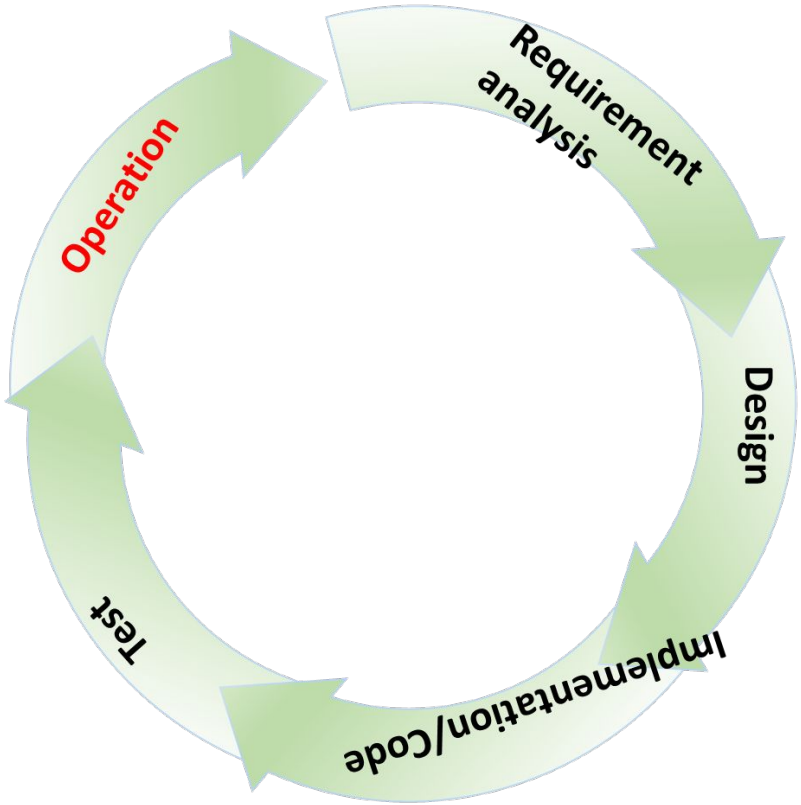
Music Industry Management System

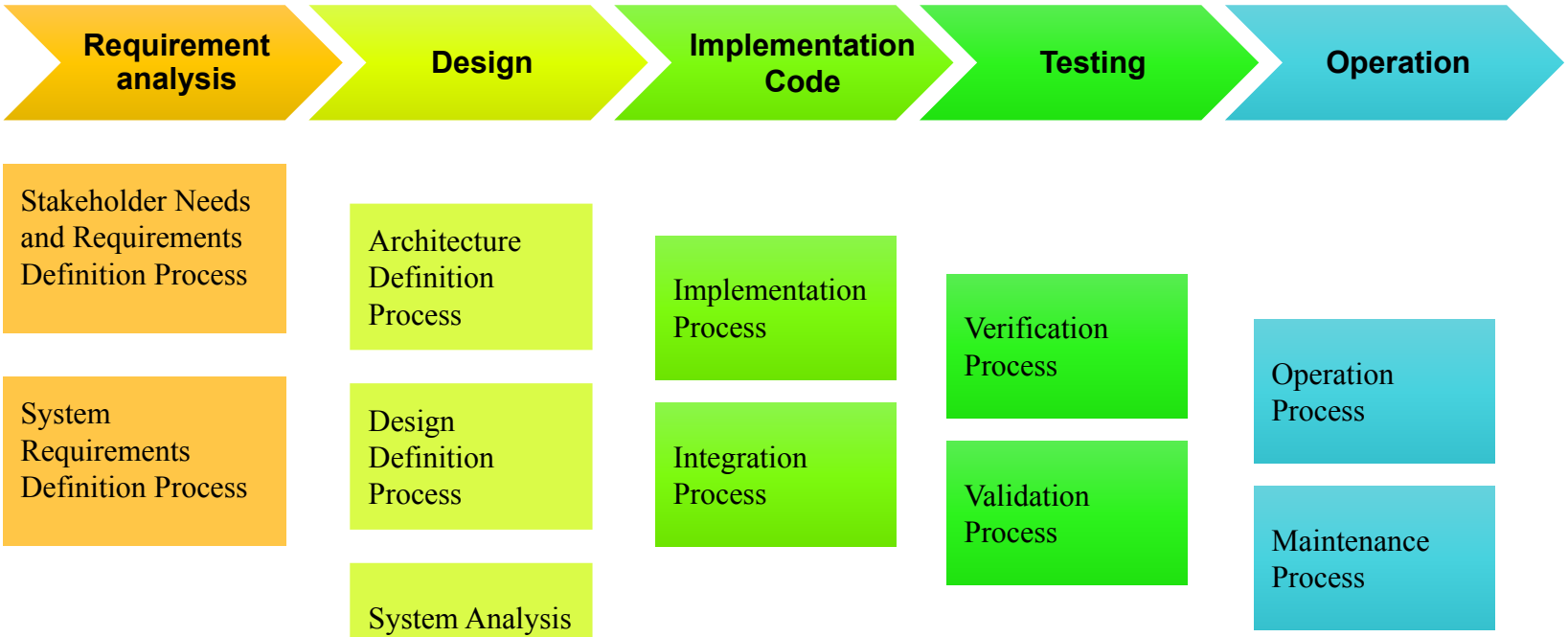
Name: 楊佑鴻 YuHung
Student ID: 111598031

SDLC



- Requirement analysis
- Design
- Implementation / Code
- Test
- Operation

SSDLC Phases



Require ment

Stakeholder Needs and Requirements Definition Process

Requirements (input)	Company previlege & assurance	system availability	Copyright of product	Billing of product	All relative systems records	Requirements information
Technique	SN-1.1/1.3/2.5/4.1/4.2: Identify previleged views SN-1.2/2.1/2.7/3.2/4.3: Define function: all views SN-5.1: Analyze previleged views needed by company	SN-1.2/2.1: Define security availability SN-1.3/2.2/3.2/4.1/4.2: Identify security availability SN-5.1: Analyze system availability	SN-1.1/1.3/4.1/4.2: define security concept SN-1.2/2.1/2.7/4.3: define security concept SN-5.1: Analyze security concept	SN-1.1/1.3/4.1/4.2: Identify security concept SN-1.2/2.1/2.7/4.3 :define security concept SN-5.1: Analyze security concept	SN-1.1/1.3/4.1/4.2: define records: security records SN-1.2/2.1/2.7/4.3: define records: security records SN-5.1: Analyze relative records	SN-1.2/2.1/2.7/4.3: define requirement: security requirement SN-5.1: Analyze security requirement
