Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Mr Qi YuTao |
| **Class** | P02 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| 1605873G | Timothy Lua |
| 1603170J | Tang Xin Wei |
| 1603477I | Goh Wei Kang |
| 1605129D | Gary Tan Jun Xian |
| 1602118J | Oliver Choy Chen Fung |

**Introduction to System Development Life Cycle (Gary Tan)**

System Development Life Cycle (SDLC) is a series of six main phases to create a hardware system only, a software system only or a combination of both to meet or exceed customer’s expectations.

SDLC consists of 6 phases,

1. Requirement gathering and analysis

This phase is mostly used to determine requirements such as users, type of data needed by the system, and the type of output by the system etc. Afterwards, the requirements are analysed for their validity and if they are able to be incorporated into the system.

1. Design

This phase helps to specify hardware and system requirement and it also helps in define overall system architecture.

1. Implementation or coding

This phase is when coding is started, this is typically the longest phase of the software development cycle.

1. Testing

Once the code is functional, the code is tested against the requirements to ensure that the product is solving the needs that was pointed out during the requirements phase.

1. Deployment

After testing the product successfully, the next phase would have the customer using the product. The customers would have to do beta testing and if any change were needed or bugs were found, they will report it to the engineering team. Once changes were made and bugs are fixed, final deployment will happen.

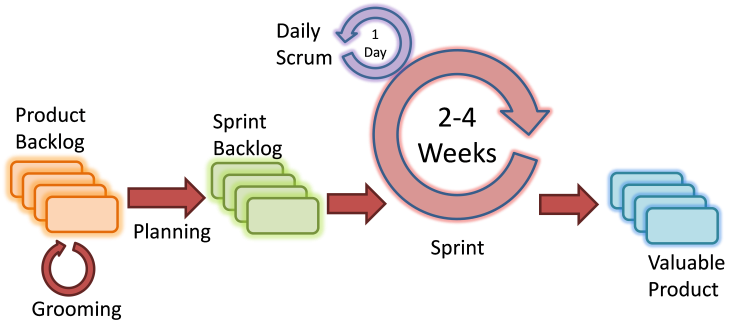
1. Maintenance

As the customer uses the developed system, if problems come up, the maintenance is required. This phase is mostly just to take care of small problems that come up from the developed product.

**Agile Model**

Description:

It is based on iterative and incremental development, where requirements and solutions evolve through collaboration between cross-functional teams.



Scrum Agile Model

The usage:

It can be used with any type of the project, but it needs more engagement from the customer and to be interactive. Also, it can be used when the customer needs to have some functional requirement ready in less than three weeks and the requirements are not clear enough.

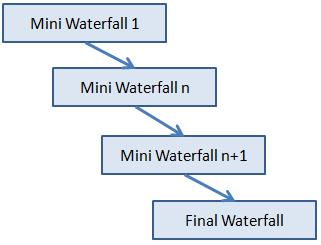
|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Decrease the time required to avail some system features. | Scalability. |
| Face to face communication and continuous inputs from customer representative leaves no space for guesswork. | The ability and collaboration of the customer to express user needs. |
| The end result is the high-quality software in the least possible time duration and satisfied customer. | Documentation is done at later stages. |
|  | Reduce the usability of components. |
|  | Needs special skills for the team. |

**Iterative and Incremental Model**

Description:

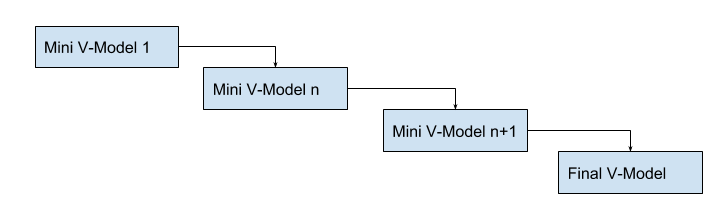
It is developed to overcome the weaknesses of the waterfall model. It starts with an initial planning and ends with deployment with the cyclic interactions in between. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental), allowing software developers to take advantage of what was learned during the development of earlier parts or versions of the system.

It can consist of mini waterfalls or mini V-Shaped model



The usage:

It is used in shrink-wrap application and large system which built-in small phases or segments. Also, can be used in a system has separated components, for example, ERP system. Which we can start with the budget module as a first iteration and then we can start with inventory module and so forth.



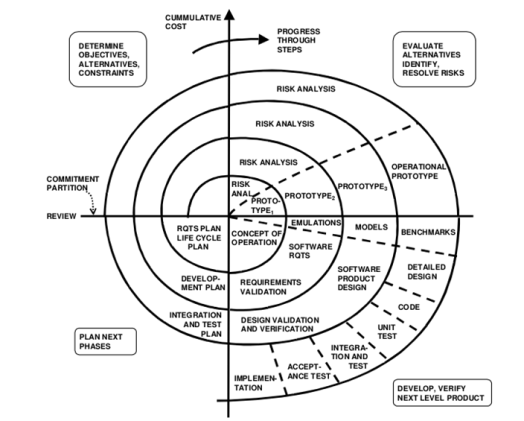
|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Produces business value early in the development lifecycle. | Requires heavy documentation. |
| Better use of scarce resources through proper increment definition. | Follows a defined set of processes. |
| Can accommodate some change requests between increments. | Defines increments based on function and feature dependencies. |
| More focused on customer value than the linear approaches. | Requires more customer involvement than the linear approaches. |
| Problems can be detected earlier. | Partitioning the functions and features might be problematic. |
|  | Integration between iteration can be an issue if this is not considered during the development. |

**Spiral Model**

Description:

It is combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts.

This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects. This model uses many of the same phases as the waterfall model, in essentially the same order, separated by planning, risk assessment, and the building of prototypes and simulations.



The usage:

It is used in the large applications and systems which built-in small phases or segments.

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Estimates (i.e. budget, schedule, etc.) become more realistic as work progressed because important issues are discovered earlier. | High cost and time to reach the final product. |
| Early involvement of developers. | Needs special skills to evaluate the risks and assumptions. |
| Manages risks and develops the system into phases. | Highly customized limiting re-usability |

Reference:

<http://istqbexamcertification.com/what-are-the-software-development-life-cycle-sdlc-phases/>

https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/

**What is SDLC? Tang Xin Wei**

SDLC stands for Software Development Life Cycle. It is a process that consists of a series of planned activities to develop or modify the software products.

**Phases of SDLC**

Phase 1: Requirement Gathering and Analysis

It is the main focus of the project managers and stakeholders. It is to meet with the clients to determine the requirement. After requirement gathering these requirements are analyzed for their legitimacy and probability of incorporating the requirement into the framework.

Phase 2: Design

This phase, the system and software design is prepared from the requirement specification that was from phase 1.

Phase 3: Implementation/Coding

In this phase the main focus is for the developer to focus on the coding.

Phase 4: Testing

In this phase, after the code is developed, it is tested against the requirements to make sure that the product is solving the needs addressed and gathered during the first phase.

Phase 5: Deployment

After testing the product, it is deployed to customer for their use.

