**Objectives**

1. Learning Java / Scala / C# / Kotlin / Swift programming language on your own.
2. Learn about Low Code - No Code (LCNC) platforms as a team.
3. Do pre-project activities and requirements engineering for a software product.

**Instructions**

1. **Your learning is your absolute responsibility.** We are here only to help **you** to learn.
2. The purpose of this assignment is to **learn, learn and learn** and then gain marks**.** If you focus on the second and forget the first, your time and life are wasted.
3. Students will work in **groups of 3 from the same section**.
4. All work submitted must be integrated and fully understood by **all team.**
5. Resources used must be (1) listed at the end of report & (2) cited in report when used.
6. Cheating or copying solutions from any source or AI is not allowed and will be punished.

**Task 1 - Learning Java / Scala / C# / Kotlin / Swift (Individual - 2 marks)**

In this task, learn either Java, Scala, Swift, Kotlin or C# **on your own**. You can study Java from tutorials or playlists like Desouki and Neseem. **You need to learn enough to make non-trivial programs.** Choose any language, but (1) Lecture examples and exams are in Java (2) **all team members should agree on the same language. Each team member will deliver a small program of choice in this language written by hand NOT AI.** Student will submit in Part1 Report:

1. They will specify the language they chose.
2. A table with team members, hours spent in learning and sources used for learning
3. They will add 6-10 screen shots and a link to a short 3 min video **for each member** showing the app he did in this language. They will have **three apps, one for each student.**
4. The **main** function of the app that shows the main logic as text in the report.
5. TA will ask any team member any question about any of the programs they made.

**Task 2 - Learning LCNC Platforms (Team - 2 marks)**

Team will study 2 LCNC tools and try them out. See **<https://quixy.com/blog/top-10-predictions-for-no-code-low-code/>** Write a report about (1) Evaluation of these tools and their potential and what they can do and what there benefits are and the quality of the systems they produce and if they will take the job of developers. (2) A comparison of the two tools against each other and what features are offered in each, (3) A sample app with one of the tools. Submit in Part1 Report:

1. An explanation of LCNC platforms and answer to all the question in item (1) above
2. A comparison in text and in a table between the two tools you tried.
3. Screens shots from the app you did with the tool and a link to 3 min video explaining the app.

**Task 3: Pre-Project Activities (Group - 4 marks)**

In task 3, add the IDs of all team members, and if odd, you will do project 1 and if even, you will do project 2. In this phase, you will teach yourself about these pre-project activities and implement them and write them in at least 5-10 pages single space font size 12 in the report of Parts I & II.

1. **Market and Gap Analysis**

* **Global Practices:** Research how other countries or foreign products handle the same problem and describe the features supported in these products **in your own words.**
* **Egypt's Market**: Look at local and Arab apps / services / platforms that do the same thing.
* **Gap Analysis:** Identify existing gaps in the current system or market in the Arab region to ensure your software adds value.

1. **Market Segmentation & Research**

* **Customer Segments**: Identify the primary customers impacted (e.g., parents or individuals managing household expenses, independent investors, young adults (20 - 30 yrs), etc.).
* **Demographics**: Age, income, education level, and tech usage patterns.
* Explain why these groups would be interested in your solution.

1. **Domain analysis.** You will perform domain analysis to understand the domain, the terminology used in the domain. See domain analysis example / structure in lectures.
2. **The Proposed Solution :**

* ****Purpose and Goals:** What the software will achieve (e.g., manage personal finances, track investments, automate budgeting). And **high-level objectives.****
* ****Key Features and Functionality:** A brief overview of the core features and functional requirements.**
* **Target users:** Who the software is intended for.
* **Technologies:** The stack of technology to use, any third-party APIs or platforms to be integrated with the system (banking systems, payment gateways, etc.), major components (e.g., backend servers, databases, user interface), technical decisions (e.g., local app, cloud-based platform, web or mobile or both, etc.)

