Contributions to Team 4

This document holds basic conversation history on Slack between members of Team 4. It is not meant to represent a module "reflection" nor an essay or report but is provided (and categorised) for insight into team collaboration.

Contents

Create a Team Contract

- 13 Aug 2021: Just a note on the "revision in document name". According to TOGAF, minor numbers represent revisions, while whole numbers "1.0", "2.0" represent "sign off" stage. So, since we are going to touch on TOGAF as part of secure software, I thought it would be good to get some practice with the principle 🙂 Although... technically, we'd store such documents in SharePoint and use the version numbers provided by such a storage tool. @Andrey, @Taylor, Our initial contract lists "version referencing" in our "Policies and Procedures" section, perhaps we need to slightly clarify the approach (or remove reference to versioning?). An approach used often is to provide a "Version History" table in the document. For example, see the next iteration of the contract

  > *Reply from Andrey Smirnov: Your proposal to include the version history is great and it is also a practice that we use at ABN for our architecture deliverables. Let's go with that but try to keep our minds on the content of the document and not focus too much on the administration details*

Team Roles

- 13 Aug 2021: Happy to rotate the roles. BUT rotating the Project Manager style role (managing teams, project direction etc) may definitely be most disruptive because of the knowledge / momentum held by a PM. If we wish to rotate, I consider the concept of "self-organising" teams and how we may leverage that to our benefit--mindful to allow members who naturally possesses a "management" mindset to take the respective role. Concerning Technical Lead. Yes, I believe you are correct in this statement. As a team, we have liberty to freely discuss ideas and solutions, but the Technical Lead may posses deeper insights that "lead the discussion". Such leading must manifest the best solution, having considered alternatives, pros and cons of

different approaches or patterns, benefits of this or that framework, constraints on development or testing, or other hidden details.

> *Reply from Cathryn Peoples: Some really great ideas being bounced around here, Team 4. The time notion is important. From now until Week 6, there is likely to be a slower pace, especially with regard to the development side of work. After Week 6, the pace will likely pick up, certainly from the perspective of the development. This may influence the management strategy. These considerations that you are progressing through now will provide excellent material for reflecting on at a later stage of the module.*

Team Meeting Schedules

- 14 Aug 2021: Hi everyone, availability is pretty good each day of the week, I'm quite flexible with time. To fall in line with other members, after 4pm BST weekdays, anytime on weekends 16:07. @Andrey Smirnov happy with your proposal to change roles once (after unit 6)

- 14 Aug 2021: There is however value in meeting each other and verbally agreeing on the expectations because verbal communication may raise points not yet considered

General

Discussion why security is an afterthought

- 16 Aug 2021: I think one reason for the possible delay-security-considerations could be that organisations are unsure of "best-practices" for how to implement secure software development. For example, GDPR (which we dealt with in previous modules) sits at the management/organisation policy level and filters down to software through functional requirements; it deals with privacy and rights over the data from a public point of view. I think secure software development focuses more on the mechanisms developers ought to use to allow or prevent an information system's access to data. Mechanisms such as JWT (JSON Web tokens), encryption in transit (SSL/TLS), encryption at rest, the use of cryptography, authorisation frameworks (OAuth) including OpenID or hardware mechanisms such as firewalls or anti-virus programs. From this landscape, developers require guidance on the best tool for the job: there is no use utilising OAuth for a small 2-man business, or even encryption

where the data never travels out the organisation's domain--though these are good to implement nevertheless. A conclusion may be that the delay of developing secure software is a complex task given points such as:

1. Developers require unfettered access to their systems during development time. Restricting it through security layers, complicates testing and development
2. It is not always obvious what role/action combinations are fully required at the time of development.
3. Infrastructure team may not have implemented the required security mechanisms at the time of development--but absolutely must be in place at deployment.
4. Security between two or more information systems may be complicated because each system authenticates/authorises using different mechanisms. For example, delegating authority across domains to a customer's authorisation server.
5. Developers (or other teams) may be unfamiliar with security-related topics such as OAuth and even OWASP (which we're going to address in this module 🙂)