Phase 6: Maintenance

This phase is to maintain the product once when the customer starts using the developed system which problems will start to arise and is needed to be solved from time to time.

**Big Bang Model**

What is big bang model?

It is an SDLC model where it does not follow any specific process. The development starts with the required money and efforts as the inputs, and the output is the software developed which might possibly according to client’s requirement.

Big Bang Model – Design and Application

This model comprises of focusing all the possible resources in the software development and coding, with few to no planning. The requirements are understood and implemented as the process go along. Any changes required may or may not need to revamp the complete software.

Big Bang Model – Pros and Cons

Advantages

* Simple model
* Little or no planning required
* Easily manageable
* Very few resources required
* Gives flexibility to developers
* Good learning aid for new comer or students

Disadvantages

* Very high risk and uncertainty
* Not a good model for complex and object-oriented projects
* Poor model for long and ongoing projects
* Can turn out to be very expensive if requirement is misunderstood

**RAD Model**

What is RAD model?

RAD stands for rapid application development. This model is based on prototyping and iterative development with no definitive planning involve.

What is RAD?

It is a software development method that uses little planning in favor of rapid prototyping. The functional modules are developed in parallel as prototypes and are integrated to make the complete product for faster product delivery.

RAD Pros and Cons

Advantages:

* Changing requirements can be accommodated.
* Progress can be measured.
* Iteration time can be short with use of powerful RAD tools.
* Productivity with fewer people in a short time.
* Reduced development time.
* Increases reusability of components.
* Quick initial reviews occur.
* Encourages customer feedback.
* Integration from very beginning solves a lot of integration issues.

Disadvantages

* Dependency on technically strong team members for identifying business requirements.
* Only system that can be modularized can be built using RAD.
* Requires highly skilled developers/designers.
* High dependency on modeling skills.
* Inapplicable to cheaper projects as cost of modeling and automated code generation is very high.
* Management complexity is more.
* Suitable for systems that are component based and scalable.
* Requires user involvement throughout the life cycle.
* Suitable for project requiring shorter development times.

**Software Prototyping**

What is software prototyping?

It is a working model of software with some limited functionality. Prototyping allows users to evaluate developer proposals and try them out before implementation. It is to help understand the requirements which are personalize, and which may not have been considered by the developer during the product design.

Software Prototyping Pros and Cons

Advantages

* Increased user involvement in the product even before its implementation.
* Since a working model of the system is displayed, the users get a better understanding of the system being developed.
* Reduces time and cost as the defects can be detected much earlier.
* Quicker user feedback is available leading to better solutions.
* Missing functionality can be identified easily.
* Confusing or difficult functions can be identified.

Disadvantages

* Risk of insufficient requirement analysis owing to too much dependency on the prototype.
* Users may get confused in the prototypes and actual systems.
* Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.
* Developers may try to reuse the existing prototypes to build the actual system, even when it is not technically feasible.
* The effort invested in building prototypes may be too much if it is not monitored properly.

Reference

<https://www.tutorialspoint.com/sdlc/index.htm>

https://www.tutorialspoint.com/sdlc/sdlc\_bigbang\_model.htm

https://www.tutorialspoint.com/sdlc/sdlc\_rad\_model.htm

https://www.tutorialspoint.com/sdlc/sdlc\_software\_prototyping.htm

<http://istqbexamcertification.com/what-are-the-software-development-life-cycle-sdlc-phases/>

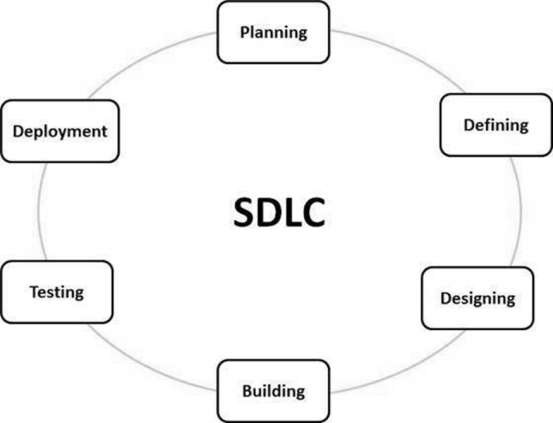
**Introduction to SDLC (Timothy Lua)**

Software Development Life Cycle (SDLC) is a process that produces highest quality software with the lowest cost and in the fastest time possible. SDLC comprises of a detailed plan on the ways to develop, modify, maintain and replace a software system.

A software engineer should choose the right SDLC model based on the requirements of the project to ensure success in developing the software.

SDLC involves multiple stages, including planning, design, building, testing, and deployment.

This diagram shows the stages of SDLC.



*https://www.tutorialspoint.com/sdlc/sdlc\_overview.htm*

**Stage 1: Planning and Requirement Analysis**

This is the most important and necessary stage in SDLC. Information is gathered from customer feedback, market surveys, input from sales department and industry experts. This information is then used to plan

### **Stage 2: Defining Requirements**

When the requirement analysis is is completed the next stage is to define and document the product requirements for it to be approved by the customer or the market analyst. This will be done using Software Requirement Specification (SRS) which contains all the product requirements to be designed and developed during the project life cycle.

### **Stage 3: Designing the Product Architecture**

The SRS will be used as a reference for the developers to come out with the best plan for the product to be developed. Based on the requirements set in the SRS, there will be multiple design approach for the product architecture proposed which will be used in Design Document Specification (DDS).

DDS will be assessed by all the important stakeholders based on risk assessment, product robustness, design modularity, budget and time restrictions. The best design approach will be chosen for the product.

A design approach specifies all the architectural modules of the product, together with communication and data flow representation with the external modules if there are any. The internal design of all the modules of the proposed architecture should be clearly specified with all the details in DDS.

### **Stage 4: Building or Developing the Product**

Development starts at this stage and the product is built. The product coding is generated as follows in the DDS during this stage. If the design is performed in a detailed and systematic way, coding can be completed without any issues.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, debuggers, etc. are used to generate the code. The programming language is chosen based on the type of software being developed which can be C, C++, Java, PHP etc.

### **Stage 5: Testing the Product**

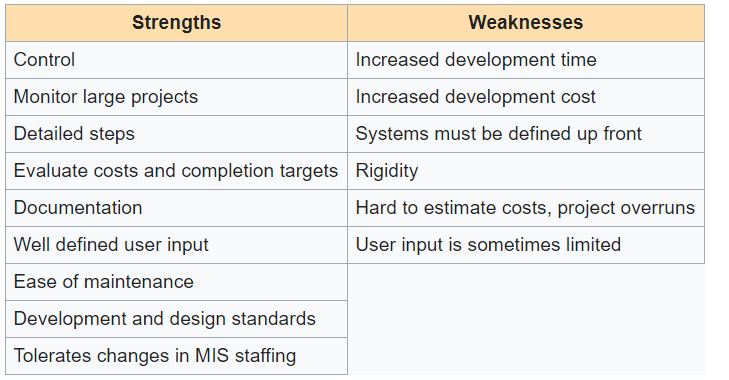
This stage refers to the testing only stage of the product where product errors are reported, tracked, fixed and retested, until the product reaches the standard quality set in the SRS.

