Validation Project execution

# Use Case #13

## Use Case Title

Provide suitable design for given requirements and describe design patterns and non-functional requirements

## Use Case Description

Create a design of the system and briefly describe its components and applied design patterns. Design of the system shall address all required nonfunctional and functional requirements. Components of the system shall be traced to requirements and applied design patterns or approaches to non-functional requirements

## Participant information

|  |  |
| --- | --- |
| **Item** | **Description** |
| Name | Ivan Shevtsiv |
| Email | ishevt@softserveinc.com |
| Job Profile + level (ex. Node.js Middle, Java Senior) | Application Architect (L4) |
| Date | 9/1/2023 |
| Generative AI used (none, ChatGPT 3.5, GPT-4, StarCoder, etc) | GPT-4 |

## Experience / Maturity levels definition\*

In some sections of this document, you will be asked to provide self-evaluation mark regarding your experience or maturity with some area, topic, tool, methodology. Use the definition below to provide your mark.

**None**: Not familiar with the subject matter and has no prior experience or knowledge in the area, tool, methodology

**Beginner**: Basic familiarity with the tool, methodology or concept, knowing about it and occasionally using it for simple use cases, tasks

**Competent**: Regularly uses the tool, methodology or concept in everyday duties and have a good level of proficiency and understanding.

**Proficient**: Advanced experience and knowledge, utilizing the tool, methodology or concept in advanced activities, understanding its intricacies, and being able to educate others about it.

**Please note that you need to open the file in the desktop application or work with the downloaded file to make inline dropdown lists work inside cells.**  **If you don’t have desktop version, please select and insert one of these definitions into the cells: None, Beginner, Competent, Proficient.**

## Use Case self-assessment\*

|  |  |
| --- | --- |
| **Area** | **Experience / Maturity**\* |
| What is your experience with Use Case? | Competent |

## Tools and Methodologies

To keep consistency Allowed to use only specified tool and methodologies during development.

**Reference architecture catalogs**

|  |
| --- |
| **Tool / Methodology** |
| <https://learn.microsoft.com/en-us/azure/architecture/> |
| <https://cloud.google.com/architecture> |
| <https://aws.amazon.com/architecture/> |
| <https://microservices.io/patterns/microservices.html> |
| <https://martinfowler.com/architecture/> |

**All technologies**

|  |  |
| --- | --- |
| **Tool / Methodology** | **Experience / Maturity**\* |
| Plant UML / Mermaid or any markdown or text-based UML diagram approaches | Competent |
| Draw.io | Competent |
| Git | Competent |
| Generative AI – ChatGPT/Copilot/etc | Beginner |
| GitHub | Competent |

## Exercise for design

A client has a desktop software developed for decades for tax calculation. This software allows to enter income and loss on forms into entry fields and tax calculation engine will calculate based on tax rules how much money person owes to state or state owes to person in case more taxes payed. When person is happy with calculation it shall be possible reliably to send it to financial regulator web service and ask for results.

Example

Client has desktop software but end result can be seem here as example <https://www.taxact.com/tools/tax-calculator>

**Requirements:**

* As a user I want to enter my personal information (Name, Surname, Tax ID, Single or married, are there any dependents) on basic info form.
* As a user I want to enter Tax Year, and answer the questions “Did you use software to complete last year’s taxes?” on basic info form.
* As a system I want to be driven by configurable rules so additional questions may be asked in case of answer
  + In case you answer as single additional Do you qualify as head of household? Are you a qualifying surviving spouse
  + In case you answer married Are you filing jointly?
    - In case you file jointly year of birth of the spouse is needed
* As a user I want to enter on income page - Taxable Wages, Spouse's taxable wages, Federal taxes withheld and estimated payments, Interest income, Unemployment income
  + In case I am married and we file taxes jointly the same information shall be entered for spouse
  + There might be more additional information required on the form and shall be configured like Dividend Income
* As a user I want to enter deductions related information on deductions page
* On each page I would like to see information on return as Forecasted Refund, Filing Status (married or single); Total Income, Adjustments, Deductions considered for forecasted refund
* Tools shall be implemented as web application with mobile support
* For more functional requirements use link provided
* Web interface is dynamic and what is show managed by tax rules or form definition in tax rules
* Fields on the form can trigger recalculation resulting in numeric values recalc (e.g. Forecasted Refund) or answers to questions in interface activate new fields
* It shall be possible for accountant to change tax rules and form at any time without rebuild of the software
* I would like to avoid any downtime of software or software shall handle it
* Financial data privacy is important. No data lost and data shall be reliable.
* System shall be secure and every use shall register and login
* It shall be possible to run Calculation engine as separate micro-service from forms or any other components
* Recalculations shall be fast under the load, sequence of recalculations matters