We can say that secure software is not trivial because it requires

- up-front consideration of roles and actions;
- implementation and understanding of supporting mechanisms;
- support from management, customers and internal teams;

Reply from *Andrey Smirnov*:

- 16 Aug 2021: I was just going over your post from earlier today, and I must say you have really hit the nail on the head with those 5 points provided in your analysis. It is disconcerting to me that each and every of your points apply to the situation that I am currently observing at the bank, even though it is safe to say that it is a mature organization that otherwise takes security extremely seriously. To elaborate:
    1. *Developers require unfettered access to their systems during development time*. Restricting it through security layers, complicates testing and development. -> Check. We have made it exceedingly difficult for our development teams to navigate the phases of their SD lifecycle. Things like requesting lower-level development environments, which one would expect to be fast-tracked, takes an ungodly amount of time at ABN.

Connecting to a “domain” service using the SOA integration pattern - we have other integration architectures such as EDA as well - involves going through a cumbersome onboarding process. The internal API gateway layer that handles infrastructure security and is positioned as a central point of SOA integration, while advertised as more lightweight than the previous ESB implementation, is anything but lightweight and has a dedicated team that has become so notorious for their tendency of making the lives of both the providers and the consumers of APIs difficult, that they have earned themselves a moniker “Internal Api Gateway Police”.

2. *It is not always obvious what role/action combinations are fully required at the time of development.* -> Also check.
3. *Infrastructure team may not have implemented the required security mechanisms at the time of development--but absolutely must be in place at deployment.* -> Again, check. The new security mechanisms are either broken, or not in place yet, and the legacy ones are forbidden except if they have a migration proxy in front of them / make use of the Strangler pattern.
4. *Security between two or more information systems may be complicated because each system authenticates/authorises using different mechanisms. For example, delegating authority across domains to a customer’s authorisation server.* -> Unfortunately, also check. Imagine a makeshift, unwieldy monolithic security application addressing every possible concern - session and token management, authentication, authorization - that is deeply integrated into hundreds of consuming applications. Now imagine this application being taken apart and broken into multiple “microservice-style” services, that are built according to modern standards like OAuth/OpenID, etc. The organization is still one foot deep in its legacy security landscape, another taking awkward first steps into the brand new future; designing solutions in this situation is anything but straightforward.
5. *Developers (or other teams) may be unfamiliar with security-related topics such as OAuth and even OWASP (which we’re going to address in this module* 🙂*)* -> I think you can already see a certain pattern in my comments.

- 16 Aug 2021: Reply to *@Andrey Smirnov* 🙂 it's not possible to respond to the question "why is secure software development an afterthought" by providing an unconsidered response like "Not sure. " So to facilitate a good discussion, it is better to provide broad thoughts that allow for richer considerations; provoking thought, inspiring research. So, the question is "why", and the initial response is "likely because of these points". Also, our discussions help us learn from each other and informs our reflections near the end of this module. We shall be as diamonds, sparkling with understanding 😀 (edited)

- 17 Aug 2021: Feedback from *Andrey Smirnov*: I like your statement that "there are many parts impacted by security requirements", and that it is not merely a concern of development teams. In bigger organizations certainly, I would expect many of these security requirements to be embedded in enterprise-level architecture guidelines, and integrated into the platform/infra architecture as well (especially in hybrid-cloud environments that mix on premise with private and public cloud services and need to ensure secure communication between those).

Discussion: Mapping Scrum Stages

- 19 Aug 2021: @Andrey Smirnov it's great to share observations that are well received 😊 because it challenges one to consider their own experiences and knowledge and research of Scrum."there are no formal 'phases'", yes. But there are, how do we say, "phases" of when actions absolutely must be accomplished, which basic reasoning concludes that management (the least educated about Scrum) realise the need for structure, steps, "phases". For example, every organisation needs:
    1. a (holy) Vision.
    2. Developing a Product Backlog.
    3. Breaking Product down into Sprints.
    4. Development.
    5. Testing.
    6. Feedback.
    7. Release.

