CHAPTER I

INTRODUCTION

1.1 The Background

Language learning has become an essential skill for communication and career advancement in the modern world. The strategy in which languages are learned and taught is constantly evolving due to the increasing influence of technology and the need for students to acquire relevant life skills. The concept of a code literacy-based English language learning model has gained significant attention in recent years. As a result, the traditional English language learning model has been reimagined to incorporate elements of code literacy in order to prepare students for the demands of the twenty-first century. The understanding and application of coding languages in various contexts is referred to as code literacy. Coding as a language learning aid is one area that has seen tremendous progress.

This study investigated the integration of coding and language learning, resulting in the development of a learning model that effectively combined these two abilities to improve language competency and code literacy among Teacher Students (TSs) in English education department. The rationale for developing the learning model stemmed from the growing importance of both coding competence and English proficiency in a wide range of professional domains, particularly for future English teachers. The model's development was prompted by the demands of the modern world, the intersection of technology and language, and the need to prepare future English teachers for success in an increasingly competitive world.

The integration of technology and language learning has a long history. The concept of computer-assisted language learning (CALL) emerged in the 1960s, signaling a significant shift in language education. CALL emphasized the use of computers to aid in language learning, with a special emphasis on the use of software and interactive programs to improve language skills. The field evolved over time, resulting in the emergence of more sophisticated and interactive language learning platforms. The importance of coding skills has increased in the twenty-first century, with an increasing emphasis on the integration of coding in various educational domains, including language learning. The evolving technological landscape, as well as the recognition of coding as a fundamental skill for the future workforce, influenced this shift. As a result, the idea of a code literacy-based English language learning model gained traction, drawing on coding principles to improve language learning and proficiency (Singh, 2015).

Digital literacy is crucial for navigating the digital world and working in the modern workforce. The intersection of coding and English language proficiency is vital in industries like technology, data science, and business. Learning to code fosters problem-solving, critical thinking, and innovation. Integrating code literacy with English language learning can bridge the digital divide, providing access to digital skills for people from all backgrounds. This model was essential for training the next generation of learners to address 21st-century challenges and opportunities (Mbae, 2017).

The theory behind code-literacy-based English language learning (CLELL) was that by giving students a distinctive and interesting learning

environment, learning to code could enhance language proficiency. It has been demonstrated that using coding to learn a language offered a number of advantages. English instruction could be combined by engaging students with code, thereby fostering 21st-century skills and language development (Stevens and Verschoor, 2017). Coding could be utilized to develop interactive, engaging, and learner-specific language learning aids. In addition to language development, the utilization of coding as an educational tool could facilitate the development of critical thinking and problem-solving abilities among students, both of which were fundamental competencies necessary for achieving success across several domains of life. Additionally, learning to code could give students the chance to explore and interact with a wide range of subjects and issues, thereby enhancing their vocabulary and understanding (Barr, 2011).

Learning English through code literacy could be done in a variety of ways. One strategy was to utilize coding as a tool to make interactive games and quizzes for language learning. Another strategy was to educate students how to code so they may produce their own language learning content, such podcasts or videos. A third method involved using coding in order to provide students with guidance on effectively engaging and communicating with individuals through online forums and social media platforms (Code academy, 2021). Domenach (2022) explained that the integration of programming into English teaching was a transformative and influential concept aimed at enhancing the practicality and pertinence of both disciplines for primary school students' educational journeys.

This research was also inspired by an international English book for Elementary students –English Code which integrated English and coding

concepts (Morgan, 2020). Meanwhile, the learning model in the present research was the integration of English and coding concepts and practices. This research was significant because it addressed two crucial concerns in education: the requirement for digital literacy skills and the need to enhance language learning outcome. Students' learning experiences could be made more interactive and engaging with the incorporation of coding, which could enhance their language proficiency. Additionally, coding called for strong problem-solving and critical thinking abilities, two crucial aspects of digital literacy. Language teachers must consider the impact of the development of an English language learning model centered on code literacy. Teachers may foster student-centered learning and a more dynamic learning environment in the classroom by integrating coding into their lesson plans.