### **Stage 6: Deployment in the Market and Maintenance**

Once the product is tested and ready to be deployed it is released in the market. The product may be released first in a limited section and tested in the real world environment User Acceptance Testing (UAT).

Afterwards, based on the feedback given the product may be released as it is or with suggested improvements in the targeted market. Once the product is released into the market, its maintenance is done for the existing customers.

**Strengths & Weakness of SDLC**



*https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf*

**Benefits of SDLC**

When SDLC is used it can allow high level of management control and documentation.

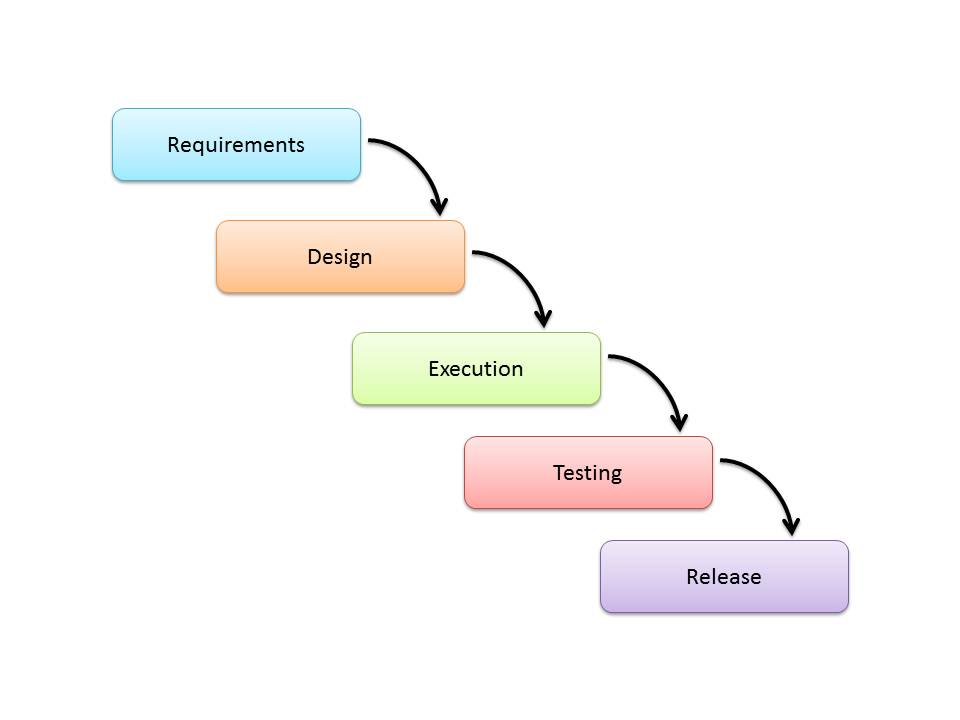
Developers will have a clear understanding of what to build and why. When client and developers come to an agreement on the goals at the beginning and come up with a plan to develop the product. Everyone will be clear of the costs and the resources required to devlopment product.

Some difficulties can turn the SDLC process into more of a obstacle to development than a tool. Failure to take into consideration the needs of customers and all the stakeholders can lead to a poor understanding of the system requirements at the start. Benefits of SDLC will be seen if only the plan is followed closely.

**Waterfall Model**

**Description**

In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases (Sami, 2012). The waterfall model does not take into consideration to go back to the previous phase if there are any changes in the requirement.

****

*https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/*

**Usage**

This model will be suitable for projects that have no changes in the requirements. Example would be projects from request for proposals (RFPs), as the client will have a very clear set of requirements given. In the RFP it will include what problem the client wants the product to solve and the functions they require.

Situations where most appropriate for waterfall model:

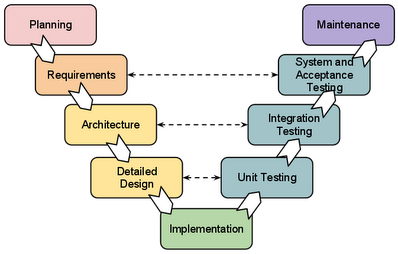
1. Project is for development of a mainframe-based or transaction-oriented batch system.
2. Project is large, expensive, and complicated.
3. Project has clear objectives and solution.
4. Pressure does not exist for immediate implementation.
5. Project requirements can be stated unambiguously and comprehensively.
6. Project requirements are stable or unchanging during the system development life cycle.
7. User community is fully knowledgeable in the business and application.
8. Team members may be inexperienced.
9. Team composition is unstable and expected to fluctuate.
10. Project manager may not be fully experienced.
11. Resources need to be conserved.
12. Strict requirement exists for formal approvals at designated milestones.

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| * Easy to explain to users * Structures approach * Stages and activities are well defined * Helps plan and schedule project * Verification at each stage ensures early detection of errors * Each phase has specific deliverables | * Assumes the requirements of the system is always the same * Difficult to go back to any stage after it is finished * Little flexibility * Adjusting scope is difficult and expensive * Costly and requires more time to plan |

**V-Shaped Model**

**Description**

This is an extension of the waterfall model. In this model the process steps are bent upwards after the implementation phase to form a V shape. The V-shaped model has early test planning compared to waterfall model.



*https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/*

**Usage**

V-shaped model is very similar to the waterfall model, because both the models are sequential models. Requirements have to be very defined before the project begins, as it is very expensive to revert and make changes.

Situations where most appropriate for waterfall model:

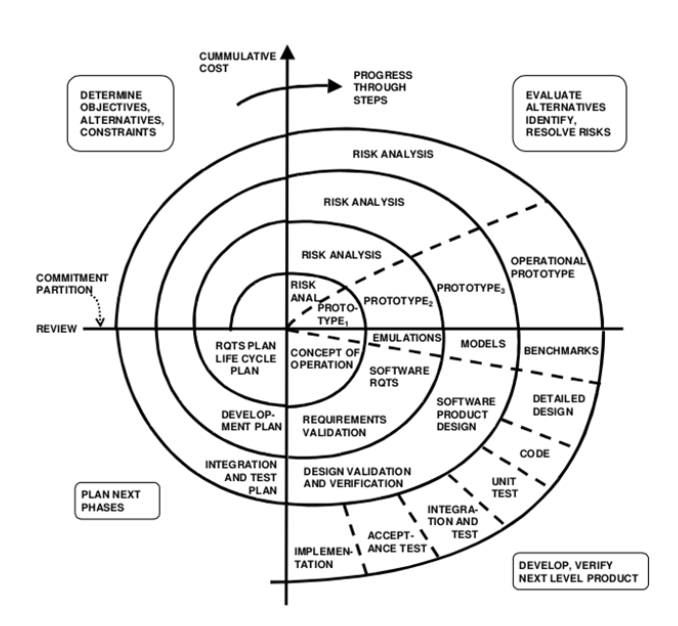
1. Requirements are well defined, clearly documented and fixed.
2. Product definition is stable.
3. Technology is not dynamic and is well understood by the project team.
4. There are no ambiguous or undefined requirements.
5. The project is short.