Output	Priority & Assurance views	security of system availability	Copyright security concept	Billing security concept	System security Records	Requirements document

Require ment

System Requirements Definition Process

Requirements (input)	Priority & Assurance views	security of system availability	Copyright security concept	Billing security concept	System security Records	Requirements document
Technique	SR-1.1/1.2/1.3/2.1/2.2 /3.2: Define boundary of views SR-1.4/4.1: Identify security aspect of views SR-3.1/4.1/4.3: Analyze security concerns of views	SR-1.1/1.2/1.3/2.1/2. 2/3.2: Define boundary of security availability SR-1.4/4.1: Identify security aspect of system availability SR-3.1/4.3: Analyze security concerns of system availability	SR-1.1/1.2/1.3: Define security model SR-1.4: Identify security model SR-3.1/4.1: Analyze security model	SR-1.1/1.2/1.3: Define security model SR-1.4: Identify security model SR-3.1/4.1: Analyze security model	SN-1.1/1.2/1.3/2. 1/2.2/2.3: Define security records for system SR-1.4: Identify security records for system	SN-1.1/1.2/1.3/2.1/2.2/2. 3: Define security requirements to write in to document SR-1.4: Identify security requirements to write in to document
Output	Priority & Assurance views	security of system availability	Copyright security model	Billing security model	System Records (Log)	Documentations (testing strategy, technique and tool to be used etc.)