Coding has long been incorporated into language teaching and learning. In reality, since the 1960s, coding has been utilized as a method to improve language learning (Barron, 2013). However, it was now simpler to include coding into language teaching and learning to the development of new tools and platforms like Scratch and Code.org. These platforms provided students dynamic, entertaining coding exercises that may be customized to meet their language learning requirements. There were several possible advantages to creating an ELL paradigm based on code literacy. Firstly, learning to code could give students opportunities to apply language in real-world situations, which could improve their communication abilities. Students could gain problem-solving and critical thinking skills through coding activities, which were crucial for language development. Coding activities can help students improve their digital literacy,

which is becoming more and more crucial in today's digital environment. However, creating an ELL model based on code literacy is not without its share of difficulties. The effectiveness of this strategy has not been adequately studied. As a result, it is challenging to assess whether it is a useful method for assisting language learners in developing their skills. Secondly, there aren't enough tools and resources for teachers who want to incorporate coding into their methods of teaching languages. Thirdly, to employ coding effectively in their language teaching, teachers must have a solid comprehension of the subject. To establish the effectiveness of this strategy and to create tools and support for instructors who want to incorporate coding into their language teaching practice, more research is required.

The process of programming involves the formulation of sequential instructions that are intelligible to a computer. Young learners have the opportunity to incorporate communication, critical thinking, and problem solving through early coding (McLennan, 2017). The majority of students spend a significant amount of time playing online games, but few can create their own. Students who learn to code are more likely to become creators than mere consumers of the technology they employ. In addition, coding facilitates the growth of essential skills such as problem-solving, logic, and critical reasoning. In coding, there are frequently multiple ways addressing a problem require solutions that are simpler and more effective are frequently preferable. In addition, analyzing the processes of critical thinking and problem-solving can provide valuable language practice.

Lack of knowledge and awareness of the significance of coding in

language acquisition is one of this methodology's most significant flaws. Many language teachers and students may not prioritize the significance of coding in the context of language learning because they do not recognize its value or relevance. Another deficiency is the absence of appropriate teaching and learning tools for coding in the context of language acquisition. It is important to create materials that provide a clear and intelligible introduction to coding principles and terminology and are designed specifically for language learners. Furthermore, there are no measurement standards for code literacy in language education. Without a precise and standardized method to quantify code-literacy, it is difficult to evaluate the effectiveness and impact of integrating coding into language-learning programs (Garca-Sánchez, 2020).

Nowadays, computer applications have proliferated across all spheres of life. Those who can communicate in English have greater access to social and economic opportunities. However, they will have increasing possibilities by having skills in the use of computers or ICT especially in coding. If English teachers disregard coding skills, they have fewer opportunities. Thus, the integration of English skills and coding will be much more beneficial. The fact shows that the Teacher Educators (TEs) and Teacher Students (TSs) need to be provided with the skills. However, ICT is lacking, particularly on the use of coding programs. Thus, the situation needs overcoming and this research addressed the situation.

In the 21st century, teachers should help all students discover how to learn. Teachers are expected to inspire creativity, promote cooperation, demand and reward critical thought, and teach children how to communicate. The

education of the twenty-first century equips students with the knowledge, skills, and abilities necessary for success in the twenty-first century. Competences include numeracy literacy, scientific literacy, ICT literacy, financial literacy, and cultural and civic literacy (Hallerman.et.al, 2019; Voogt and Pareja, 2012).

With reference to Minister of Education and Culture policy (2022), Indonesia has targeted to have 9 million of digital talents in preparing for industry 4.0. However, 60% of teachers have limited skills in mastering information and technology (ICT). Indonesian English teachers' inadequate digital literacy has serious repercussions for both the quality of instruction and the nation's future economic growth. To ensure that students have access to the digital tools and learning opportunities they need to excel in the modern world, this issue must be resolved. The CLELL model is important to assist language teachers in finding a solution to this issue. This makes a gap in the realization of the policy.

The system of education for Indonesian students now includes computational thinking as one of the new competencies. This policy is motivated by the government's efforts to educate the next generation. Numerous nations have recognized the significance of computational thinking in education (Zahid, 2020) due to the accelerated growth of technology and computerization. The performance of Indonesian students in literacy, mathematics, and science ranks 74th, 73rd, and 71st, respectively (Schleicher, 2019).

Computational Thinking (CT) becomes as important as reading, arithmetic, and writing. Indonesian Minister of Education and Culture Nadiem Makarim plans to incorporate CT into the curriculum to improve its scores and rankings in PISA. This will equip the younger generation with relevant

competencies and skills, preparing them to become qualified individuals for global competition. Integrating CT into the curriculum can help Indonesia improve its scores and rankings in these areas, ultimately preparing its students for the future (Zahid, 2020).

