

**Bachelor of Science in Computing**

**Software Engineering**

**Year 2022-23 / Semester 2**

Internal Examiner: Wei Ren

Internal Moderator: Wenhao Fu

External Examiner: Maria Barry

Date: 24/04/2023

Duration: 2 Hours

Time: 10:00-12:00

Exam Weighting: 60%

Exam Delivery: Computer

**Instructions**

1. This paper contains 1 section with 1 question.
2. You must attempt all parts of the question.
3. All questions are marked out of 100.
4. Please write all answers on the script provided.
5. Clearly number all questions.
6. This is an open-book exam.

***Please do not turn over this page until instructed to do so****. The use of programmable or text storing calculators is expressly forbidden. Please note that where a candidate answers more than the required number of questions, the examiner will mark all questions attempted and then select the highest scoring ones.*

**Question 1**

Please answer the question based on the following scenario: You are managing a software development team that has been invited by “Dorset Bank” to develop an operating system for their ATM machines. The ATM machines should allow clients to withdraw money at any time and deposit cash or checks. Additionally, the machines should display the client's account balance. You have 4 weeks to do this project.

1. Select a development method for this project and provide the reason for your choice, (e.g. waterfall development method, agile development method, etc.).

**[14 marks]**

**I would choose the Agile methodology because of time constraints. Agile is a methodology focused on interactive development and delivery. The software is divided into smaller pieces (for example, withdraw money, deposit cash/ or checks and display account balance) which are called sprints, using this methodology I would assure my client would receive a working software according with the time constraints and if the requirements change would be easier to change the software in each sprint. Agile has many advantages and among them is the interaction between development team and client.**

1. Provide a comprehensive list of the steps involved in the software development process.

**[5 marks]**

**Requirements gathering**

**Design**

**Implementation**

**Testing**

**Deployment**

**Maintenance**

1. Write a requirement document based on the client's previous requirements, including functional requirements, interface requirements infrastructure requirements.

**[25 marks]**

**Functional requirements**

User should be able to safely login in its account.

User should be able to see account’s balance.

User should be able to withdraw cash from account.

User should be able to deposit cash/ checks in account.

User should be able to safely logout from its account.

**Interface requirements**

Interface should be user friendly and intuitive to navigate.

Interface should be responsive and items well placed on screen.

**Infrastructure requirements**

Software should be secure and data should be protected by encryption.

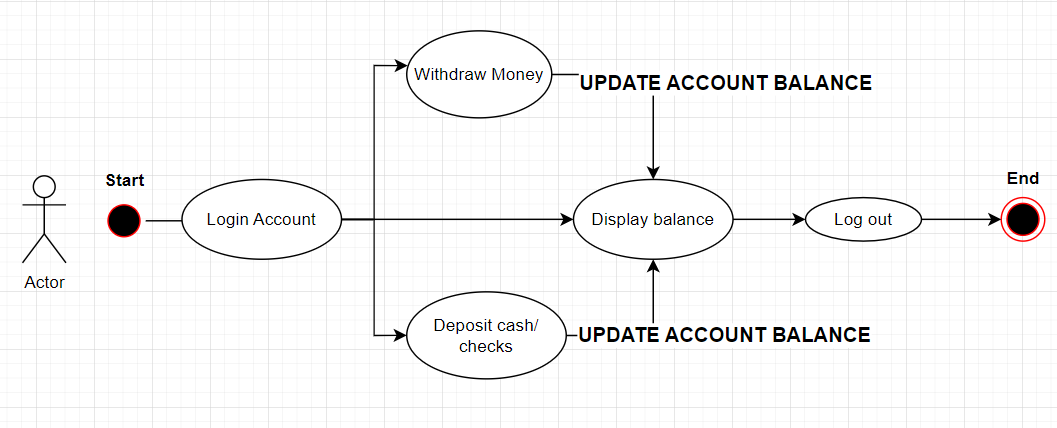
Software should be scalable and easy to expand.

Software should use an exclusive operation system to run.

1. Create a Use Case Diagram.

**[6 marks]**

**Use Case Diagram:**

****

1. Create a Gantt Chart to manage development process.

**[10 marks]**

**Requirements gathering –** 3days

**Design –** 4 days

**Implementation –** 2weeks

**Testing –** 4days

**Deployment –** 3 days

**Maintenance –** Not measure during development process

1. Provide an example code snippet, in any programming language, that includes the class name, function name, and any other necessary variables, to demonstrate how to achieve the goals of this project. You do not need to include the implementation details of the functions. Please show class diagrams or template code.

**[20 marks]**

**JavaScript programming language**

**Export default app () => {**

**Return <login>**

**}**

**export default login () => {**

**const [account, setAccount] = useState(null);**

**const [password, setPassword] = useState(null);**

**const checkAccount () => {**

**if (account & password != null) {**

**return <Account/>**

**} else {**

**Return <div> Account not found! </div>**

**} }**

**return (**

**<div>Account number: {setAccount(*userInput*)} </div>**

**<div>Password: {setPassword(*userInput*)} </div>**

**)**

**}**

**export default account () => {**

**Const [amount, setAmount] = useState(0);**

**Const [balance. setBalance] = useState(0);**

**Let newBalance;**

**Const cashWithdraw () => {**

**newBalance = amount - balance;**

**setBalance(newBalance);**

**}**

**Const Deposit () => {**

**newBalance = amount + balance;**

**setBalance(newBalance)**

**}**

**return (**

**<div>**

**Account balance = {balance}**

**</div>**

**<div>**

**<Input placeholder= “Enter Withdraw amount”>**

**Withdraw: setAmount(*userInput*) </input>**

**</div>**

**<div>**

**<Input placeholder= “Enter Deposit amount”>**

**Deposit: setAmount(*userInput*) </input>**

**</div>**

**)**

**}**

1. Provide a detailed explanation of the validation and verification process that should be followed to ensure the successful completion of this project.

**[15 marks]**

**Requirements validation:** Validate and verify the requirements with client to comply with client needs, validating and verifying the requirements with client will assure we are developing a software that will fulfil the client needs.

**Design review:** Review the software interface with client to assure the interface requirements are being accomplished and the final software is going to be what the client expect.

**Code review:** Review the front-end and backend of the software making sure the functional requirements and infrastructure requirements are being satisfied and the final software is in order with what the client expect.

**Testing review:** Create and review test routines to assure the software is well design and implemented. If any bug or error is encountered it should be passed to the development team to rapid resolution and software update.

**Deployment review:** Deploy the software to a test server to assure it is working software and all the requirements are being satisfied, such as data protection and encryption.

**Maintenance:** Create a maintenance report to assure all the functionality of the software is working, and any bug or error is promptly addressed and resolved. The maintenance should be also responsible for keeping the software up to date with any new update on the tools utilized for the development, such as the security and encryption.

1. Create a repository on GitHub and upload your answers to it. Please note that the answers on GitHub will not be considered for marking, so it is essential to submit all your answers to Moodle before the exam deadline.

**[5 marks]**

**Link to GitHub:** [JosueDilmo/24061\_SoftwareEngineeringExam: Software Engineering Exam 2023 (github.com)](https://github.com/JosueDilmo/24061_SoftwareEngineeringExam)

**[Total 100 marks]**