**Thinley Rabgay’s Reviewed Articles**

**Results and analysis of results**

**RQ1: What are the different teaching strategies used in software architecture education?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Explanation** | **Reference** |
| **Agile Methodologies (Scrum)** | Scrum with three phases were implemented.  Initial phase: general objectives of the assignments were outlined.  Development phase: Sprints (3 sprints) producing one product based on assignment after series of cycles.  Closure phase: When all requirements are met, the final product is ready for review. | P1 |
| **Case-based Learning** | Uses guided inquiry method and provides more structure during small group sessions.  Implementation of case-based learning the form of bullet-cases, mini cases and descriptive cases. | P2, P12 |
| **Problem-based Learning** | Uses more unguided approach. | P2 |
| **Learning by Doing** | Used to teach the students how to be able to write, visualise, and turn the 2D shapes into 3D units manually.  Learners perform one or a few architectural activities based on knowledge acquired before or during the actual experience. | P3, P4 |
| **Experiential Learning** | Used to fabricate self-structured two-meter-high pavilions using 4mm cardboard where they select the rules and different designs.  The trial-and-error processes were involved and documented all.  Experiential learning strategy with four stages: Concrete Experience, Reflective Observation, Abstract Conceptualisation and Active Experimentation. | P3, P7 |
| **Lectures/Traditional teaching** | Guest Lectures to illustrate the relevancy of topic covered outside the open-source systems as well.  Instructor explained the concept at the beginning of laboratory session using slides, web pages, and code extracts from 15 to 55 minutes. | P11, P13 |
| **Project-based Learning** | Mimic the software development in industry working in groups promoting teamwork.  Students learn to put practice of industrial context complexity trough medium complexity project in a team.  Involves complex challenges and problem-solving skills with real-world case studies which promotes practical skills. | P4, P5, P9 |
| **Collaborative Learning** | Students contribute to achieve a particular goal.  Cross-collaboration through three models: Collaborative Studio Model, Trans disciplinary Encounter Model, Real team Collaboration where students work on their own and then working in team.  Collaborative Learning through open-source projects such as Git and GitHub in groups since the members have different levels of experiences. | P4, P6, P11 |
| **Preparing Teaching Manuals** | To overcome the multiple teaching methodologies, detailed teaching manuals consisting of learning objectives, outcomes and teaching methodologies for every topic included in the curriculum. | P5 |
| **Game-based Learning (Kahoot! DecidArch)** | The games support the learning process of specific topics, and it involved the card style, requiring face-to-face interaction.  The Kahoot! was used as interactive test to check the students’ understanding of the concepts from videos at the beginning of laboratory session during flipped classroom methodology.  The DecidArch as used for teaching a particular topic where students play the game in a team of architects and bringing various skills and competencies together. They will play the game at midpoint where the learners attend lecturers and working on their practical project parallel. | P4, P13, P14 |
| **Flipped Classroom** | Instructor recorded the videos in their own office not more than 6 minutes and uploaded to a YouTube playlist which was created as an unlisted. | P13 |
| **Team-based Learning** | Focuses on teamwork and peer-evaluation through group activities. | P9 |
| **Online Teaching** | Synchronous teaching using Zoom, MS Teams, and WhatsApp.  Asynchronous methods included recorded lectures and supplementary learning via platforms like YouTube, NPTEL, SWAYAM, EdX, and Coursera. | P8 |
| **Architectural Kata Structure** | It is a novel, and exciting learning activities with several exercises.  As a learning activity for students to practice the skills needed to complete their assignments where teacher act as a moderator.  The workshop had two phases: Discussion and Design Phase where students explore the kata problem and design the solution. In Peer Review and Voting Phase, they present a vision of their solution and other students evaluate and vote the solution. | P5, P10 |