This research was designed in a developmental research which tried to weave language learning with coding activities-especially block-based coding. This model tried to see the feasibility of the two frameworks – English and code in promoting computational thinking skills and at the same time improving language skills, especially reading comprehension. The research was expected to give a new lens on the perspective of language teaching since the modern goals of language teaching is considered as a tool for learning to understand other disciplines rather than an end in itself. This research also evaluated the effectiveness of the model in improving students' language proficiency and coding skills.

Scratch is a practical coding program that enables users to create and share interactive stories, games, and animations. It is popular among children and young adults to enhance their computational thinking and creativity. With over 65 million registered users globally, Scratch has gained popularity in Indonesia. The majority of users are aged 10-14, with those aged 15-18 following. The service has a global audience and is accessible to users from various backgrounds, with the United States, Brazil, and Russia having the highest concentrations while Indonesia ranks 12th with over 1.5 million registered users in August 2021.

Using Scratch, Daher, Nimer, Otman, and Juhaina (2020) investigated a

method for improving the metacognitive resolution of math-based programming issues. Sarasa (2019) characterized Scratch as a tool that can aid in the acquisition of second languages by teachers and students. In the meantime, in Indonesia, researchers examined the impact of students' Scratch-created digital narrative designs on their reflective problem-solving abilities. Bahar and Budaka (2021) investigated the effect ofincorporating Scratch into language teaching for students on linguistic and cognitive skills. In addition, Junardin (2021) explained that students taught using scratch learning media achieve substantially better learning outcomes than students taught using conventional models. However, Dohn (2019) showed that the findings reveal a minor, but substantial, detrimental influence on students' overall interest in coding and mathematics. Future English teachers should familiarize themselves with the various programs and websites available for teaching and learning English (Azmina, 2019).

Some studies have addressed the use of coding programs in education, however, to date, few researchers have explored the use of coding and computing concepts in conjunction with English language teaching and learning. As a result, there was still a need for additional research on how English instruction can be combined with encouraging students to work with code in order to instill aspects of 21st-century skills while fostering language development.

Even though there had been programming experiences in K-12 institutions over the past fifty years, Papert's vision is now becoming a reality. Coding now incorporates expressiveness, collaboration, and creativity, in addition to an aptitude for high-demand jobs. Code literacy is a fundamental

competency for various types of 21st-century workers and all students should have the opportunity to develop this type of literacy (Orsini, 2013). Thus, coding literacy is an important sort of literacy that is relevant to classroom instruction.

According to a research by Bers, González-González, and Armas-Torres (2019), teachers can confidently and effectively incorporate coding and computational thinking into their present teaching practices by making connections between the concepts and social studies, music, and art. In addition, they discovered that students, regardless of the subject matter, acquired good behaviors in learning environments that supported coding. Student who learned to code were found to be better able to: understand abstract mathematical concepts in more concrete ways; dynamically model abstract mathematical concepts; and increase their confidence in their own ability and agency as learners. This was particularly true of mathematics (Gadanidis, 2017).

Rushkoff (2012) observed that while over 2 billion people, including teachers and students, use the site superficially to share the minutiae of their lives with friends, the deeper purpose of Facebook, for instance, is to sell data points on those minutiae to high bidders and that an understanding of the deeper mechanics of how the digital world works is an essential literacy skill for the twenty-first century. Individuals can regain control of the world through the process of learning to code. Those who learn to code will be able to program, as opposed to being programmed.

Learning human languages is similarly empowering, as evidenced by the

number of individuals willing to hire instructors to assist them do so. Learning to code within the context of language development ought to be doubly empowering, as the development of either, even on its own, or, ideally, both, facilitates critical thinking and leverages future opportunities. As Rushkin argued, code-literate students stop to accept the applications and websites they use at face value and instead engage with them critically and with intent.

In Italy, for instance, coding education at the basic and secondary levels is developing as a serious educational concern. There has been a surge in the research on ways to integrate information technology into lower secondary schools. The report describes an experimental survey conducted during a coding intense program. The findings are positive since they show that coding is both extremely fascinating and inspiring for students (Banzato, 2017). In Japan, teaching English through programming was previously missing from the primary school curriculum. A range of approaches is advocated to teaching programming and English to improve both primary school programming and English language instruction (Kazahaya, 2017). CLIL concepts are being incorporated into the activities to make learners more interesting, participatory, and relevant. In Indonesia, Bambang (2019) stated that the coding ability of human resources is one of the big problems facing the digital era in Indonesia.

