook instances as an editor is that the code and the output are displayed on the one page and code can be run a section at a time for others to learn how the code works.

The test environment will be hosted in a separate Jupyter Notebook Instance for each test level based on the production instance broken down as follows:

Test Level Notebook Instance	Contents
Static Review	Notebook not needed for this level.
Unit Tests	Isolated tests of individual components, confirmation of each stage of the process individually.
Integration Tests	Combination of 2 or more unit tests to ensure compatibility when joined together for each stage of the processing.

Functional Acceptance test	Where possible the notebook will contain tests which ensure the system complies with requirements. This will be complemented by review by an actor.
User Acceptance Test	Where possible the notebook will contain tests which ensure the system complies with requirements. This will be complemented by review by an actor.
System Test	Combining all components to ensure that images can be processed with correct output.

8. Management

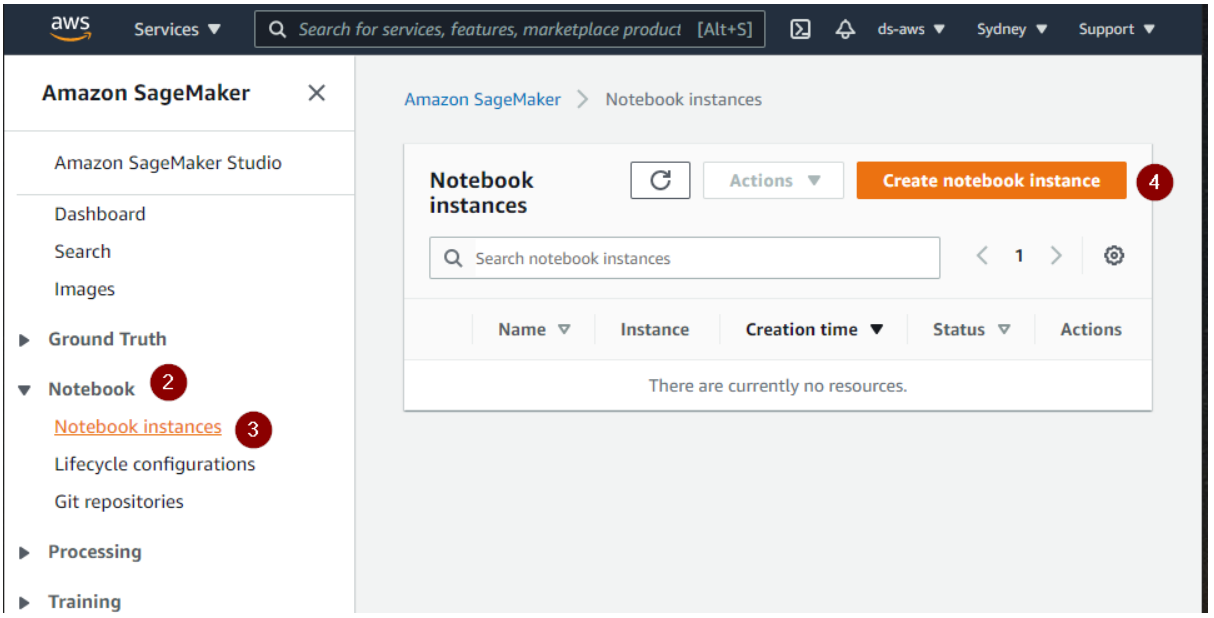
a. Test process management

Management of the test process will be the responsibility of the allocated Test Manager. The manager will oversee testing in consultation with Group 5:

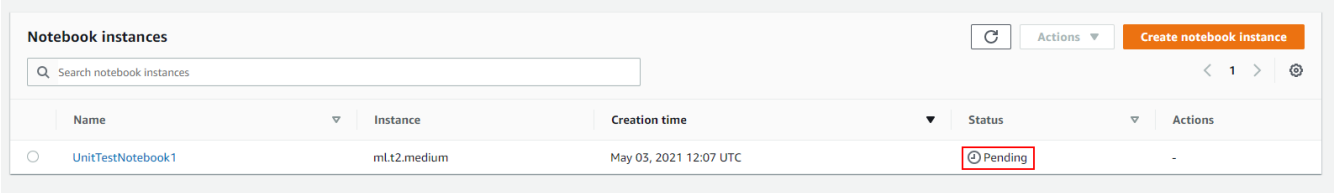
- Documentation of testing.
- Implementation of the test environment.
- Allocation of resources to each test level.
- Execution of Testing.
- Supply of required reporting.