  Though, I do agree with your statement "there is no [one] size fits all" approach; and that's exactly what a framework is: customizable. (Global Trust Association,

n.d., https://globaltrustassociation.org/is-scrum-a-framework-or-a-methodology/). I respectfully disagree with this statement "there is only sprint planning when you mostly talk on the level of individual user stories", because what is therefore the purpose of a Product Backlog? That product backlog has to originate from those who understand customers' needs in a product. For example, from management, product teams, marketing teams, customer-focused teams. Why? Because products should never be developed based solely on what developers think a customer requires. Indeed, experience shows that such "planning" often takes place a few sprints ahead of the current development sprint, and that such high-level plans are derived from business analysts, CTO's, dev leads, architects, test managers etc. But then again, since Scrum is a framework, perhaps some organisations choose to develop on-the-fly? Risky.

Knowledge Sharing

- 30 Aug 2021: I want to share this link, cause I thought it was quite useful https://www.whitesourcesoftware.com/most-secure-programming-languages/

- 6 Sep 2021: Hi guys, just thought this was interesting. On Netflix is a show titled “Not a Game” that features a number of Drs from Essex University 😊 the professors and lecturers had input into various aspects of game design and psychology

- 15 Sep 2021: Morning guys, I know we dealt with Regex's last unit, nevertheless I thought this person's opinion brings another perspective to the idea of regexes being evil or ;bad https://www.youtube.com/watch?v=FO0YdgSUCHo&t=592s

- 27 Sep 2021: https://softwarebrothers.co/blog/how-to-avoid-over-engineering/

- 5 Oct 2021: Hi guys, I came across this interesting video this morning concerning hardware security and how Microsoft is attempting to protect uses from hackers with Windows 11https://www.youtube.com/watch?v=tg9QUrnVFho&t=915s

- 8 Oct 2021: https://stackoverflow.com/questions/2357230/what-is-the-proper-way-to-comment-functions-in-python

Team Project

Understanding Customer Requirements

- So actually it will be quite interesting to address, because we have the Tech Lead for Part 2 (who is considering deliverables) and Architect for Part 1 (who is considering design). Each one has particular viewpoints on the system and we need to ensure that ultimately, the **customer's requirements** are met 100% (or as close as possible, with log of those requirements that cannot be delivered). In our module, we can argue that the "customer" is @cathryn.peoples /UoEO since they have provided the brief to our team.

Document Outputs

- 17 Aug 2021: *Question to Taylor Edgell:* I liked your references, thank you. The notion of a "lessons learned" document is interesting, one that we have previously "used". what is your experience with utilising such a document? Is there any tangible benefits as opposed to just documenting something for the sake of it?

  Reply from *Taylor Edgell*:

  - In regards to the "Lessons learned" document, this is less a document that we should keep, and more of a requirement that was recommended that the ISS have a system in place to allow for this functionality. I thought it would be good functionality to add into our application as it would allow us the potential to allow users to add new "learned lessons" and to search a database of "lessons" that have been added previously. We could potentially add a degree of access control, as only certain levels of clearance may only be able to view certain project reports. It would allow us to fulfil one of the potential requirements of the ISS. In essence a "Lessons learnt" document is an expanded version of a FAQ. I can see it as being invaluable to the ISS as it would give them a compendium of reflections of previous projects they have experienced.

Codio Structure

- 21 Sep 2021: Hi @Andrey Smirnov , yes I'm happy with the proposed folder structure however

  "/bin" or "/scripts" - I don't think we will require a "bin" folder because we're not generating any output in the sense that Cathryn will run the program inside Codio (or similar environment)

  "/conf" - I agree, this likely will not be need, but it does not hurt to create the structure anyways

  "/tests" - this is nice because we can direct Cathryn to this folder to specifically execute tests separately

Code-specific

- 10 Oct 2021: Hi guys, I worked on the project today and have made the following updates/additions/changes:

  1. Introduction of "Screen" to handle menu inputs. Please see the "screens" folder and especially the "start_screen.py" file
  2. Improve the authentication class
  3. Fix the issue with loading modules from separate directories. Please see the "main.py". Also, a number of "import" statements have to use "from x import y"
  4. Work on the menu.py file.