Because of the rapid progress of technology, there is a greater demand for those who have both good language abilities and technological expertise. In today's globalized world, knowledge in the English language is a great advantage, and coding has become a necessary skill in a variety of businesses. However,

coding instruction in language learning programs is still a much undeveloped area of research. The current research needs was in the development of a code-literacy-based English language learning model that could successfully merge coding education and language learning. While studies have been conducted to evaluate the effectiveness of introducing coding into language learning, they have generally focused on the influence of coding instruction on language acquisition rather than the construction of a comprehensive model that blends the two (Bahar et. al, 2021; Murtafi'ah and Nur, 2019).

Furthermore, the majority of research on this problem has focused on computer science education rather than language learning (Pratiwi, Ahmat, and Parulian, 2022; Ozdemir, Rabia, Aleyna, Patmanur, and Kubra 2021; Budak, Aynur, and Arzu, 2020; Lonati, 2020; Daher, Nimer, Otman, and Juhaina, 2020; Endah, Eko, Nurdin, Adi, and Kabul, 2020; Dohn, 2019; Cabezuelo, 2019; Permatasari, 2018; Prayitno and Johan, 2017). This underscores the need for additional research into the potential of incorporating coding into language learning programs and developing a model that can be utilized to effectively teach both abilities.

There hasn't been much research on code literacy-based language learning, and there isn't much concrete data to back up its effectiveness. Although several researches had looked into how coding can be incorporated into language learning, these studies have tended to concentrate on particular facets of language learning, including vocabulary learning or grammar (Xu, Ashlynn, and Lauren 2019; Sourani and Ihmaid, 2019). More thorough research is required to examine the overall effects of code literacy on language development. More research is

required to examine the effectiveness of various code integration strategies in ELT, which is another area of research need. Coding can be used in ELT in a variety of ways, such as by using it as a project-based learning tool or by embedding coding exercises into language courses. Nevertheless, studies comparing the efficiency of various methodologies in terms of language learning outcomes and students' motivation and involvement are rare (Banzato, 2017; Soykan, 2018).

Individuals must be proficient in both English language skills and coding literacy in today's swiftly developing technological environment. However, many traditional language learning model fails to include the necessary coding skills in their curricula, resulting in inadequate preparation for the demands of the modern workforce (Bureau, 2020). Lack of collaboration between language learning and computer programming has resulted in a theoretical gap in code literacy-based English language learning models. To bridge this gap, an English language learning paradigm based on code literacy should be developed, and coding exercises should be integrated into language learning curricula. This would provide students with the technical skills necessary to effectively integrate coding and English language learning (Saito, 2019).

It's critical to evaluate the effect this method has on student progress as the use of coding in language learning gains popularity. It was found that students who used a computer-based language learning program that included coding tasks outperformed those who utilized conventional language learning methods on language competency exams. The researchers hypothesized that the coding exercises offered students a motivating and dynamic learning experience that

assisted in enhancing their language proficiency (Gee, 2003). In another study, it was discovered that students who took part in a coding-based English language learning program significantly improved their language proficiency as well as their capacity for critical thought and problem-solving. The researchers hypothesized that by giving students the chance to examine and resolve actual problems, coding activities assisted in the development of these crucial abilities (Resnick, 2009).

Previous researches had demonstrated the significance of digital literacy. For example, Hammarström (2021) investigated how Swedish upper secondary schools incorporate digital literacy into the English course. According to the findings of the research, teaching digital literacy in accordance with Swedish guiding texts had the greatest influence on students. Other studies have also found that e-learning and online courses can provide students with an effective learning environment. It has also been discovered that mobile devices can enhance literacy opportunities for students in low-middle income and developing nations (LMI) (Selmo, 2020; Oakley, 2019).

Gammification of learning is gaining popularity as a means of making language learning more engaging and relevant to the younger generation. It is associated with the use of coding in English language instruction. Gamification is the use of game metaphors, features, and concepts in non-game contexts. It seeks to boost student enthusiasm and commitment while also influencing their behavior during the learning process (Marczewski, 2013). According to Zichermann and Cunningham (2011), gamification may simply be described as ensuring user interaction and problem solving. Code literacy in the context of

coding and gamification may be attained through integrated learning in the form of code-literacy-based English language learning. According to Rushkoff (2012), code literacy is significant for involvement in today's digital environment. Future English teachers should therefore be conversant with a variety of programs and websites that can be used to teach and learn English (Azmina, 2019).