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| * Simple and easy to use * Each phase has specific deliverables * Higher chance of success due to the development of test plans early on during the life cycle. * Works well for when requirements are clear * Verification and validation of the product in early stages of software development | * Very inflexible * Adjusting scope is difficult and expensive * No early prototypes of the software are produced. * Does not provide a clear path for problems found during testing phases * Costly and requires more time to plan |

**Spiral Model**

**Description**

The spiral model is a combination of both design and prototyping in stages, which combines the advantages of top-down and bottom-up concepts. This model combines elements of the prototyping model and waterfall model. Spiral model uses many of the same phases as the waterfall model, in the same order, separated by planning, risk assessment and building of prototypes and simulations.



*https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/*

**Usage**

This model is used in the large applications and systems which built-in small phases or segments. This model is suitable for large, expensive and complicated projects. Situations where most appropriate for spiral model:

1. When there is a budget constraint and risk evaluation is important.
2. For medium to high-risk projects.
3. Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
4. Customer is not sure of their requirements which is usually the case.
5. Requirements are complex and need evaluation to get clarity.
6. New product line which should be released in phases to get enough customer feedback.
7. Significant changes are expected in the product during the development cycle.

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| * Estimates (i.e. budget, schedule, etc.) become more feasible as work progressed because crucial problems are found earlier. * Early participation of developers * Manages risks and develops the system into stages | * High cost and time to get to final product * Needs expertise to evaluate the risks and assumptions * Highly customized limiting re-usability |

**References**:

Sami, M. (2017, August 29). *Software development life cycle models and methodologies*. Retrieved from

https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/

Selecting a development approach. (2008, March 27). Retrieved from

https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf

Tutorials Point. (2017, August 15). *SDLC V-Model*. Retrieved from

https://www.tutorialspoint.com/sdlc/sdlc\_v\_model.htm

What is SDLC? Understand the Software Development Life Cycle. (2017, October 04). Retrieved from

https://stackify.com/what-is-sdlc/

What is SDLC? (Wei Kang)

The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process. SDLC is used as a model for the development team of the software organization to follow. It consists of a detailed plan describing how to develop, maintain and replace specific software. The life cycle defines a methodology for improving the quality of software and the overall development process. There are 6 phases of the software development life cycle.

1.Requirements gathering and analysis

Business requirements are gathered in this phase. There will be meetings with managers, stake holders and users to determine the requirements like; who is going to use the system? How will they use the system? These are some of the questions that will be answered during a requirement gathering phase. Afterwards, these requirements will be complied and analyzed for validity and a study of whether the requirements can be incorporated into the system will be carried out. Then, a requirements specification document is made which serves as a guideline for the next phase.

2. Design

This phase prepares the system and software design from the requirement specifications which were studied in the first place. System Design helps in specifying hardware and system requirements and helps in defining overall system architecture. The system design specifications serve as input for the next phase of the model.

3.Implementation/coding

After receiving the system design document, the work is spilt up into units and the coding is started. This is the longest phase of the software development life cycle.

4.Testing

After the code is developed, it is tested against the requirements to make sure the product is actually useful and solving the needs addressed and gathered. All tests like functional testing, unit testing, integration testing, system testing, acceptance testing, and non-functional testing are done during this phase.

5.Deployment

After the tests have been run and are successful, the product is deployed to the customer for their use.

6.Maintenance

Once the customers start using the developed system and actual problems comes up, it needs to be solved from time to time. This process of solving the problems is known as maintenance.

**Waterfall Model**

The waterfall model is a linear sequential (non-iterative) design approach for software development, in which progress flows in one direction downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, deployment and maintenance.

Advantages

* Model is simple and easy to understand and use
* Easy to manage due to the rigidity of the model
* Works well for small projects where the requirements have been understood
* The phases are processed and completed one at a time

Disadvantages

* Once an app has reached the testing stage, it will be difficult to reverse and alter something that was not well thought of in the concept stage
* Has high amounts of risks and uncertainty
* Unsuitable for complex and object-oriented projects
* Not suitable for projects where the requirements have a high tendency of changing
* No working software is produced until late during the life cycle

**Spiral Model**

The spiral model is similar to the [incremental model](http://istqbexamcertification.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it/), with more emphasis placed on risk analysis. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements are gathered, and risk is assessed. Each subsequent spiral build on the baseline spiral.

Advantages

* Adequate risk analysis is done hence avoiding number of risks
* Ideal for large and mission-critical projects
* New functions can be added at later dates
* Software is produced early in the software life cycle

Disadvantages

* High cost for usage of the model
* Requires experts for risk analysis
* Success of the project is reliant on the risk analysis phase
* Not suitable for small projects

**Iterative and Incremental Model**

In incremental model the whole requirement is divided into various builds. Multiple development cycles take place here, making the life cycle a [“multi-waterfall” cycle](http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/).  Cycles are divided up into smaller, more easily managed modules. Incremental model is a type of software development model like [V-model](http://istqbexamcertification.com/what-is-v-model-advantages-disadvantages-and-when-to-use-it/), [Agile model](http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/) etc.

Advantages

* Develops working software quickly and early during the software life cycle
* The model is flexible thus the changing of scope and requirements will be cheaper
* Low initial delivery cost
* Easy to manage risks because risky pieces are identifies and dealt with during the iterations

Disadvantages

* Requires good planning and design
* Cost is higher than waterfall model
* Requires a clear and complete definition of the whole system before it can be broken down and built incrementally

References

<http://istqbexamcertification.com/what-is-spiral-model-advantages-disadvantages-and-when-to-use-it/>

<http://istqbexamcertification.com/what-is-incremental-model-advantages-disadvantages-and-when-to-use-it/>

<https://www.techopedia.com/definition/22193/software-development-life-cycle-sdlc>

Name: Oliver Choy Chen Fung (1602118j) Class: P02

**What is SDLC?**

The Software Development Life Cycle(SDLC) is a process used by the software industry to design, develop and test software. The aim of the SDLC is to produce high-quality software that will either meet or exceed the expectations of the customers, and attain completion within the time and money estimates.

The Software Development Life Cycle is a process that contains a detailed plan on how to develop, maintain, replace and alter or enhance a specific software. A typical SDLC consists of Planning, Defining, Designing, Building, Testing and Deployment.

**Stage 1: Planning and Requirement Analysis**

This stage is usually done by the senior members of the teams with information provided by various parties such as the customers and sales department. The information provided by the various parties helps to form the basic project approach, as well as conduct product feasibility studies in areas such as economical, operational and technical. For example, feasibility study in the technical area is done to define the different technical approaches the project can adhere to be completed successfully with minimal risks. Quality assurance requirements and identifying the risks related to the project is also done in this phase.

**Stage 2: Defining Requirements**

This purpose of this phase is to clearly define and document the project requirements, which has to be then approved by either the customers or market analysts. A document known as the Software Requirement Specification (SRS) contains all the product requirements to be designed and developed during the life cycle of the project.