  …Please do try out the updated version from a terminal near you

- 10 Oct 2021: Please note that anything to do with MySql throws a FATAL exception within the Python environment (Segmentation failure) and the stack track shows that it's related to mysql. @Andrey Smirnov is it possible for you to look into why the environment throws this error? Perhaps mysql is not configured correctly or there is something in the "conf" file that needs to be updated?

  @Taylor Edgell code from menu.py has been moved into relevant screen classes. @Andrey Smirnov @Taylor Edgell based on our discussion on Friday 8 Oct 2021, the menu items are dynamically shown/hidden based on the user (admin/normal). This is achieved using "lambda" functions attached to the definition of each menu item. please see the start_screen.py for example

- 10 Oct 2021: One consideration for enhancement: Each screen has a collection of menu items and each menu item (currently) can be shown/hidden based on some condition. One thought is we can extend this concept to include additional access level permissions based on the logged in user. Fortunately our current software design allows for this to take place

- 10 Oct 2021: @Taylor Edgell we may have to reconsider the tokens in Authentication class. I put in the functionality to allocate new tokens and consume them when creating an account BUT when allocating to the __tokens array using “append” does not seem to work. For example I added a menu item “view list of tokens” so’s we can see the list growing and prove this part; but the array doesn't grow.It seems that class variables in python are assigned ONCE and thereafter their values never change.For this reason we may have to consider populating tokens in the DB

- 10 Oct 2021: Hi @Andrey Smirnov , thanks for fixing the cursor issue.The ORM mapping is in place. Please take a look at the map_to_user method inside user_repo.py. This works easily because the cursor had to be updated to output each result row into a dictionary. This is seen in the following code:

  with self.conn.cursor(buffered=True, dictionary=True) as cursor:

  Regarding construction of empty User objects. Yes, there is a scenario in which we definitely use an empty User instance. The scenario is: obtain an instance of the current logged on user. If nobody has logged on, then return an empty instance (as opposed to a None instance)

Python anomalies

- 11 Oct 2021: After some further investigation, there is some level of progress with regards to the import shenanigans based on trying to "python3 src/domain/auth/authentication.py". The following approach may work for other files too, not sure at the mo'
  1. Use of PYTHONPATH. This environment variable is set to the actual folders to be referenced

2. Moving imports to the BOTTOM of a class file. this is interesting because it delays the processing of files until everything ABOVE the imports has been processed.
3. Removing redundant import statement. Removing cyclic imports

API vs monolithic

- 18 Oct 2021: The reason to leverage Authentication from our monolithic approach is because the API exposes business functionality to a public audience using this chain of calls:

```
(API)
entries = SafetyStore().search_for_entries(search_terms)---
> (SafetyStore().search_for_entries)
        if
not Authorisation.test_logged_on_sys_access_level(Levels.SEARCH_ENTR
IES):
       --->  if not Authentication.user_logged_on():
return False
user = Authentication.logged_on_user()
```

What alternative would you propose if this approach is not suitable?

Code tests

- 24 Oct 2021: @Andrey Smirnov it's great work. Given that the readme document is probably expected to be minimal, I think there's some really good content in here, dressed to impress.

  Guys, for the tests, a few small changes had to be made

  1. Admin password is by default "Admin123!" (this matches our password policy)
  2. slight extra work on the "map_to_user" function in the "user_repo" class
  3. password regular expression moved into authentication class

the tests make use of the "unittest" module and they are configured to hide the "print" statements generated by our code during each test. Please do take a look see whether the tests are sufficient or you would like to more

Project Management

- 17 Aug 2021: Hi *@Andrey Smirnov @Taylor Edgell* please find attached a proposed project schedule for our team. The basic idea is
    1. Focus on Requirements for 3 weeks (unit 2, 3, 4)
    2. Focus on Report for 2 weeks (5, 6)
    3. Regular meetings every Friday at 17:00 BST to support each other for the first part of the assignment.
    4. Initialize discussions of code implementation from unit 5 onwards.