Tynker is a popular programming platform that teaches game design, web design, animation, and robotics to children. Moura (2020) studied the impact of gamification on learning, focusing on language learning, STEM, and history through applications like Math Learner and Tynker. With over 60 million users globally, Tynker is one of the most widely used platforms, with the majority being from China, India, and the United States. The platform's popularity is due to its effectiveness in fostering computational thinking abilities. In Indonesia, over 70% of primary schools use Tynker as an extracurricular activity, while only 40% of secondary schools use it as part of their curriculum.

Bahar, et al. (2021) studied the impact of Scratch on children's English language and cognitive development, examining its effects on language and cognitive skills. Ozdemir et al. (2021) investigated the perspectives of mathematics educators on Scratch-created digital games. They found the games enjoyable, understandable, and reinforcing, with some teachers finding them useful in the classroom. Budak et al. (2021) found that Scratch programming positively affects students' problem-solving reflective thinking abilities in Archaeology, Philosophy, and History, particularly in the inquiring sub-dimension. Future studies may use project-based, observational applications, and achievement examinations to assess these skills.

Cabezuelo (2019) highlights Scratch as a visual programming language that can be used to teach second languages and other fields of expertise. Permatasari (2018) conducted a study on the educational utilization of Scratch, focusing on its implementation to improve learning outcomes and student motivation in Basic Programming subjects. The results showed that Scratch can significantly enhance both learning outcomes and student motivation in this field.

Sourani et al (2019) and Pessoa et al (2016) conducted studies on the use of Scratch in English language instruction. Sourani et al's study found that Scratch improved sixth-grade students' vocabulary, retention, and self-effectiveness. Pessoa et al's research showed that Scratch may have influenced students' academic and linguistic development, but conclusive results are difficult to draw due to the need for further data review.

Bebras is a multinational program aimed at promoting computational thinking and problem-solving skills in young people. It offers the Bebras Challenge, an interactive online problem set in over 60 countries. In 2019, over 5 million students participated, particularly in developing nations with limited computer science instruction. The initiative has attracted nearly 50,000 students from 15 countries in Africa, demonstrating its ability to connect with underserved groups and advance computer science education. The Bebras Challenge is available in multiple languages and is open to students aged 6-19.

Pratiwi et al (2022) conducted training on computational thinking and Bebras tasks for elementary and junior high school teachers, aiming to improve higher-order thinking skills using software engineering methodologies. Lonati

(2020) explored how teachers can use Bebras material and incorporate it into their teaching practice. A qualitative analysis of teaching projects created by Italian teachers revealed that teachers can build on challenges but also expose crucial issues. Daher et al. (2020) conducted research on meta-cognitive solving of Mathematics-based programming problems with Scratch, finding that negotiation processes helped develop prospective teachers' metacognitive processes.

Bebras is a global initiative promoting computational thinking among elementary school students and teachers. The program has been found to increase understanding of computational and informatics mindset among students, particularly in grade four and five. Dohn (2019) emphasized the importance of coding computer programs for literacy. A case study in two sixth-grade courses found that Scratch coding, used as part of six math lectures, had a minor but significant negative impact on students' interest in coding and mathematics. The study highlights the need for design concepts that encourage tinkering and the need for more emphasis on design knowledge in teacher professional development.

In 21st-century education, effective teaching requires teachers to be decision makers, designers, managers, facilitators, and motivators, as well as creative, reflective, technology-literate, self-confident, and communicative. They should integrate language learning and technology, and foster discovery through exploration and creation. Research has focused on improving teachers' digital literacy, but not digital literacy with Code.

The English Education study program at Universitas Negeri Medan

focuses on transferring knowledge and skills in language teaching. Teacher educators aim to equip students with necessary skills for the 21st and 22nd centuries. Students must possess professional and pedagogical skills, integrate learning, and be qualified for fieldwork. Development and upgrading are necessary for improved education quality and society's need. Based on the observations, code literacy activities were not covered in the EYL syllabus or module. This issue inspired the researcher to develop learning model based on code literacy and problem-based learning. This study also evaluated the feasibility of incorporating coding programs into English language learning. Theoretically, research into STEAM (Science, Technology, Engineering, Arts, and Mathematics) implementation in English language learning remained limited. As a result, more comprehensive research on this topic was still required.