**Stage 3: Designing the Product Architecture**

Product architects usually use the Software Requirement Specification as a point of reference to plan the best architecture for the product to be developed. There will usually be more than one design, depending on the requirements, which will be documented in the Design Document Specification (DDS). The DDS is reviewed by important stakeholders and assess parameters such as product robustness and budget constraints. The internal designs of the modules must be clearly defined in the DDS.

**Stage 4: Building or Developing the Product**

This stage is where the actual development of the product is at. Programming codes are generated based on the Design Document Specification. The programming language chosen for the project depends on the software being developed.

**Stage 5: Testing the Product**

Most stages includes testing, however this is the only stage where the product defects are reported, tracked fixed and retested until it meets the standards stated in the Software Requirement Specification.

**Stage 6: Deployment in the Market and Maintenance**

Once the product is certified market-ready, it is officially deployed into the market. At times, product deployment can be released in stages, depending on the business strategy of the organisation. Based on the feedback generated, the product can remain as it is or certain improvement on the current product will be made.

**Waterfall Model**

The Waterfall Model design is the first Process Model to be introduced. This model is also known as a linear-sequential life cycle model. In this model, each phase must be completed before the next phase can begin. For example, the Planning phase has to be completed before the Defining phase can be started. This model also ensures that phases do not overlap.

**Advantages**

* Simplicity and ease of understanding and usage of it.
* Phases are processed and completed one by one.
* Processes and results are well documented.
* Ease in arranging tasks.
* Stages are well defined.

**Disadvantages**

* Many risks and uncertainty .
* Not a good model for projects that are complex and object oriented .
* There will not be any working software until very late into the life cycle.
* The adjustment of the scope during the life cycle and end a project.
* Poor model for long projects.

**Iterative Model**

This process starts out with a simple implementation of a sub-part of the software requirements and repeatedly enhance each upgraded version until the completed system is attained. Changes such as design modifications and additional functional capabilities are implemented in each iteration.

**Advantages**

* During the early life cycle, some working functionalities can be developed.
* Risks are discovered during each iteration which can be then resolved.
* Results are attained earlier and on a periodic basis.
* Progress can be tracked.
* Risk analysis is better.

**Disadvantages**

* More resources might be needed.
* The identifying of increments may need the complete system to be defined.
* More attention from the management is needed.
* Not suitable for small projects.
* End of project is not known, hence posing a risk.

**Agile Model**

This model is a combination of both the iterative and incremental process model, focusing on both the process adaptability and rapid delivery of working software to meet the customer’s satisfaction. Just like the Iterative model, the Agile method also splits the product into small incremental builds. The Agile model believes that each project needs to be tailored differently to suit its own needs. There are some Agile Manifesto principles, and one of them is known as Customer Collaboration. This principle requires a constant interaction with the customer to get proper product requirements, as not much information cannot be acquired at the beginning.

**Advantages**

* A realistic approach to software development.
* It is simple to manage
* Provides developers with flexibility.
* Little to no planning needed.
* Provides early partial working solutions.

**Disadvantages**

* Not able to handle complex dependencies
* It has a higher risk of sustainability, maintainability and extensibility.
* Requires a high level of independent dependency, as there is very little documentation acquired
* Requires a lot of customer interaction, hence if the customer is not clear, could lead the team in the wrong direction.
* Lack of documentation can prove to be challenging when transferring the technology to new team members.

**References**

Tutorialspoint.com. (2017, August 15). SDLC Overview. Retrieved from https://www.tutorialspoint.com/sdlc/sdlc\_overview.htm

Tutorialspoint.com. (2017, August 15). SDLC Waterfall Model. Retrieved from https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm

Tutorialspoint.com. (2017, August 15). SDLC Iterative Model. Retrieved from https://www.tutorialspoint.com/sdlc/sdlc\_iterative\_model.htm

Tutorialspoint.com. (2017, August 15). SDLC Agile Model. Retrieved from https://www.tutorialspoint.com/sdlc/sdlc\_agile\_model.htm

**Temasek Polytechnic**

**School of Informatics and IT**

**Diploma in Information Technology (IT)**

Terms of Reference

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Mr Qi Yutao |
| **Class** | P02 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| **1605129D** | **Gary Tan Jun Xian** |
| **1603477I** | **Goh Wei Kang** |
| **1602118J** | **Oliver Choy Chen Fung** |
| **1603170J** | **Tang Xin Wei** |
| **1605873G** | **Timothy Lua** |

**1. Introduction**

The owners of the hotel Delonix Regia has commissioned a development firm called Pantheon Systems, in which a group of students from Temasek Polytechnic who are on our Student Internship Programme are working at, to create a hotel management system for their hotel. The project was given to the students to do.

Hotel management systems are necessity due to its ability to provide better management tools to the staff, which would allow the staff to provide better hospitality. Better hospitality can lead to an increase of customers for the hotel. The lack of customers could be due to the fact that they were missing a hotel management system, leading to long wait times for replies, room service, etc.

The fact that the hotel Delonix Regia is located at a good district but fails to attract customers only supports our theory, hence this hotel management system is essential in attracting customers.

**2. Objectives of the Project**

Describe the objectives that you want to achieve through this project. Objectives may be both technical and non-technical.

* We will be doing 5 modules for our hotel management system
  + **Customer Relationship Management (CRM)**
    - The aim of the Customer Relationship Management is to improve the relations between the customer and the hotel. As such, this module needs to work on improving this relationship. With the Hotel Management System, the hotel will be able to cater to the customers needs, as well as provide them with a FAQ( Frequently Asked Question) and a customer support hotline.
  + **Housekeeping Management (HKM)**
    - The aim of the Housekeeping Management is to do cleaning up of rooms and topping up of amenities. The Hotel Management System will help to filter out the rooms that require housekeeping and which do not need. This will be more systematic as employees do not need to check if every room requires housekeeping.
  + **Room Booking Software (RBS)**
    - The aim of the Room Booking Software is to enable the customers to book their room online. The Hotel Management System will allow both customers and employees to see if a room is available through the real-time information on room availability. This module will also allow the hotel to take down the needs of customers as well as confirm their mode of payment.
  + **Room and Facilities Management (RFM)**
    - The aim of the Room and Facilities Management is to provide employees with better management of rooms and facilities. With the Hotel Management System, the employees can assign the customers different room locations depending on their needs, for example if a customer wants a smoking room, the employees can allocate them that room. As for facilities, this module can manage facilities such as concierge.

* + **Human Resource Management (HRM)**
    - The aim of the Human Resource Management is to provide the hotel with a better overview of their employees. The Hotel Management System will be able to store records of employees such as their leave and reimbursement amount.

**Non-technical:**

* Improve hotel’s overall management
* Improve hotel’s image

The implementation of a Hotel Management System will not only benefit the hotel technically, but also non-technically. As the Hotel Management System provides a smoother and easier process of important tasks of a hotel, such as booking a room, this will allow customers to have confidence in the hotel. Customers would be more willing to come back to the hotel, as compared to without a Hotel Management System. The ratings of the hotel will increase due to customer’s satisfaction. This will in turn boost both the hotel’s overall management as well as its image.

