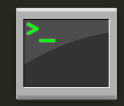
WP00

This Course

Design of Software Architectures

Dr. Vadim Zaytsev aka @grammarware, 7 September 2022

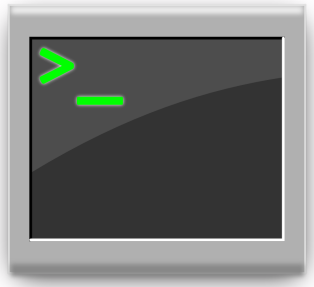


UNIVERSITY OF TWENTE.

Vadim Zaytsev

- Professor in Software Evolution (at FMT)
  - Ex-Analyst/Developer (Raincode)
  - Ex-Chief Science Officer (Raincode Labs)
  - Ex-UvA teacher (MSc SE)
  - Ex-CWI/VU/… researcher (DBLP)
- Programme Director (TCS & CS)
- Focus:
  - legacy software systems
  - software language engineering
  - software analytics
  - grammarware
- Follow **@grammarware**!
- https://discord.gg/n7VQAPNBPD?

http://grammarware.net



Design of Software Architectures

- Goals:
  - understand what software architecture is
  - understand when software architecture is needed
  - understand why software architecture is needed
  - have a conceptual framework for software architects
  - be able to explain the above to others
  - (know the rest)

the rest in https://osiris.utwente.nl/student/OnderwijsCatalogusZoekCursus.do

In DoSA, you will:

Simulate architecting a real system

Apply knowledge from sessions

Realise it is not exact science

Focus on understanding concepts and principles

Establish consistency throughout the document

Provide rationale/justification of choices

Apply the knowledge that you already have

Evaluate architecture of another group

Suggest improvements

Cases

TurboWorkbench

TurboWorkbench is the integrated development environment of the future. It supports any programming language known to humankind, as well as any target platform by plugging in existing compilers at its backend. It is configurable to be used by junior interns that are making their first humble steps in learning the art of software engineering, or by senior developers who have seen it all and are not afraid to perform large scale refactorings across the codebase with one keystroke, or even by the chief officers in the top management who are interested in project progress and product internal quality than in the code itself.

TurboWorkbench always exceeds user expectations. Not only it highlights your code, but also visualises it in the most appropriate manner, fitting the use case and the context of the situation at hand. It allows companies to develop and express their own coding standards, keeping an eye on the prevalence of undesired constructs in the code, as well as enforcing the use of certain language idioms that are considered good practice in the language of choice. Code completion works so well that some programmers reportedly create entire applications just by naming their new projects appropriately and agreeing repeatedly to suggestions given by **TurboWorkbench**. With one click of a button a developer can go into the state of flow and stay in the zone, avoiding distractions. When collaboration with other human developers becomes unavoidable, **TurboWorkbench** also offers comfortable computer-supported collaborative code editing experience with built in audiochat, integrated screen sharing and control delegation, fused in ticketing system, plugins for planning poker with suggestions based on task affinity to the developer profile, etc.

The code analysis engine of **TurboWorkbench** is second to none in the modern world. It quickly performs the most sophisticated cross-language analysis algorithms, routinely incorporates their results in the workflow and incentivises its users to act on reported signals, preventing creation and accumulation of technical debt. In fact, the analysis engine does not stop with code, it also oversees the entire history of the codebase, profiles past and present contributors to it, and adjusts itself according to the developer profiles, matching both the profile of the **TurboWorkbench** user and the profiles of the previous committers of the code being inspected or worked upon.

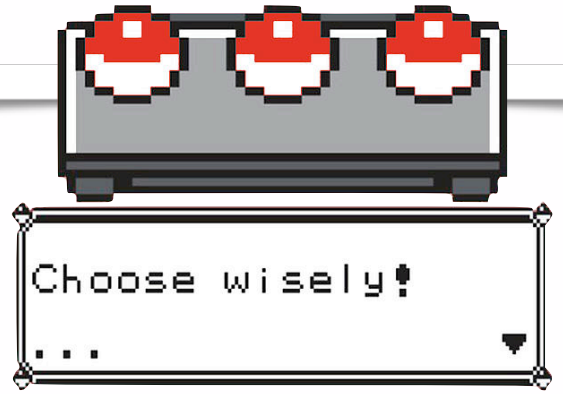
Wellerness

Wellerness aims for a sustainable future, where only the most convenient and green methods of shipment and transportation are chosen and their capacity is shared as much as possible. It helps to connect boat and ship owners with those that would like to use them for travelling or transporting goods.

On one side, owners of ships, barges, boats and ferries can register at **Wellerness** and propose their potential reach (i.e., places they would be willing to sail to, if the demand is high enough), as well as preliminary itineraries (i.e., the paths they are intending to follow more or less exactly following the time schema). On the other side, individual customers without strict time constraints can register at **Wellerness** as well and request products shipped from elusive locations or submit flexible transportation requests. For example, a ship sailing from the Caribbean towards Europe with a stop at Barbados can pick up an exotic box of sugar and tea and rum that someone from the Netherlands would like to eventually have. Alternatively, someone could let it be known to the system that a group of 5 people intends to visit fjords at some point during the summer, which is a request equally well matched with a ship sailing to Sweden in June or to Norway in July.