Attempts to improve students' 21st-century learning skills could be seen in a variety of curriculum subjects, including ICT in language learning (Semester I), Digital Literacy (Semester V), and Digital Educational Content Creator. This study focused on students enrolled in English for Young Learners (EYL) classes during the fifth semester. To promote students' critical thinking, problem-solving, and computational thinking, coding or computer programming must be incorporated into English language instruction. To achieve this goal, creating a code-literacy English language learning model was critical in order to meet the needs of 21st century learners.

Contextual Teaching and Learning (CTL) is crucial in 21st-century education for authentic learning. It helps teachers relate information to real-world scenarios, encouraging students to connect their knowledge with everyday life

(Triyoga, 2010). CTL concepts include self-regulation, self-critical thinking, teamwork, critical and creative thinking, supporting the individual, and striving for excellence (Johnson, 2002). Research has shown that using customization and contextual teaching strategies can enhance English competency among students (Elandeef, 2021).

Problem-Based Learning (PBL) is an instructional model that uses concepts to solve real-world problems, improving critical thinking, problem-solving, and communication abilities. It encourages collaboration, research material discovery, and lifelong learning (Duch, 2001). Studies have shown that it improves persuasive writing, speaking skills, and writing abilities (Learni, 2021; Rahmadhani, 2022). Most students prefer PBL to other learning methods, and Astuti (2020) discovered that PBL significantly improved students' writing skills.

This research evaluated students' needs for learning English through coding activities and developed a Code Literacy-based English language learning model. The model book, teacher's guide, student's guide, and activity book were designed. The products were validated by experts and implemented to assess its effectiveness. The next chapter discussed literature reviews related to this research.

1.2 The Identification of Problems

Based on the above research context, several issues can be identified, including:

1. English language education goes beyond linguistic proficiency, emphasizing on developing 21st-century skills such as critical thinking, creativity, communication, collaboration, and cultural competence, but integrated

language teaching and learning with coding was limited.

2. Coding is essential for digital literacy, but it has not been integrated into English learning, requiring students to master English skills. The widespread integration of digital tools and resources in English language classrooms promotes digital literacy and enhancing language skills through technology.

3. To fulfill the demand of 21st century education, educators do not only master language skills but they must also have digital literacy, especially digital literacy with coding. This can be achieved through integrating coding in English teaching and learning.

4. Online communication tools enable students to engage in cultural exchange and global collaborations, establishing connections with peers from various countries for the purpose of enhancing language proficiency.

5. The syllabus used in English Young Learners class at Universitas Negeri Medan was still single disciplinary frameworks which was inappropriate to support and guide language learning in 21st century.

6. Coding programs such as, Scratch, Tynker, and Bebras Tasks could be used as learning media in English learning, but integration between code literacy and problem-based English language learning was still limited.

7. The Integration of Gamification and Immersive Learning is necessary to augment student engagement and motivation in the context of language learning, and integrate gamified components and immersive experiences.

8. Problem-based learning emphasizes in which learners engage in language-

intensive, real-world initiatives that cultivate critical thinking skills and practical application of language.

1.3 The Focus of the Research

The goal of this study was to develop a learning model that promoted dual proficiency, improved practical application, catered to diverse learners, aligned with industry demand, and maximized learning synergy. The model, which was based on problem-based learning, demonstrated how block-based coding using Tynker and Scratch, as well as computing concepts using Bebras Tasks, were integrated into English teaching and learning. The study focused on Teacher Students (TSs) taken English for Young Learners (EYL) classes at the English and Literature Department of Universitas Negeri Medan. Reading comprehension was the primary language skill. The data were limited to TSs' needs in terms of incorporating coding into English language teaching, particularly in EYL classes.

1.4 The Problems of the Research

Based on the above-mentioned background, problems identification, and focus of the research, the problems of research were formulated as follows: ”How was Code Literacy-Based English Language Learning Model developed to cope with the Teacher Students’ needs?”

Furthermore, to conduct the research, the problems were formulated into sub problems.

1. What were the Teacher Students’ needs in Code Literacy-based English Language Learning (CLELL) model?
2. How was Code Literacy-based English Language Learning (CLELL)

model developed?