b. Test infrastructure management

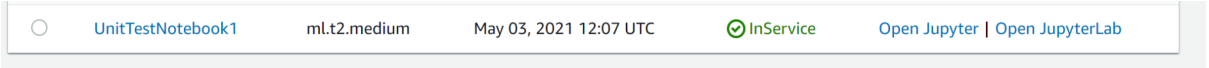
Each test environment consists of a Jupyter Notebook Instance. To create a new instance:



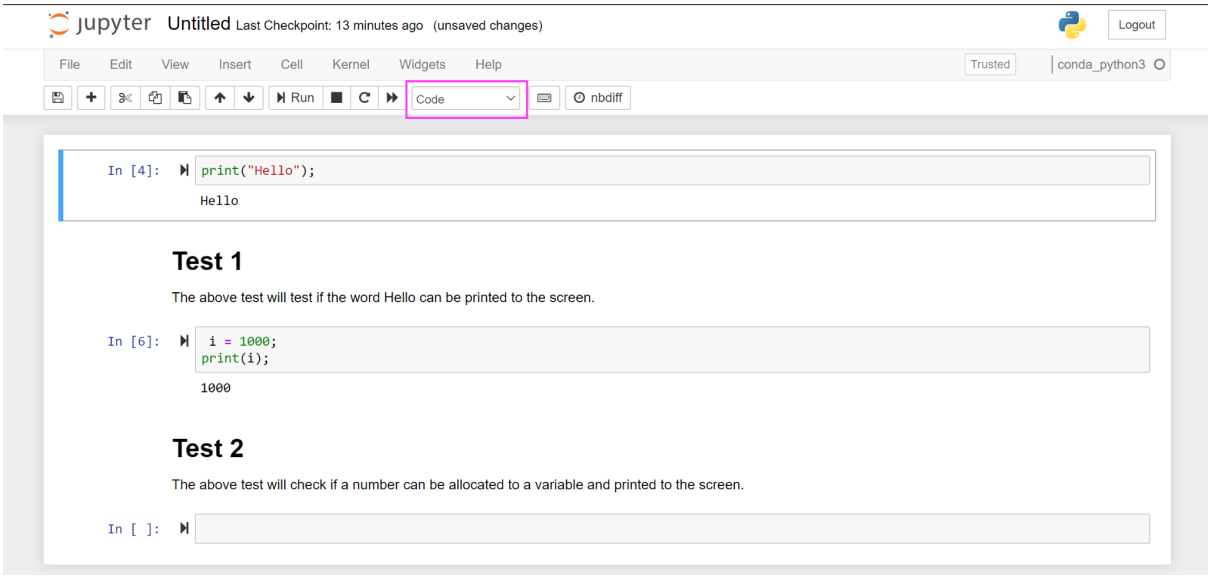
1. Open Amazon SageMaker
2. Click **Notebook**
3. Click **Notebook instances**
4. Click **create notebook instance**
5. Provide a name for the Notebook instance
6. Choose **ml.t2.meduim** as the notebook instance type
7. Select **Create Notebook Instance**.
8. Once a notebook is created it will be pending for a couple of minutes while it is being initialised.



9. Once the notebook is listed as InService it is ready to use. Click **Open Jupyter**.



Once loaded the Jupyter notebook can be used to create cells with code, which will be followed by results and Cells with Markdown to describe the tests. To flag a cell as code or markdown, select a cell and use the toggle at the top.