  Please do take a look at provide feedback for where we can do better or if you would prefer to propose alternate target dates.

- 17 Aug 2021: Reply to *@Taylor Edgell*: Thanks for the feedback. Yes, i am with you on the "template" statement because I am considering that it may be possible to work almost in parallel with taking your output and putting into the Technical Report as we go along. Andrey can also carry out QA on the content as it chugs along. there is hope that we may be able to consider coding much sooner than the end of unit 6. The part 2 requirement is that we deliver an API and a monolithic solution, that is both secure and testable; for this reason, I think keeping Part 2 in mind is key.Vital though is that security is baked into our design from the start

- 18 Aug 2021: Reply to *@Andrey Smirnov:* Yes agreed, the "feature creep" is a concern we must address in our meeting. Because it may look fantastic to build a real-world imitation in our designs for Part 1, but it is Part 2 that requires consideration because we have 3 deliverables for it:
    1. API / Microservices
    2. Monolithic
    3. Secure (including GDPR) and Authentication

- 20 Aug 2021: Morning guys, attached is the agenda form for today's meeting to give us some guidance on what we wish to achieve as outcome. The biggest requirement from the meeting is that we obtain challenges and requirements. All other points, such as assumptions, approaches, tools and design will slot in provided we understand the two points

- 23 Aug 2021: Hi guys, please find attached the agenda for tomorrow's meeting. (@Andrey Smirnov you did express earlier to meet without notes, but I think for our benefit, it is good to capture something each time we have a "formal" meeting. We can add this to our portfolio and show that we work well and collaborated well in reaching decisions)@Andrey Smirnov @Taylor Edgell however, if yous absolutely do not wish for meeting notes to be captured, that's fine; please do let me know.

- 30 Aug 2021: @Taylor Edgell we have a team discussion about secure programming languages. I think it would be useful to also include your opinions in the discussion. So.... is it ok if Andrey and I focus on implementation discussion today, and when we meet later this week then we pickup the team discussion where we can all be participants?

- 2 Sep 2021: Hi guys, please find attached a summary of points from our notes related to the question "What is a secure programming language". I provide the collated notes here, because I am unsure of the best approach we must take and so err on the side of caution and too-much is better than too-little. Please do feel free to ignore the artefact if you do not require it.

UML Diagrams Feedback

- 23 Aug 2021: @Taylor Edgell We can answer the question of "Do we require swimlanes", by challenging it with another: "How important is depiction of responsibility to the stakeholders? “From my perspective there is no harm in delivering BOTH styles, time-dependant. (1) a complete overview regardless of responsibility, and (2) one that is partitioned by responsibility. Though... it may be a duplication of effort, which we want to avoid. Our goal is to ensure designs meet the requirements of stakeholders, and clearly help them to understand the solution. As

you are acting in the role of an architect, I would recommend that yes, activity diagram with swimlanes are provided. Why? Doing so will aid consideration of the inputs and outputs of each component involved and how the data is managed across boundaries (or "domains").
And since we are concerned about the security of the system, swim lanes allow us to more clearly visualise the boundaries that require greater attention to security detail. @Andrey Smirnov What is your experience / input regarding use of swim lanes and stakeholders?