## Acceptance criteria

* Design MUST addresses all functional and non-functional requirements.
* Design COULD contain optional or redundant element, in such case explanation why MUST be provided.
* All components of the system MUST be described, technology proposed for them MUST be included.
* Behavior of the system MUST be provided for at least one major use case.
* Any description SHOULD contain project specific or reference to requirements, e.g. proposal to use AWS Lambda must be rationalized in terms of given case or requirements
* You COULD provide different technological or architectural optional as variability in components description
* You COULD use any cloud or any open-source libraries or well-known proprietary technology or libraries.
* Complete AI chat MUST be stored in provided repository.
* Diagrams MUST be provided in any suitable form as file – e.g. Draw.io, Mermaid, PlantUML, <https://swimlanes.io/> etc.

Instructions

1. Track architecture in provided git repository.
2. Please try to document everything based on SoftServe’s Architecture Vision (Views & Beyond) template in plain text file. In case you need to use table use bullet list which corresponds to row or use markdown syntax. You may use editor of your choice (e.g., Visual Studio Code)
3. Fill sections – design constraints, quality attributes, major use cases and future solution architecture
4. Define measurable NFRs.
5. Provide component diagram.
6. Describe behavior of the system for main use case
7. Describe how each component will be implemented and deployed.
8. Describe which patterns will be used for this system design and rationalize them.
9. Design of the system shall address all required nonfunctional and functional requirements.
10. Use any materials you handy even from previous discoveries.

## Use Case Implementation

### Start

* Create new repository. Use main branch for your commits. Make sure to make first commit C#1 with README.md, as time of this commit will be considered as project start. Add commit message “Architecture Vision#1 Start”.

Tools

* Install or open draw.io in web. When working with diagrams in AI tools use PlantUML syntax
* Create a new file for each section. Markdown e.g. <https://stackedit.io/> or plain text can be used
* Use <https://swimlanes.io/>

**Finalization**

* Store AI chat conversation for each section or completely as plain text “chat.txt” file and commit it to the repository. If you did without AI support create an empty file.
* Structure all deliverables as separate md or txt files – e.g. “design constraints”, “quality attributes”, “major use cases” and “future solution architecture”
* Put diagram describing components of your system in repo as “components.drawio” if created with drawio components.plantuml if created with PlantUML syntax with AI
* Use <https://swimlanes.io/> synaxt or PlantUML syntax for behavior of your system and document main sequence of operations and put it as “sequence.txt”
* Commit to repository as “UC#2 Chat”.
* Time between C#1 and C#2 metric for Infrastructure setup productivity

|  |  |
| --- | --- |
| **Repository** | **Link** |
| Provide Link to created GitHub repo | https://github.com/kehas63290/GenAI |

## Participant feedback

|  |  |
| --- | --- |
| **Item** | **Description** |
| Chat GPT | Never expected that level of generation, was interesting to generate diagrams with plantUML, the interesting thing that I’ve spotted is that the chat is not always tracks the context and could easily generate something as a separate solution for the same case. |
|  |  |
|  |  |
|  |  |

## Expert feedback

|  |  |  |
| --- | --- | --- |
| **Criterion** | **Evaluation\*** | **Notes** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**\*4 - excellent (smth was done with extra mile);**

**3 - good (everything was done according to acceptance criteria);**

**2 – bad (has gaps, need some extra time for completion);**

**1 - the worst (a lot of mistakes/no result/step was omitted, must be disqualified).**

|  |
| --- |
| **Comment** |
|  |

|  |  |
| --- | --- |
| **Final Score** | **Evaluation\*** |
| Implementation score (solution suitability in accordance with the functional criteria) |  |
| Quality score (quality of the implemented solution) |  |