**3. Scope of the Project (module)**

Describe the key features of the software system you are developing. This description does not need to be detailed; it can simply be a few sentences that give a general explanation of the feature.

Key Features:

**Customer Relationship Management (CRM)**

This Module helps to manage the business efficiently

* Sales management and purchase management

This function managers the sales by how the hotel promote themselves and purchase that the hotel needs to buy like amenities that the hotel provides.

* Help Desk functions

This function will provide customers with assistance with the hotel’s services or any other information. Examples will be FAQ (Frequently asked questions) and customer support hotline.

* Control of customers events

This function allows hotel management to take note of the customer’s event to avoid any clashes between events.

* Planning of tasks

This functions allows hotel management to make a plan for the employees on what they should do

**Housekeeping Management**

This module helps to manage the hotel’s housekeeping process.

* Management of housekeeping activities

This function will allow the hotel management to oversee all the hotel rooms that require housekeeping or have already been cleaned. They are able to assign staff to rooms for housekeeping after customer has checked out or at customer’s request.

* Management of room housekeeping

This function is to improve on the process of completing the tasks that the housekeeping department have been tasked to do.

* Control of room inspection

This function allows the hotel management to send staff to inspect room after housekeeping has been done and oversee the rooms that have been inspected or not.

**Room Booking Software**

This module is a hotel booking and reservation system.

* Management of reservation, concierge service

This function controls the reservation to avoid any clashes with other customer and it

also controls the front desk with employees hospitality.

* Management of agreements with hotel’s guests, modes of payment, etc.

This function comes up with ways to accommodate hotel’s guests request and other services requested. This confirms how the customer is going to make payment.

**Room and Facilities Management**

With this module it can help with the management of the facilities in the hotel.

* Management of room’s types

This function manages the different types of room that the hotel have.

* Control of rooms locations

This function manages the different room locations in the hotel, hotel staff can assign customer to different room locations depending on their needs.

* Management of services type

This function allows the hotel management to manage the different services types for example, facilities, transportation and travel.

* Management of services

This function allows the hotel manage to to send request for specific services they require that is not listed.

**Human Resource Management**

Helps to manage the company’s employees and manager details.

* Management of employees

This function manages the amount of employees that will be working in the hotel per shift.

* Timetable/ timesheets and absence plan

This function will manage time timetable/timesheets of staffs and their leave/absence plan.

* Follow-up of services

This function allows the hotel to improve on what customers have complaint about them.

* Sign in/Sign out module

This function will allow the hotel staff to clock in and clock of for the timesheets.

**4. Distribution of Workload**

Determine which members of the team will be responsible for what areas of project work. Individual’s responsibilities should be clearly spelt out.

|  |  |
| --- | --- |
| **Objectives/Deliverables** | **Members** |
| Introduction | Gary |
| Objectives | Wei Kang, Oliver |
| Scope of the Project | Xin Wei, Timothy |
| Distribution of Workload Chart | Gary |
| Constraints | Gary, Xin Wei |
| Resources | Gary, Timothy |
| Product Position | Wei Kang, Oliver |
| Approach and Methodology of the Project | Timothy |

**5. Constraints**

List any constraints that you expect to face e.g. lab opening hours, different timetable schedules, etc.

* Manpower is limited for client and developer
* Hotel’s budget constraints
* Other ongoing modules
* Other outside commitments of members e.g. CCAs, part-time jobs

**6. Resources**

List the hardware and software resources that your project will need.

Application Server X1

HP ProLiant MicroServer Gen8 Ultra Micro Tower Server (783958S01)

* $976.66
* 2.3 GHz Intel Xeon E3-1220L
* 4 GB DDR3
* 2 TB 7200 rpm Hard Drive

*https://www.amazon.com/HP-ProLiant-MicroServer-Server-783958S01/dp/B00LGJ9NHK*



Microsoft Server 2016 software

* $684

Desktop workstation & Monitor X5

(Dell Inspiron Small Desktop 3268) + Monitor(Dell Monitor E2016HV) + Keyboard + Mouse

* $958 Per computer - $4790
* 7th Generation Intel® Core™ i5-7400 processor (6MB Cache up to 3.50 GHz)
* 8GB RAM, DDR4, 2400MHz
* 1TB 7200 rpm HDD

*http://www.dell.com/sg/p/inspiron-3268-desktop/pd?ref=PD\_Family#overrides=w21685315ssgw10:6932~AG-MONE2016HV*

Microsoft Office

* $135 per year

Microsoft Visual Studios

* Free

Developer’s Salary X5

* $12/hr - per developer
* 8hr per day for 50 days
* $4,800 per developer - $24,000

**Total budget: $ 30,585.66**

**7. Product Positioning in the Market/Company**

Describe how your product differentiates itself from similar products in the market. Describe any interesting or unique features that your product has.

Examples of other products in the market

**Hotelogix PMS**

Pricing:

-$3.99/room/month (premium price)

Features:

-multi device booking engine

**MSI CloudPm**

Pricing:

-$4/room/month

Features:

-web-based instructor training to use the software

Assuming that we have 50 rooms for our hotel and the life cycle of the hotel lasts more than 20 years, the cost will be $2394 a year if we are using Hotelogix PMS and $2400 for MSI CloudPM. The total cost for us to build the hotel management system is around $31000. Our maintenance is also free throughout the life cycle of the hotel. That means that our system proved to be the cheaper alternative in the long run. Also, we provides instructors that go down personally to conduct training workshops for learning how to use the software.

The most important factor is that this product is built specifically for Delonix Regia, this means that problems that only affect this particular hotel can be easily be solved. We will also be adding in features that will be suited for Delonix Regia. The system will also be more adjusted to the singaporean taste where they will be terms or icons that will be familiar to a fellow singaporean.

Since the main user of this product does not have any experience in using hotel management systems, the product aims to be as easy to use as possible.

We aim to include as many automated parts as possible, this helps the users as there will be lesser potential mistakes that they will make as compared to having many manual parts. In conclusion, our product aims to be cheap, custom made, more suited and localized for Delonix Regia.

**8. Approach and Methodology of the Project**

Describe the development model your team will be adopting for the project. Describe any potential risks or problems your team might have adopting the development model and what you can do to overcome them.

Agile Model  
It is based on iterative and incremental development, where requirements and solutions evolve through collaboration between cross-functional teams.

Usage

It can be used with any type of the project, but it needs more engagement from the customer and to be interactive. Also, it can be used when the customer needs to have some functional requirement ready in less than three weeks and the requirements are not clear enough.



Justification:

The reason why we chose agile model approach for our project is because agile model is very flexible and adaptive compared to the other traditional development methods.

The flexible approach of agile model does not require a lot of detailed planning and the tasks are straightforward to develop the necessary features. Feature based development allows our team to adapt if there are any changes in project requirements. The product is also tested regularly at every phase decrease any risks of major errors.

For agile method our team will work closely with the client and communicate with each other often. This allows us to receive any feedback from the client side and make the appropriate adjustments or improvements to the product.