UML Activity Diagram Feedback

- 23 Aug 2021: @Taylor Edgell Activity Diagram Feedback

1. Please label each decision node with what is being decided on.
2. "Test Security Token" and "Test encrypted input against encrypted password" come across as too vague; we can strengthen what is being done by naming it for example "Compare token hash" and "Compare hashed input with encrypted password"
3. The flow for activity "Lock account for 1 hour" does not completely make sense in that it should rather terminate with a display message "Account has been locked. Please try again in 1 hour". This way, it is explicit that the flow must be re-run after the specified time period.
4. The assumption note is a nice touch! 🙂 (because we're making explicit any assumptions that must exist outside of this flow)
5. The activity "Allow access to application". Can we expand this to talk a bit more about what does it mean to allow access? We have referenced "tokens" earlier in the flow, so "Allow access" one assumes will then deal with "roles", "permissions" and updating the "token"

- 23 Aug 2021: @Taylor Edgell Activity Diagram Feedback: SafetyDB

1. Activities "Access application" and "log in" are both covered in the previous activity diagram. Recommend that we replace this with a single activity "Access application" and make a note to reference the previous designed activity flow.
2. Please label all decisions nodes with what is being decided on.

3. What does "Access safety DB" mean? Is this an activity that navigates to a search page or does it mean to establish a connection to the Safety DB (for example, "Access upload option")?
4. Access Upload Option:
    a. For consistency, can we please use "input xxx" whenever a user enters a value
    b. What does "Test document" mean? Are we checking the content, testing for viruses, performing automatic text detection? Perhaps we can title this activity a little clearer on what test are performed. It may be that we can add a few more activities here (or create a separate activity flow describing this activity in more detail)
    c. The activity "Restrict Account" is quite something 😊 One strike and they're out! What's the thinking behind this activity?
    d. The activity "Store document in DB" seems to conflict with "Upload document" because later in the flow there is a "Delete Upload" activity. From where are we deleting the upload?
5. Search DB:
    a. The activity "Match search term" makes sense, however, where does the determination of a "match" come from? Perhaps we obtain it from ElasticSearch, or a cloud service or the local DB. I wonder if we should rename this activity
    b. The activity "Request section access" looks like another activity flow chart in the making...
    c. The activity "Download document" looks like another activity flow chart in the making because there are multiple steps hidden by this activity. Especially from a security perspectve which we need to show.

- 23 Aug 2021: @Andrey Smirnov yes, I find that ArchiMate is a lovely higher-abstraction of modelling, designed not much for programmers, but more for management. It is a beautiful language and I confess, has influenced my UML thinking. The biggest beauty of ArchiMate is the idea of derived relationships. This is such a powerful concept because we can model at different levels of abstraction based on needs of the stakeholder.

- 23 Aug 2021: @Taylor Edgell Class Diagram V2 Feedback I believe the composition is the wrong way around. It is the Creator who owns/creates the reports. If we delete

a Standard Report or Safety Entry, we do not remove the Creator because the Creator must still be able to login and do stuff

- 23 Aug 2021: @Taylor Edgell Class Diagram V2 FeedbackThis class looks more like a "Permissions" table, rather than a "User Interface" class because it holds what clearly looks like permissions

- 23 Aug 2021: @Taylor Edgell Class Diagram V2 FeedbackSome actions on this class do not make sense on the data item. For instance "upload Document" (and one assumes "Download Document") don't seem to be intrinsic actions of an "entry". Why? Because we can argue that an "Entry" is just a piece of basic safety data, therefore the actions of download or upload are handled elsewhere

Model Reasoning

- 23 Aug 2021: @Taylor Edgell hmmm... yes, I see what you mean. Should we allow such a scenario to take place? For instance, if a Creator is removed (because they leave the project), we ought to keep the Reports anyhow. BUT if we remove the reports, then we remove all associated Creators. The consideration is from the perspective that a Creator is a user who will logon and do stuff. Therefore, one resolution is that we never delete Creators, merely mark their accounts as "inactive" From a modelling perspective, it seems odd to speak the statement " A standard report OWNS/CREATES a Creator" But it sounds more sensical to speak the statement "A Create OWNS/CREATES a Report"

- 23 Aug 2021: @Taylor Edgell Class Diagram V2 FeedbackThe name of this class does not match with the content of the fields "Contractor_xxxx". Perhaps you mean this class to represent a "Contractor"?