**Task 4: Project Phase 1 - Requirements Engineering (5 marks)**

1. Collect as much information as you can on your project topic (Budget management or investment management) from (1) similar products, (2) talking to experts, (3) Reading, and (4) AI tools (use this as the last source- ِAI is useful but is not analyst or engineer)
2. Prepare a list of questions that you need answers too. **All the class makes one list for each project. Someone needs to lead this work. Then we try to find experts to answer them.**
3. **Read carefully Software Requirements Specifications** template given. Read the instructions in red very carefully to know how fill the form. **Then delete the red parts.**
4. There is also a sample SRS for a previous project that can give guidance on how it looks like.
5. Write the SRS document of your project, including all the required details and use case model and enriched user stories. **Put at least 12 detailed enriched user stories**
6. Make a presentation for your TA about all parts of the project. Tasks 1 to 4.
7. Upload the **SRS by 16 March 2023 as a pdf file,** including all items and a **cover page with team names, IDs, emails, course name and number and document topic and title.** File name must be **CS251-2025-SectionNum-TA-ID1-ID2-ID3-ProjectName-SRSv1.0.pdf**

**Project Ideas**

**Project One**

### ****Personal Budgeting Software: Brief Description****

See: [How to Budget: The basics of budgeting | Intuit](https://www.intuit.com/blog/budgeting/the-basics-of-budgeting/" \l ":~:text=Learning how to be on a budget 1,... 7 7. Track your financial goals ) and [Beginner's Guide To Budgeting | Step By Step Tutorial! 2025 Setup For You!](https://www.youtube.com/watch?v=M7ucIl9MGs0&t=1s)

Personal budgeting software is designed to help individuals manage their personal finances by tracking income, expenses, and savings. These tools aim to provide users with a clear picture of their financial situation, set up and maintain budgets, and make smarter financial decisions. Key features may include:

* **Income and Expense Tracking:** Users can log sources of income and various expenses (e.g., utilities, groceries, gasoline, education, installments, entertainment).
* **Budget Creation:** Tools to set spending limits for different categories and track progress.
* **Financial Goal Setting:** Helping users save (and track savings) for specific goals (e.g., buying a home or car, going for Umrah or summer vacation, paying off debt).
* **Reporting and Insights:** Visual reports (graphs, charts) showing spending trends & goal progress.
* **Integration with Bank Accounts:** Many apps sync with bank accounts or e-wallets or credit cards if possible to automate transaction tracking.
* **Sample apps:** Mint, YNAB (You Need A Budget), PocketGuard, EveryDollar and GoodBudget. Some apps are designed for detailed, hands-on users (like YNAB), while others aim for simplicity and automation (like Mint and PocketGuard), depending on individual preferences.

**Project Two**

### ****Personal Investment Management Software: Brief Overview****

See lists like [الرجل الذى يحلم بالثراء #شخبطة\_من\_كتاب](https://www.youtube.com/watch?v=N5Lgv59Slj8&list=PLOVuuXitJxISAF8d0-sdwLD4J3iljxpwT) and [5 خطوات هتضمن لك بناء ثروة من الصفر .. مهما كانت ظروفك #شخبطة](https://www.youtube.com/watch?v=R4gSwTtBano)

Personal investment management apps help individuals track and manage their investments (stocks, real estate, bank accounts, land, crypto, etc.). They offer portfolio tracking, asset allocation, performance analysis, and risk assessments, helping users make informed financial decisions and manage their wealth.

### ****Key Features:****

* **Portfolio Tracking:** Monitors different assets like stocks, real estate, and savings.
* **Income Tracking:** Track the income streams generated from different investments.
* **Computation of Net-Worth and Rate of Investment (ROI).**
* **Visualization & Graphs:** Displays investment performance through charts and graphs for easy analysis and comparison with historical data.
* **Target Setting:** Allows users to set income goals, wealth targets, and retirement objectives.
* **Risk & Asset Allocation:** Analyzes risk & balances investments across asset types to diversify.
* **Integration:** Syncs with bank accounts, stock prices, brokerage accounts, etc.

### ****Top Personal Investment Apps:****

1. **Personal Capital:** Tracks investments and retirement accounts with performance insights.
2. **Mint:** Combines budgeting and investment tracking with goal-setting features.
3. **Wealthfront:** Automates investment management, optimizes taxes & retirement planning.
4. **RealtyMogul:** Tracks commercial real estate investments.
5. **Kubera:** Tracks diverse assets including bank accounts, stocks, and real estate.