Basic syntax for the markdown text can be found at https://www.markdownguide.org/basic-syntax/ however the standard for each test will be:

```
# Heading / Test Name
Text describing test here.
```

Ensure the notebook instance is shut down following testing to minimise costs.

c. Defects procedure

For the purpose of this testing a defect will be defined as a variance found within the software from the requirements of the project. During testing if a defect is found it will be logged within a defect report which contains the following information:

- **ID** - Unique identification number for the defect.
- **Description** - A description of the defect with steps for reproducing.
- **Date** - When the defect was found.
- **Test Case** - The test case which was being tested when the defect was identified.
- **Resolution** - How the defect was resolved and who by.

Defects will be discussed by Group 5 at the regular weekly meeting during testing with high priority defects being discussed as they are found.

Group 5 members will be given the task of researching and resolving a given defect and may work in collaboration.

9. Test process risks and countermeasures

Risks which apply directly to the test process or as a result of testing. This list will be updated throughout the test process to ensure all risks are documented. See also the Overall Risk Assessment document for risks related to the whole project. Impact is rated with A being the highest impact, and C being the lowest.

Test Process Risk	Impact	Mitigant
Scope of Testing could expand.	B - Medium impact to project	Minimise scope where possible, or allocate adequate test resources as required.
Minimal time allocated to testing.	B - Medium impact to project	Minimise scope creep, ensure testing is well scoped from beginning.
Large issues not identified at an early stage in the project.	A - High Impact to project	Using Risk Based Testing, testing highest risks first.
Testing for a given Unified Process Phase runs overtime.	A - High Impact to project	Ensure adequate test resources are allocated within a detailed test plan.
Unable to test a given requirement.	C - Low Impact to project	Unlikely to happen as testing will include actors for situations where automation is unsuitable.
Dependency reliance	C - Low impact to project.	Testing is dependent on several Amazon Web Services, high uptime provides certainty that the test environment will be available when required.
Inadequate Documentation	B - Medium Impact to project.	Lack of requirements or documentation could lead

		to inaccurate testing. Ensure all requirements and documentation are up to date.
Inadequate skills for defect resolution.	B - Medium Impact to project.	While Group 5 are relatively new to Amazon Web Services, Intellify are assisting with regular technical meetings and may assist with defect resolution.

10. Planning

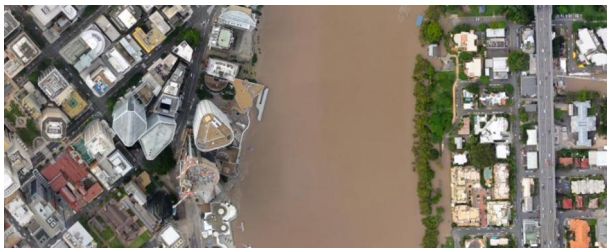
Test activities undertaken during the project.

Phase	Iteration	Dates	Activities
Inception	I-2	16/3 - 12/4	Begin Static Review
Elaboration Phase	E-2	24/4 - 7/5	Produce Master Test Plan
	E-4	22/5 - 2/6	Produce Detailed Specification for test environment. Produce Test Plan for each test level.
Construction Phase	C -1	10/7 - 23/7	Unit testing of components as they are created. Populate Test Results. Populate Defect report.
	C - 2	24/7 - 6/8	Unit testing of

			components. Integration Testing of components Populate Test Results. Populate Defect report.
	C - 3	7/8 - 20/8	Integration Testing of Components. Functional Acceptance Test. Populate Test Results. Populate Defect report.
	C - 4	21/8 - 30/9	User Acceptance Testing. System Testing. Populate Test Results. Populate Defect report. Go / no go decision.
Transition Phase	T - 1	4/9 - 17/9	System Testing in Live Environment.
	T - 2	18/9 - 1/10	Provide Test Results and Defect Report.
	T - 3	2/10 - 13/10	

11. References

Title page cover photo.
Riverside CBD flooded. (2011). Retrieved from https://www.abc.net.au/news/2011-06-03/brisbane-floods3a-before-and-after-article/2742794?nw=0



AWS Architecture Icons for Diagrams.
Amazon Web Services Inc. (2021). Retrieved from https://aws.amazon.com/architecture/icons/