Design

Architecture Definition Process

Requirements (input)	Priority & Assurance views	security of system availability	Copyright security model	Billing security modelt	System Records (Log)	Documentations (testing strategy, technique and tool to be used etc.)
Technique	AR-1.1/1.5/2.3: Identify security architecture AR-1.3/1.4/2.2:Define the pages for user/admin/musician AR-2.5/3.4: Spring boot/JS AR-6.1: Analyze: authorities, accountabilities, and responsibility is enough	AR-1.1/1.5/3.5: Identify security availability AR-1.3/4.2/4.5: define the scope of availability AR-6.6/6.7: Analyze: authorities availability, security availability	AR-1.1/1.2/1.5/2.3: Identify copyright of music AR-2.2/2.4/3.1/3.4: define model for copyright security AR-5.1/5.4/6.6/6.7:Analyze: select appropriate model for copyright security	AR-1.1/1.2/1.5/2.3: Identify billing of music AR-2.2/2.4/3.1/3.4: define model for billing security AR-5.1/5.4/6.6/6.7: Analyze: select appropriate model for billing security	AR-1.1/2.3/3.5: Identify message that need to be record availability AR-1.3/4.2/4.5: define the scope of necessary records	AR-1.1/2.3/3.5: Identify requirement for documentations AR-5.1/5.2/5.3/6.2/6.6:Analyze: system architecture, security model documentation
Output	Priority and assurance views: severity, priority level	High availability: re-syn time	Copyright security model: severity, priority level	Billing security model: severity, priority level	System Records (Log)	Documentations (testing strategy, technique and tool to be used etc.)

Design

Design Definition Process

Requirements (input)	Priority and assurance views: severity, priority level	High availability: re-syn time	Copyright security model: severity, priority level	Billing security model: severity, priority level	System Records (Log)	Documentations (testing strategy/methods, contraintsand tool to be used etc.)
Technique	DE-1.2/1.6: define the ID/passwd authentication, 2-factor authentication DE-2.2: list all functionalities: musican view: copyright page User: copyright page can read after musican give the permission; register page: read/write; music progress: read Admin: copyright page can not write DE-2.6: design plan: mapping the function to archiecture function	DE-1.2/1.6: define system availability to reach hight level DE-2.6/3.3: design plan: mapping the function to reach high availability function	DE-1.1/1.3/1.5: design copyright security model: authorized music DE-1.4/1.5/2.5: define interfaces for copyright model DE-4.1/4.3/4.4: Analyze: provide security-relevant information for model	DE-1.1/1.3/1.5: design billing security model: authorized music DE-1.4/1.5/2.5: define interfaces for billing model DE-4.1/4.3/4.4: Analyze: provide security-relevant information for model	DE-1.3/1.5/2.5: design security records level: normal, warming, dangerous DE-4.1/4.2/4.3/4.4: Analyze: security-relevant information to record	DE-1.3/1.5/2.5: design security model documentations: copyright, billing DE-4.1/4.2/4.3/4.4: Analyze: security-relevant information for model, traceability of the security aspects
Output	Design controls of priority and assurance model: severity, priority level	Design high availability: re-syn time	Design copyright security model : severity, priority level	Design billing security model: severity, priority level	Design system records (Log)	Documentations (testing strategy/methods, contraintsand and tool to be used etc.)