**RQ1a:** **Which teaching strategy aligns with industry expectations and requirements?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Industry Alignment** | **Reference** |
| **Agile Methodologies (Scrum)** | It used Agile Methodologies which are highly practiced in current software industries.  It uses iterative sprints to simulate real-world development cycles, making students industry ready (learning experience more authentic). | P1 |
| **Case-based Learning** | It aligns with industry expectations and requirements since they worked on the real-world case studies and use of frameworks like the Rational Unified process which are required for industry standards.  CBL is effective for learners with less working experience, and it can be foundation for the first-time office goers. | P2, P12 |
| **Problem-based Learning** | It aligns with industry expectations and requirements since they worked on the real-world case studies and use of frameworks like the Rational Unified process which are required for industry standards. | P2 |
| **Learning by Doing** | Learning by Doing provides hands-on-fabrication process mirrors design-build methodologies in practice. | P3 |
| **Experiential Learning** | Experiential Learning encourages analytical thinking, material experimentation, and manual simulation of algorithms, which are valuable in computational and parametric design fields.  Simulating real-world software risks and emphasising architectural decision-making**.** | P3, P7 |
| **Lectures/Traditional teaching** |  |  |
| **Project-based Learning** | Project-based learning promoting teamwork activities closely mimic real-world expectations, aligning well with industry needs.  The rationale behind this strategy is to provide an environment as close to an organisation as possible.  As they enhance teamwork and ability to handle the real-world problems through experiment on a realistic job environment. | P4, P5, P9 |
| **Collaborative Learning** | The use of interdisciplinary and multidisciplinary collaborative learning, mirroring real-world practices, working as a team.  Collaborative learning approach since they work though open-source projects which deals with real industry workflows (by using GitHub). | P6, P11 |
| **Preparing Teaching Manuals** | By developing online repository, students developing their skills. | P5 |
| **Game-based Learning (Kahoot! DecidArch)** | The game reflects real-world architecture decision processes by simulating design process and conflicting stakeholder requirements, closely working in a team with given time bound. | P14 |
| **Flipped Classroom** | Flipping labs helps align with industry practices by emphasizing hands-on work, collaborative problem-solving, and independent preparation, echoing professional self-paced learning with support. | P13 |
| **Team-based Learning** | As they enhance teamwork and ability to handle the real-world problems through experiment on a realistic job environment. | P9 |
| **Online Teaching** |  |  |
| **Architectural Kata Structure** | The students put design’s theoretical knowledge into practice through real, practical and exciting challenges.  It mimicked the real-world architectural problem-solving and encourages decision-making and trade-off analysis. | P5, P10 |

**RQ2: How are students provided hands-on practical experiences related to software architecture teaching?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Hands-on Experiences** | **Reference** |
| **Agile Methodologies (Scrum)** | The course employs hands-on projects, GitHub repositories for version control, Trello for task tracking, and a final project requiring students to develop a blockchain-based NFT marketplace.  Students worked with Docker, Kubernetes, API design, CI/CD pipelines, and cloud deployments. | P1 |
| **Case-based Learning** | Introduced software tools and created architectural diagrams and later made optional due to the steep learning curve.  Encouraged peer discussions to understand architectural trade-offs.  Students gain hands-on practical experiences through case discussions, debates, and public hearings, where they apply theoretical knowledge to solve real-world problems. | P2, P12 |
| **Problem-based Learning** | Introduced software tools and created architectural diagrams and later made optional due to the steep learning curve.  Encouraged peer discussions to understand architectural trade-offs. | P2 |
| **Learning by Doing** | Students fabricated four self-structured pavilions using cardboard without digital software.  Exercises involving software architecture design and evaluation. | P3, P4 |
| **Experiential Learning** | Hands-on activities included unit transformations, connection types, assembly processes, and structural evaluations.  Involved in realistic hands-on activities, simulating architectural risks related to availability, security, and performance.  Design for software qualities during session 2. | P3, P7 |
| **Lectures/Traditional teaching** |  |  |
| **Project-based Learning** | Students work on real-world projects, sometimes in collaboration with companies.  Students worked on medium-complexity projects, simulating real organisational environments.  Students work on a project by applying ATAM (Architecture Tradeoff Analysis Method) to evaluate a real-world case. | P4, P5, P9 |
| **Collaborative Learning** | Collaboration involving different stakeholders to enhance problem-solving skills.  Through the architectural design studio and project-based group work architectural design processes, experiencing practical collaboration.  Students apply theoretical concepts to real, by selecting open-source projects hosted on GitHub.  Contributes to open-source and collaboratively documenting the works where they meet real software architects and learn from them. | P4, P6, P11 |
| **Preparing Teaching Manuals** | Building up the online repository of projects, design books, industrial reports, and case studies. | P5 |
| **Game-based Learning (Kahoot! DecidArch)** | Used for to support the learning process of specific topics (ATAM games).  Labs focused on solving exercises collaboratively after preparing with videos and quizzes (via Kahoot).  Students work on industry related projects and apply game-based decision-making learning approach where they got to experience the process of decision making and working in team. | P4, P13, P14 |
| **Flipped Classroom** | Students developed mashup web applications using real APIs, deployed them on the cloud, and followed architectural styles like MVC. | P13 |
| **Team-based Learning** | They are grouped by individually selecting a role, and activities are performed in groups, discussing their proposals through peer-review and brainstorming. | P9 |
| **Online Teaching** |  |  |
| **Architectural Kata Structure** | The students presented industry-acceptable proposals by applying architectural patterns, quality attributes, and design patterns where they apply their skills in a simulated environment, evaluated by industry professionals.  They were provided through analysing of realistic kata problems, designing architectural solutions and presenting the solutions and peer-review feedback. | P5, P10 |