This are the advantages of agile model:

* Is a very realistic approach to software development.
* Promotes teamwork and cross training.
* Functionality can be developed rapidly and demonstrated.
* Resource requirements are minimum.
* Suitable for fixed or changing requirements
* Delivers early partial working solutions.
* Good model for environments that change steadily.
* Minimal rules, documentation easily employed.
* Enables concurrent development and delivery within an overall planned context.
* Little or no planning required.
* Easy to manage.
* Gives flexibility to developers.

In summary, our team felt that agile model is the most suitable for this project due to its flexibility, frequent customer interactions and regular testing. This will minimize any miscommunication and risks of major errors of the product for a successful development of the system. Thus making a product customized to fit customer’s needs.

References:

The 5 Most Popular Hotel Management Software Solutions For Small Hotels Compared Capterra Blog [Web log post]. (2017, March 28). Retrieved from <https://blog.capterra.com/the-5-most-popular-hotel-management-software-solutions-for-small-hotels-compared/>

ERP Software Singapore l idatafive l Hotel Management System l ERP Software Singapore. (2017, October 21). Retrieved from <https://www.idatafive.com.sg/hotelmanagementerpsystemsingapore/>

MSI CloudPM Reviews: Overview, Pricing and Features. (n.d.). Retrieved from <https://reviews.financesonline.com/p/msi-cloudpm/>

Software Development Life Cycle Models and Methodologies. (2012, March 15). Retrieved from <https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/>

Tutorialspoint.com. (2017, August 15). SDLC Quick Guide. Retrieved from <https://www.tutorialspoint.com/sdlc/sdlc_quick_guide.htm>

What TigernixHMS Offers. (2017, August 15). Retrieved from <https://www.tigernix.com/home/software/hotel-management-system/features>

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Project Plan

**Project Particulars**

|  |  |
| --- | --- |
| **Tutor** | Mr Qi Yutao |
| **Class** | P02 |
| **Project Title** | Delonix Regia Hotel Management System |

**Project Team’s Particulars**

|  |  |
| --- | --- |
| **Matric Number** | **Student Name** |
| **1605129D** | **Gary Tan Jun Xian** |
| **1603477I** | **Goh Wei Kang** |
| **1602118J** | **Oliver Choy Chen Fung** |
| **1603170J** | **Tang Xin Wei** |
| **1605873G** | **Timothy Lua** |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 30/10/2017 | 0.1 | First Version | All |
| 3/10/2017 | 0.2 | Finish Part 1 and 2 | All |
| 4/10/2017 | 0.3 | Beta Version | All |
| 8/10/2017 | 1 | Final Version | All |

Table of Contents

1. Introduction
   1. Objectives and scope of the project………………………………..………..………39
   2. Assumptions and constraints………………………………………...…..….………39
   3. Definitions and acronyms…………………………….………………….….………39
2. Roles and responsibilities………………………………………………………………...40
3. Estimates and project schedule…………………………………………………………..40
   1. Work breakdown structure………………………………………………………......40
   2. Project schedule……………………………………………………………………..42
   3. Budget summary…………………………………………………………………….43
4. Risk management plan…………………………………………………………………...44

Project Plan

1 Introduction

1.1 Objectives and scope of the project

The objective of the project is to produce a hotel management system for the hotel Delonix Regia, the list of core features that will be developed are Customer relationship Management module, Housekeeping Management module, Room Booking Software modules, Room and Facilities Management module, and Human Resource Management module.

The items we plan to deliver at the end of this project is the hotel management system, a server hosting this system, and 5 computers for the hotel.

1.2 Assumptions and constraints

* Manpower is limited for both client and developer side
* Hotel Staff is not trained in computer IT skills
* Before the implementation of the system, there was no other system in place for the hotel
* Time, money, manpower
* Hotel’s budget constraints
* Other ongoing modules
* Other outside commitments of members e.g. CCAs, part-time jobs

1.3 Definitions and acronyms

CRM - Customer Relationship Management

HRM - Human Resource Management

RBS - Room Booking Software

RFM - Room and Facilities Management

HKM - Housekeeping Management

FAQ - Frequently Asked Questions

CCA - Co-Curricular activity

RPM - Revolution Per Minute

RAM - Random Access Memory

HDD - Hard Disk Drive

DDR - Double Data Rate

MB - MegaBytes

TB - TeraBytes

GB - GigaBytes

2 Roles and responsibilities

|  |  |
| --- | --- |
| 1.1 Objective and Scope of the Project | Timothy, Xin Wei, Oliver, Wei Kang |
| 1.2 Assumption and Constraints | Gary |
| 1.3 Definition and Acronyms | Xin Wei, Wei Kang |
| 2 Roles and Responsibilities | Timothy, Oliver |
| 3.1 Work Breakdown Structure | Wei Kang, Gary |
| 3.2 Project Schedule (Gantt Chart) | All |
| 3.3 Budget Summary | All |
| 4. Risk Management | All |

3 Estimates and project schedule

3.1 Work breakdown structure

**Phase 1:**

* Planning of all the modules (All)
* HM (Gary)
* RFM (Xin Wei)
* RBS (Timothy)
* HRM (Oliver)
* CRM (Wei Kang)

We decided that each of our members is responsible for taking one module each so that we are some efficient with the time and manpower we have.

* Requirement of Analysis

We conducted interviews with the hotel staff and see what they need, the functions they require and what problems they are facing right now. We need these in order to help them as best we can.

* Designing of Features

After getting the requirements from the clients we are able to get an idea of what they want for the design of the product they want.

* Implementing (All)

For this part, each member is programming each of the modules they are responsible respectively.

* Testing (All)

Everyone in the team will help test out each other’s modules to find out any bugs/errors so that we are able to rectify any errors.

**Phase 2:**

* Planning of Modules

Everyone in the team will see if they need to alter their modules that they are responsible for.

* Review Requirements

We will check with the clients again whether we have fulfilled all their requirements and double check with our the modules that each of us is doing respectively.

* Improve Design

After reviewing the requirements again, checking with the clients whether they like the design we come up with if not we are to improve on the design to their preference.

* Implementation

For this part, each member is programming each of the modules they are responsible respectively.

* Testing

Everyone in the team will help test out each other’s modules to find out any bugs/errors so that we are able to rectify any errors.