**Useful Resources**

**Software Requirements**

How to write requirements <https://www.youtube.com/watch?v=PtJmjPkrSUE>

How to write SRS document <https://jelvix.com/blog/software-requirements-specification>

Functional vs non-functional requirements <https://www.youtube.com/watch?v=3fgfUHKITts>

Functional vs non-fun requirements <https://jelvix.com/blog/functional-vs-nonfunctional-requirements/>

Excellent explanation of non-fun req <https://www.altexsoft.com/blog/non-functional-requirements/>

Requirements tutorial <https://www.youtube.com/watch?v=XxxJZ_oduqo>

**Class diagram building examples**

<https://www.cs.unb.ca/~wdu/cs2013f05/a4ans.htm>

<https://www.youtube.com/watch?v=wsKdiPD7W8U&list=PLS4AbOQwMQKuVO4CufKPKQdLJdgZs7k5w>

**Online modeling tools**

<https://cruise.umple.org/umpleonline/> (Draw class diagram by coding in Umple and it is auto generated)

[https://online.visual-paradigm.com/drive/#infoart:proj=0&dashboard](https://online.visual-paradigm.com/drive/" \l "infoart:proj=0&dashboard)

https://creately.com/

**Standalone modeling tools**

<https://www.visual-paradigm.com/download/community.jsp>

<https://argouml.en.softonic.com/>

**What to Deliver**

Team will provide one integrated pdf that is well-formatted that includes these items.

**Delivery date: Draft of Tasks 1 & 2 28 Feb midnight**

1. A **well organized report** 12-20 pages with single space, font times new roman size 12.
2. Cover page, title, date, faculty name, course name, team names and emails, **IDs, section**, **phone of team leader**. Filename **CS251-A1-Part1-2-Section-ID1-ID2-ID3.pdf**
3. A table of contents and page numbers.
4. A table showing work breakdown and who did what in Parts 1 and 2.
5. Task 1:

* A table with team members, the language they learned, hours spent in learning and sources used for learning.
* 6-10 screen shots and a link to a short 3 min video **for each member** showing the app he did in this language.
* Code of the **main** function of the app that shows the main logic as text.

1. Task 2:

* An explanation of LCNC platforms and answer to all the question under Task 2.
* A comparison in text and in a table between the two tools you tried.
* Screens shots from the app you did with the tool and a link to 3 min video explaining the app.

1. List of references you used in the report.
2. **Delivery date: Final Tasks1 & 2 & 3 7 Mar midnight**

* Tasks 1 & 2 report
* Task 3: All the items mentioned under Task 3.
* Do pre-project activities and implement them and write them in at least 5-10 pages single space font size 12.

1. **Delivery date: Final Tasks 1 & 2 & 3 & 4 16 Mar midnight**

* One team member loads for the team in google classroom
* Each report should have a header with logo of FCAI, name of course and assignment, names of team, IDs, emails, title of the document, phone of team leader, date, etc.
* Submit one zip file that has the following 4 pdf files: (yes submit them again after any necessary changes u need to make for them)
  + CS251-2025-SectionNum-TA-ID1-ID2-ID3-ProjectName-Individual-LCNC-v2.0.pdf
  + CS251-2025-SectionNum-TA-ID1-ID2-ID3-ProjectName-PreProject-v2.0.pdf
  + CS251-2025-SectionNum-TA-ID1-ID2-ID3-ProjectName-SRS-v1.0.pdf
  + CS251-2025-SectionNum-TA-ID1-ID2-ID3-ProjectName-Presentation-v1.0.pdf
* Include the individual projects and LCNC of the current team members.
* Include slides of the 10-15 min presentation about your project, all tasks.

**Marking Criteria**

1. **2 marks**  Developing own prog. in a new language & evidences/screenshots/video

TA will ask you to demo the apps you did.

1. **1 mark x 2** Study of 2 LCNC platforms and discussing and comparing them

TA will ask you to demo the apps you did.

1. **1 mark x 4** For well done detailed descriptions for each of these items:

**Market and Gap Analysis**

**Market Segmentation & Research**

**Domain Analysis**

**Outline of the Proposed Solution**

1. **5 marks Software Requirements Specifications**

**1.5** marks For excellent and complete SRS, using the given template

**1** marks For excellent and complete use case model

**2.5** marks For complete and detailed enriched user stories

**-0.5** 0.25 for not naming the file properly

1. **-1** or **-2 marks** For a disorganized poorly formatted reports or wrong file names
2. **Possible 0** for any student not contributing with the team
3. **Possible –ve mark up to – 13** if any part is copied from net or books or other sources or AI-assistants are used when they should not be used.

**Policy Regarding Plagiarism:**

1. تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة
2. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم و لكن تبادل الحلول غير مقبول و يعتبر غشا.
3. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
4. قد توجد على النت حلول مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
5. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
6. فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.