Design

System Analysis Process

Requirements (input)	Design controls of priority and assurance model: severity, priority level	Design high availability: re-syn time	Design copyright security model : severity, priority level	Design billing security model: severity, priority level	System Records (Log)	Documentations (testing strategy/methods, contraintsand tool to be used etc.)
Technique	SA-1.3/1.5: define the user can't modify the music data. SA-1.4/2.2: use tool:Checkmarx tool to do security analysis SA-3.1/3.2: using tool: mantis to tracing the issue	SA-1.1/1.3: define high availability for system SA-3.1/3.2: use traceability to maintain high availability	SA-1.1/1.2/1.3/1.5/1.6 : Identify and validate copyright security to protect copyright SA-3.1/3.2: use traceability to check modify for model	SA-1.1/1.2/1.3/1.5/1.6: Identify and validate copyright security to protect copyright SA-3.1/3.2: use traceability to check modify for model	SA-1.1/1.3/1.4/1.5/1.6: Identify analysis result that need to be record SA-2.3/2.5: Record the results of the security aspects of system analysis	SA-1.1/1.3/1.4/1.5/1.6: Identify analysis result that need to write into documentation SA-2.5/3.1/3.2: Analyze: system analysis results, system analysis information items
Output	Controls of priority and assurance model: severity, priority level	High availability: re-syn time	Copyright security model : severity, priority level	Billing security model: severity, priority level	System Records (Log)	Documentations (testing strategy/methods, contraintsand and System Analysis report.)

Implementation Code

Implementation Process

Requirements (input)	Controls of priority and assurance model: severity, priority level	High availability: re-syn time	Copyright security model : severity, priority level	Billing security model: severity, priority level	System Records (Log)	Documentations (testing strategy/methods, contraints and System Analysis report.)
Technique	IP-1.1/2.1/3-1: development testcases IP-2.1: use java to coding IP-2.2: implement user and stakeholders guide IP-2.3/3.2: each function need to relative to specific requirement by using document on tool: JIRA	IP-1.1/1.2/1.3/2.2: develop system with great availability IP-3.1/3.2/3.3: use documentations to trace any modify to the system	IP-1.1/1.2/1.3/2.2: develop system with designed model	IP-1.1/1.2/1.3/2.2: develop system with designed model	IP-1.1/1.2/1.3/2.2: record informations when develop system IP-2.4/3.1: record system log that meet the requirments	IP-1.1/1.2/1.3/2.2: document informations when develop system IP-3.1/3.2/3.3: Analyze: system implementation results, system implementation information items
Output	Implementation of controls of priority and assurance subsystem: severity, priority level	High availability: re-syn time	Implementation of copyright security subsystem: severity, priority level	Implementation of Billing security subsystem: severity, priority level	System Implementation Records (Log)	Documentations (testing strategy/methods, contraints and System Implementation report.)

Implementation Code

Integration Process

Requirements (input)	Implementation of controls of priority and assurance model: severity, priority level	High availability: re-syn time	Implementation of copyright security model : severity, priority level	Implementation of Billing security model: severity, priority level	System Implementation Records (Log)	Documentations (testing strategy/methods, contraints and System Implementation report.)
Technique	IN-1.1/2-2/2-3: development integration testcases IN-3.1: testing using tool: mantis to records. Manually/automation. Automation uses tool: selenium to develop automation steps IN-3.2: use mantis to trace	IN-1.1/1.3/2.2/: Integrate system to maintain high availability IN-3.1/3.2/3.3: use record to trake integration status	IN-1.1/1.3/2.2/: Integrate system to maintain security factor IN-2.1/2.3: Realize or adapt system to integrate each other	IN-1.1/1.3/2.2/: Integrate system to maintain security factor IN-2.1/2.3: Realize or adapt system to integrate each other	IN-1.1/1.3/2.2/: record necessary system integrated results IN-2.4/3.1/3.2/3.3: record implementation results	IN-1.1/1.3/2.2/: documentate necessary system integrate requirements IN-2.4/3.1/3.2/3.3: document Integration results, Manually/automation. Automation results, Integration information
Output	Music industry management system	High availability: re-syn time	Music industry management system	Music industry management system	System Integration Records (Log)	Documentations (testing strategy/methods, contraints and all kind of process documents include.)