**RQ3: What are the various assessment methods used to assess their students?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Assessment Methods** | **Reference** |
| **Agile Methodologies (Scrum)** | Pre- and post-course assessments, conceptual quizzes, self-evaluation surveys, project-based grading (Assignments 30%, Midterm 30%, Final Project 40%). | P1 |
| **Case-based Learning** | Competency-based assessment aligned with an accredited skills framework.  Peer reviews and discussions as formative assessment tools.  Assessments are conducted using public hearing formats, discussions, and debates. | P2, P12 |
| **Problem-based Learning** | Competency-based assessment aligned with an accredited skills framework.  Peer reviews and discussions as formative assessment tools. | P2 |
| **Learning by Doing** | Peer reviews and feedback for collaborative learning and improvements.  Trial-and-error testing to evaluate students' understanding of design rules, form stability, and material behaviour.  Final assessment based on structural stability and creativity. | P3 |
| **Experiential Learning** | Peer reviews and feedback for collaborative learning and improvements.  Trial-and-error testing to evaluate students' understanding of design rules, form stability, and material behaviour.  Final assessment based on structural stability and creativity.  Assessment based on participation, written examination, in-class quizzes and exercises, and presentation of risk findings for all software qualities. | P3, P7 |
| **Lectures/Traditional teaching** | Individual grades also depend on participation, activity journals, and engagement during lectures.  Assessment is performed through quizzes (using Kahoot!), practical, and final exams. | P11, P13 |
| **Project-based Learning** | Architectural Kata Workshops where software architects evaluate their solutions and focus group discussion was set up have discussion.  Assessment methods used was by evaluating team assignments and brainstorming outputs based on technical aspects (quality of solutions, comprehension of concepts) and interpersonal skills (time to present their decisions, individual participation, peer-review contributions, contribution to peer-reviews, their willingness to accept peers’ requests). | P5, P9 |
| **Collaborative Learning** | Assignments graded using predefined rubrics to apply the theory to practice.  Book chapters evaluated for content, depth, writing, and originality.  Peer review process using EasyChair to simulate academic review practices | P11 |
| **Preparing Teaching Manuals** | Architectural Kata Workshops where software architects evaluate their solutions and focus group discussion was set up have discussion. | P5 |
| **Game-based Learning (Kahoot! DecidArch)** | Assessment is performed through quizzes (using Kahoot!), practical, and final exams.  There is no definite assessment methods used but however they were provided with Decision Taking Templates which would have helped them to log and reflect reasoning for their decisions. | P13, P14 |
| **Flipped Classroom** | Assessment is performed through quizzes (using Kahoot!), practical, and final exams. | P13 |
| **Team-based Learning** | Assessment methods used was by evaluating team assignments and brainstorming outputs based on technical aspects (quality of solutions, comprehension of concepts) and interpersonal skills (time to present their decisions, individual participation, peer-review contributions, contribution to peer-reviews, their willingness to accept peers’ requests). | P9 |
| **Online Teaching** | Continuous assessments through quizzes, assignments, and presentations were conducted via online platforms. | P8 |
| **Architectural Kata Structure** | Architectural Kata Workshops where software architects evaluate their solutions and focus group discussion was set up have discussion.  Problem-based group projects (three assignments to design, document and evaluate architectural solutions for a software project case).  Individual final written examination. | P5, P10 |