**Phase 3:**

* Bug Testing

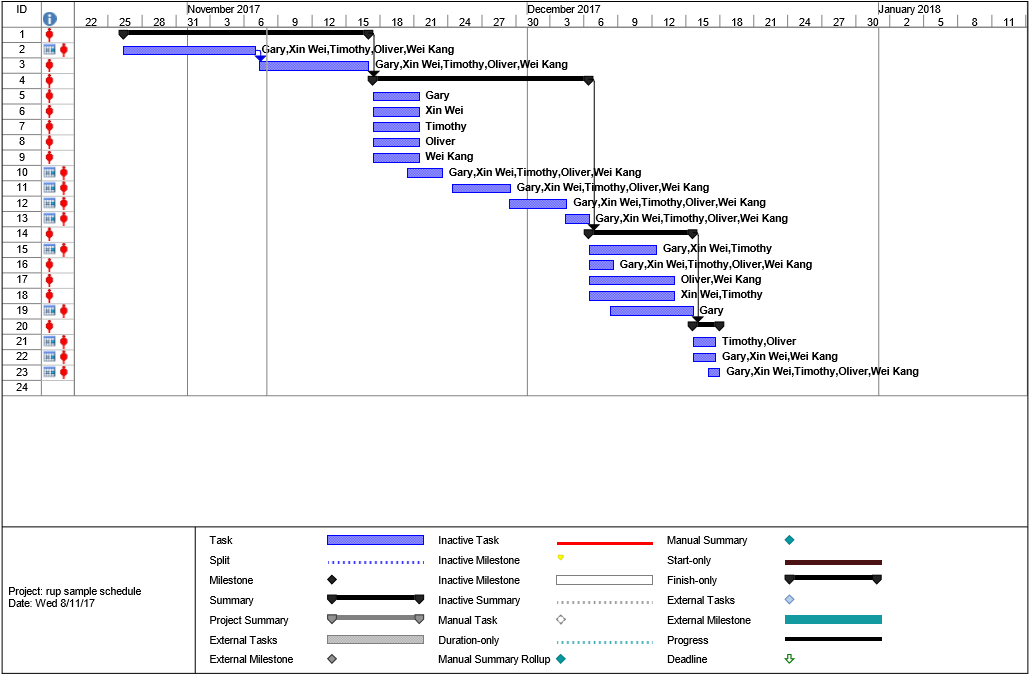
After all the modules are developed and compiled together into a complete system, the team will do multiple comprehensive tests to find out any errors or bugs in the system.

* Maintenance

The team will fix any errors or bugs faced by the client and also improve the performance of the system with new features.

* Deployment

After finalising and fixing all the bugs found in the bug testing, we will release the final version of our software to the client. Deployment also includes other activities like preparing the hardware and software to the server and computers and other subsequent activities. Finally, installation and activation of the software will be done by us at the hotel.

3.2 Project Schedule (Gantt chart)

3.3 Budget Summary

Application Server X1

HP ProLiant MicroServer Gen8 Ultra Micro Tower Server (783958S01)

* $976.66
* 2.3 GHz Intel Xeon E3-1220L
* 4 GB DDR3
* 2 TB 7200 rpm Hard Drive

*https://www.amazon.com/HP-ProLiant-MicroServer-Server-783958S01/dp/B00LGJ9NHK*



Microsoft Server 2016 software

* $684

Desktop workstation & Monitor X5

(Dell Inspiron Small Desktop 3268) + Monitor(Dell Monitor E2016HV) + Keyboard + Mouse

* $958 Per computer - $4790
* 7th Generation Intel® Core™ i5-7400 processor (6MB Cache up to 3.50 GHz)
* 8GB RAM, DDr4, 2400MHz
* 1TB 7200 rpm HDD

*http://www.dell.com/sg/p/inspiron-3268-desktop/pd?ref=PD\_Family#overrides=w21685315ssgw10:6932~AG-MONE2016HV*

Microsoft Office

* $135 per year

Microsoft Visual Studios

* Free

Developer’s Salary X5

* $12/hr - per developer
* 8hr per day for 50 days
* $4,800 per developer - $24,000

**Total Budget: $30,585.66**

4 Risk Management Plan

1. Using Agile model we are able to adapt to any changes from the client’s requirements.
2. Minimise any errors of the product with regular testing at any phase.
3. Miscommunication with the clients, by using Agile model reduces the risk of miscommunication between the developers and clients by meeting often.
4. To minimise any risk to clients not being able to understand the product, frequent meetings with the developers would help to alleviate this potential risk.
5. If anything were to happen to any of the developers, this will lead to not being this project in time. So in order to solve this, we have a backup member to prevent this from happening as the last resort.
6. If the client does not have enough resources for the developers, the developers have adapted to the resources that we have access to.
7. Using the agile model, the developers have come up with a Gantt consisting of deadlines of the task so that they manage their time properly.

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No:**  1605129D | **Student Name:**  Gary Tan Jun Xian |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:**  Everyone sticks by their word, and speaks out when something is clearly wrong instead of silently accepting the problem. |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_\_**Gary Tan Jun Xian **\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_**8/11/2017 **\_\_\_\_\_\_\_\_\_\_\_\_**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No:**  1603170J | **Student Name:**  Tang Xin Wei |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:**  This group is hardworking and they are fun to work with. |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_\_**Tang Xin Wei **\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_**8/11/2017 **\_\_\_\_\_\_\_\_\_\_\_\_**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No:**  1605873G | **Student Name:**  Timothy Lua |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:**  Parts assigned to group members are done timely of are of standard. |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_\_**Timothy Lua **\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_**8/11/2017 **\_\_\_\_\_\_\_\_\_\_\_\_**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No:**  1603477i | **Student Name:**  Goh Wei Kang |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Oliver Choy Chun Feng | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:**  Great teamwork in overall. Best project meetings ever. |
|  |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_Goh Wei Kang\_\_\_\_\_\_\_\_ Date: \_\_\_\_**8/11/2017 **\_\_\_\_\_\_\_\_\_\_\_\_**

Temasek Polytechnic

School of Informatics and IT

**Diploma in Information Technology (IT)**

Team/Peer Evaluation

|  |  |
| --- | --- |
| **Project Title:**  Delonix Regia Hotel Management System | |
| **Student No:**  1602118j | **Student Name:**  Oliver Choy Chen Fung |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the overall team performance against each criterion. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Team spirit | **1** | **2** | **3** | **4** | **5** |
| Overall effectiveness | **1** | **2** | **3** | **4** | **5** |
| Rewarding experience | **1** | **2** | **3** | **4** | **5** |
| Team productivity | **1** | **2** | **3** | **4** | **5** |
| Process quality | **1** | **2** | **3** | **4** | **5** |
| Product quality | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the contribution of each team member (including yourself). Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rate the quality of work (including timeliness) of each team member (including yourself). Circle one number from 1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |
| Rate the help and support you have received from each team member. For yourself, rate the support and help you have given to other team members. Circle one number from  1 (inadequate) to 5 (superior) | | | | | |
| Myself | **1** | **2** | **3** | **4** | **5** |
| Gary Tan Jun Xian | **1** | **2** | **3** | **4** | **5** |
| Tang Xin Wei | **1** | **2** | **3** | **4** | **5** |
| Timothy Lua Wei Sheng | **1** | **2** | **3** | **4** | **5** |
| Goh Wei Kang | **1** | **2** | **3** | **4** | **5** |

|  |
| --- |
| **Comments:** |
| **All my groupmates were terrific from start to end. All of us came for all the group meetings.** |
|  |
|  |
|  |
|  |
|  |

**Signature: \_\_\_**Oliver Choy Chen Fung **\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_**8/11/2017 **\_\_\_\_\_\_\_\_\_\_\_\_**