Testing

Verification Process

Requirements (input)	Music industry management system	High availability: re-syn time	System Integration Records (Log)	Documentations (testing strategy/methods, contraints and all kind of process documents include.)
Technique	VE-1.1 development TC with requirement, architecture design VE-1.3/2.2/3.5. Manually/automation. Automation uses tool: selenium to develop automation steps VE-1.4/1.5: pass rate must be 90% without amy P1/P2 issues for deliver VE-1.6: only support 100 concurrent users VE-2.1: using checkmax to do vulnerability scan VE-3.1/3.4: use mantis to record the results	VE-1.1/1.2/1.3/1.4/1.6/2.1: verificate system to maintain high availability VE-3.1/3.2/3.3: use the verification process result to check if there is any potential issues	VE-1.1/1.2/1.6/2.1: records verificated system results VE-3.1/3.2/3.3: record all verification process results in order to prevent issue	VE-1.1/1.2/1.6/2.1: document verificated system results VE-3.3/3.4/3.5: document traceability, verification results manually/automation. automation results, verification process results
Output	Music industry management system	High availability: re-syn time	System Verification Records (Log)	Documentations (all kind of security -focused process documents include that verificated by stakeholers.)

Testing

Validation Process

Requirements (input)	Music industry management system	High availability: re-syn time	System Verification Records (Log)	Documentations (all kind of security -focused process documents include that verificated by stakeholers.)
Technique	VA-1.1 development VP with requirement, architecture design VA-1.3/2.2/3.5. Manually/automation. Automation uses tool: selenium to develop automation steps VA-1.4/1.5: pass rate must be 100% without P1/P2 issues to be accepted VA-2.2: using checkmax to do pentest VA-3.1/3.4: use mantis to record the results	VA-1.1/1.2/1.5/2.1/2.2: define and perform security validation procedures	VA-1.1/1.2/1.5/2.1/2.2: documentate validation process results VA-3.1/3.2: record validation process results to meet the requirements	VA-1.1/1.2/1.5/2.1/2.2: documentate validation process results VA-3.4/3.5: document traceability, validation results manually/automation. automation results, validation process results
Output	Music industry management system	High availability: re-syn time	System Validation Records (Log)	Documentations (all kind of security -focused process documents include that validated by stakeholers.)

Operation

Operation Process

Requirements (input)	Music industry management system	High availability: re-syn time	System Verification Records (Log)	Documentations (all kind of security -focused process documents include that validated by stakeholers.)
Technique	OP-1.1/1.2/2.1:develop the security operation to be incorporated into the system requirements, OP-2.4/2.5: perform system security operation if needed.	OP-1.1/1.2/2.1:develop the security operation to maintain high availability OP-2.4/2.5: perform system security operation in order th validate system meet the high availability	OP-2.4/3.1/3.2/4.2: record every operation that perform on the system	OP-3.3/3.4/4.1/4.3:document operation results manually/automation. automation results, operation process results that meet user request or not.
Output	Music industry management system	High availability: re-syn time	System Operation Records (Log)	Documentations (all kind of security -focused process documents include.)

Operation

Maintenance Process

Requirements (input)	Music industry management system	High availability: re-syn time	System Operation Records (Log)	Documentations (all kind of security -focused process documents include that validated by stakeholers.)
Technique	MA-1.1/1.3/1.4: identify security aspects of system maintenance and improve system security MA-2.1/2.3/2.4/4.4: Implement action to restore the system to secure operational status	MA-1.2/1.3/1.4: maintain system with tools to prevent security issues MA-2.1/2.2/2./4.1/4.2: review record to check if maintenance process that harm the availability	MA-1.2/1.3/1.4/2.2/2.3: record tool that use to maintain system and any issues when ending maintained. MA-2.2/4.1/4.2: record maintenance process results that is in correct status.	MA-1.2/1.3/1.4: Document tool that use to maintain system MA-4.4/4.5: document every maintenance results maintenance process results to check if system is in the safe status.
Output	Maintenance of Music industry management system	High availability and maintenance: re-syn time	System Maintenance Records (Log)	Documentations (all kind of security -focused process documents include.)