3. How was the validity level of Code Literacy-based English Language Learning (CLELL) model?
4. How was the effectiveness level of Code Literacy-based English LanguageLearning (CLELL)?

1.5 The Objectives of the Research

Based on the problem of the research above, the objectives of the research were to develop code literacy-Based English Language Learning Model tocope with the teacher students' needs. Furthermore, the objectives of sub problems of the research were as the following.

1. To evaluate Teacher Students' needs in Code-Literacy-based (CLELL) English Language Learning model;
2. To develop Code Literacy-based English Language Learning (CLELL) model;
3. To evaluate the validity level of Code Literacy-based English LanguageLearning (CLELL) model;
4. To evaluate the effectiveness level of Code-Literacy-based English Language Learning (CLELL) model.

1.6 The Significances of the Research

This research explored the cognitive benefits of integrating language and coding to improve problem-solving skills. It highlighted the practical applications of the model in a tech-centric world, fostering dual proficiency essential for diverse career opportunities and navigating technology-driven landscapes. The development of a Code literacy-based English language learning

model has several theoretical significances, including:

1. The integration of code literacy into language learning has been proven to enhance cognitive functions, promote logical thinking, and improve problem-solving abilities.
2. Understanding both coding and language aided in the development of metacognitive abilities, enabling students to assess their learning strategies and processes.
3. The integration of language learning and coding could bridge disciplinary boundaries, fostering a comprehensive understanding of the interrelationships among various subjects.
4. The integration of coding into the language learning process fostered creativity by encouraging students to devise novel methods of problem-solving and idea expression that utilize computational in addition to linguistic techniques.
5. The integration of coding activities accommodated a wide range of learning styles, including auditory, kinesthetic, and visual learners, by means of interactive programming exercises.
6. The model facilitated the advancement of digital literacy by equipping learners with knowledge of the technological environment, thereby equipping them to proficiently navigate and make valuable contributions in the digital age.
7. The integration of code literacy into language learning promoted the development of global citizenship skills through an awareness of the significance of technology in facilitating intercultural communication,

cooperation, and comprehension.

8. Providing students with code literacy ensuring that they were adequately equipped to navigate forthcoming labor markets, where technological expertise is progressively emerging as a fundamental skill.

Several practical significances of the learning model were:

1. The incorporation of code literacy supported individuals to navigate technology-centric labor markets, thereby augmenting their prospects of securing employment in diverse sectors.
2. The practical coding skills that learners acquired via the model were in line with the requirements of the industry, enhancing their ability to address practical challenges in professional environments.
3. The model promoted innovation acceleration through enabling students to creatively employ their coding skills, thereby contributing to the development of novel solutions and approaches in their respective domains particularly in language teaching.
4. Individuals acquired the capacity to articulate thoughts and ideas regarding technology in a manner that fosters collaboration and enhanced comprehension in professional environments where technology is pervasive.
5. Proficient code literacy empowered individuals to investigate entrepreneurial endeavors by allowing them to utilize their coding and language abilities in the creation of digital products or services.
6. Through the provision of practical coding experiences, the model developed learners' practical problem-solving capabilities, allowing them to effectively respond to changing challenges in dynamic work environments.

7. By engaging in hands-on coding exercises, comprehending ethical considerations, and employing technology responsibly, students cultivated responsible digital citizenship.
8. The model fostered a climate of ongoing learning by having students participate in hands-on coding tasks that motivate investigation and adjustment to emerging technologies.

1.7 Terms Related to the Research

a. **Code literacy** is defined as Code literacy involves understanding and generating basic computer programming code, enabling individuals to contribute to a technologically advanced and digital society through understanding programming principles, syntax, and logic.

b. **Tynker** is an instructional platform that uses interactive games and activities to teach coding and programming skills, aiming to make it enjoyable and accessible for learners.

c. **Scratch** is an online community and visual programming language founded by MIT, allows users to create interactive narratives, games, and animations using code blocks, making coding accessible to novices.

d. **Bebras** is an international initiative that promotes computational reasoning and problem-solving skills through a yearly competition for computer science and logical reasoning-related tasks.

e. **Teacher Student** is instructor teaches knowledge, provides guidance, and offers support to become a qualified educator.

f. **Teacher Educator** is a professional who assists in educating and training people to become teachers.