**RQ4: What are the outcomes of software architecture teaching strategies?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Outcomes** | **Reference** |
| **Agile Methodologies (Scrum)** | Significant improvement in knowledge and confidence levels (e.g., confidence in microservices implementation increased to 4.5/5).  Students developed industry-relevant skills in cloud-native, API-first, and microservices-based applications. | S48 |
| **Case-based Learning** | Discussions and team exercise enable the leaners to learn from peer and apply what they learn.  79% of feedback comments were positive and students wanted more practical exposure and tool-based learning.  Effectively bridges theoretical knowledge with real-world practice, enhancing analytical, technical, and communication skills. | S49, S59 |
| **Problem-based Learning** | Discussions and team exercise enable the leaners to learn from peer and apply what they learn.  79% of feedback comments were positive and students wanted more practical exposure and tool-based learning. | S49 |
| **Learning by Doing** | Students developed deeper material knowledge by working hands-on with shape grammars.  Increased confidence in problem-solving and structural evaluation.  Enhance students’ ability to apply concepts. | S50, S51 |
| **Experiential Learning** | Improved spatial reasoning skills through form manipulation and assembly exercises.  Better preparedness for computational design (CD) methodologies.  Ability to learn from others and compare different solutions to solve a concrete problem.  Improved student understanding and positive sentiment. | S50, S54 |
| **Lectures/Traditional teaching** |  |  |
| **Project-based Learning** | Enhance students’ ability to apply concepts.  Students gained proficiency in SOLID principles, architectural patterns, quality attributes, and decision-making.  The PBL/TBL approach improves both technical and interpersonal skills including technical language and vocabulary.    And it improves the course attendance, peer learning without intervention of instructor and active participations. | S51, S52, S56 |
| **Collaborative Learning** | Promote teamwork skills.  Generated group work spirit.  Learnt to understand inter-cultural differences.  Improved communication and presentation styles.  Exposure to different architectural systems and methods.  Growth in leadership and team management.  Students successfully contribute to open-source projects and improved understanding of architectural concepts through practical experience.  Developed technical and social skills required for software architects by putting theory to practice.  Enhanced collaboration skills through group work and peer feedback. | S51, S53, S58 |
| **Preparing Teaching Manuals** | Students gained proficiency in SOLID principles, architectural patterns, quality attributes, and decision-making. | S52 |
| **Game-based Learning (Kahoot! DecidArch)** | Enhance students’ ability to apply concepts.  Enhanced reasoning skills (LO1).  Greater appreciation for diverse solutions (LO2).  Limited but notable awareness of architectural decision dynamics (LO3). | S53, S61 |
| **Flipped Classroom** | The flipped-classroom approach shows positive outcomes with higher student satisfaction. | S60 |
| **Team-based Learning** | The PBL/TBL approach improves both technical and interpersonal skills including technical language and vocabulary.    And it improves the course attendance, peer learning without intervention of instructor and active participations. | S56 |
| **Online Teaching** | Online teaching proved effective for theoretical subjects.  Practical-based learning, especially design studios, struggled to achieve learning objectives.  Increased digital adoption, but dissatisfaction with the current state of digital tools for architecture education. | S55 |
| **Architectural Kata Structure** | Increases students’ engagement during learning and assessment.  Students learnt to make decisions and do trade-offs in situations where business needs and quality requirements complete.  They also learnt about prioritising non-functional requirements. | S52, S57 |

**RQ5: What challenges do students and educators face with the software architecture teaching approach?**

|  |  |  |
| --- | --- | --- |
| **Strategy** | **Challenges** | **Reference** |
| **Agile Methodologies (Scrum)** |  |  |
| **Case-based Learning** | Covering of extensive content within limited duration.  Lack of sufficient working experiences may influence the understanding of the case studies. | P2 |
| **Problem-based Learning** | Covering of extensive content within limited duration.  Lack of sufficient working experiences may influence the understanding of the case studies. | P2 |
| **Learning by Doing** | Students faced difficulties in scaling up designs from models to full-sized structures.  Time constraints in the course made it challenging to explore more fabrication techniques. | P3 |
| **Experiential Learning** | Understanding material properties required trial-and-error learning.  Time constraints in the course made it challenging to explore more fabrication techniques.  Time constraints, depth of content is vast, and the need for instructors with real-world experience. | P3, P7 |
| **Lectures/Traditional teaching** |  |  |
| **Project-based Learning** |  | P9 |
| **Collaborative Learning** | Language barriers since English is not their mother language.  Cultural differences among the students lead to miscommunication.  Different skill levels in software and design approaches.  Difficulty in selecting appropriate open-source projects. | P6, P11 |
| **Preparing Teaching Manuals** |  |  |
| **Game-based Learning (Kahoot! DecidArch)** | Limited time reduced the impact of “event” dynamics, weakening LO3. | P14 |
| **Flipped Classroom** | Challenges include initial effort in creating videos, technical limitations of video editing software, and ensuring consistent quality and structure of videos. | P13 |
| **Team-based Learning** |  | P9 |
| **Online Teaching** | Lack of adequate infrastructure and technological support.  Digital divide and accessibility issues.  Limited effectiveness of online tools for hands-on learning and interaction.  Lack of proper frameworks for conducting design studios effectively. | P8 |
| **Architectural Kata Structure** | Time constraints and knowledge gap. | P10 |