All goods are safe with **Wellerness** since it is certified to participate in legitimate international trade. Goods that are awaiting shipment, are collected at **Wellerness**'s warehouses and stored there safely and compactly, for senders' convenience and for simplified loading. The loading itself is done by drones, so that ships do not have to alter their course and pay harbour fees.

Sophisticated algorithms make sure all customers find what they need, and assume some degree of flexibility by proposing adaptations and modifications to the plans of both involved sides of customers. This adaptability is unique on the market, and gives **Wellerness** a decisive competitive advantage against their business rivals.



COBOL-FIT

COBOL-FIT (short for *COaching and BOosting the Lifestyle of FITness*) is the gym of the future. Instead of offering intrusive yearly subscriptions that merely provide access to the physical location filled with unfamiliar equipment as all competitors still do, **COBOL-FIT** seeks to arrange an integrated fully immersive experience, personalised for the needs and peculiarities of each user. The gym space, the equipment installed there, the built-in entertainment systems, the supporting mobile application, all form a unified purposeful whole.

Some customers come to **COBOL-FIT** because they want the best advice on strategies for reaching well-defined objectives such as growing muscle power or losing weight. For them, the ultimate quantified self experience is designed that combines data from self-tracking technology with theories known from dieting and sport science, and comprises automatic adjustments of weights in exercise machines and providing detailed instructions on the number of reps and sets, with cameras registering the movements and advising on improving the proper technique for each exercise. Some other customers seek motivation, inspiration and encouragement, and for them **COBOL-FIT** offers a range of gamification options such as incremental rewards, shareable achievements, collaborative challenges, meaningful narratives, personal branding, scientific visualisation, purposeful training trajectories, matching training buddies and augmented reality. **COBOL-FIT** always supports its customers in their own unique style of training, which it infers from their behaviour and facilitates in the most subtle and complementary way.

The proposed set of exercises is always based on the broader context, taking into account the social aspect for those training with friends or family; nutrition, since the food consumed up to several hours prior to the gym visit, is known to affect the physiology of training; gym visits in the recent past and the muscle groups targeted by those; the learnt circadian rhythm of the one who trains; recent casual activities of the same day which might have included carrying heavy objects or actively using stairs; etc.

COBOL-FIT integrates itself into the lifestyle of its customers, it even monitors (and proposes) their visits to physicians, therapists, masseurs and hospitals.

Proceed in Work Packages

- Choose a case, form a group
- Each group writes an architecture description doc
- Sessions are interactive
- Each ends in homework
  - adds a chapter to the architecture description
  - adds a cross-cutting concern
  - refines an existing piece of the architecture
- Can use us as a project coach and/or stakeholder

https://canvas.utwente.nl/courses/11270

Group Assignment: SA Description

- System Scope
- Stakeholders
- Usable Concerns
- Involvement
- Communication
- Specification Overview
- Related Systems
- Trends
- Developments
- Related Processes
- Dominant Decomposition
- Design Patterns
- Families
- Commonality
- Variability
- Quality Attributes
- Creativity
- Effort

Tips for the Group Assignment

- Start with the case description
  - be explicit about changes/additions
- Distribute responsibilities
  - document it in an appendix
- Make sure to apply each WP
  - start by writing a new chapter
- Don’t get stuck
  - make assumptions

Individual Assignment

- Evaluate the architecture of the counterpart team
- Tasks like
  - Select 2 questions that should have a better answer
  - Select 1 question that has a satisfying answer
  - Apply one QA scenario from your architecture
- Checks if you can
  - handle architecture documents
  - suggest improvements constructively

Deadlines

- Intermediate feedback
  - 19 September 18:00 (~WP07)
  - 12 October 12:00 (~WP15)
- Final document
  - 28 October 17:00 (~WP23)
- Individual
  - 11 November 17:00

week number	36	37	38	39	40	41	42	43	44	45
week type	M\L	M\L	M\L	M\L	M\L	M\L	M\L	M\L	M\E	MAE
quartile-week	1-01	1-02	1-03	1-04	1-05	1-06	1-07	1-08	1-09	1-10
Monday	5	12	D 19	26	3	10	17	24	31	7
Tuesday	6	13	S 20	27	4	D 11	18	25	1	8
Wednesday	S 7	S 14	21	S 28	5	S 12	S 19	26	2	9
Thursday	8	15	22	29	6	S 13	20	27	3	10
Friday	S 9	S 16	S 23	S 30	7	Information Day14	S 21	D 28	4	D 11
Saturday	10-09	17-09	24-09	01-10	08-10	15-10	22-10	29-10	05-11	12-11
Sunday	11-09	18-09	25-09	02-10	09-10	16-10	23-10	30-10	06-11	13-11