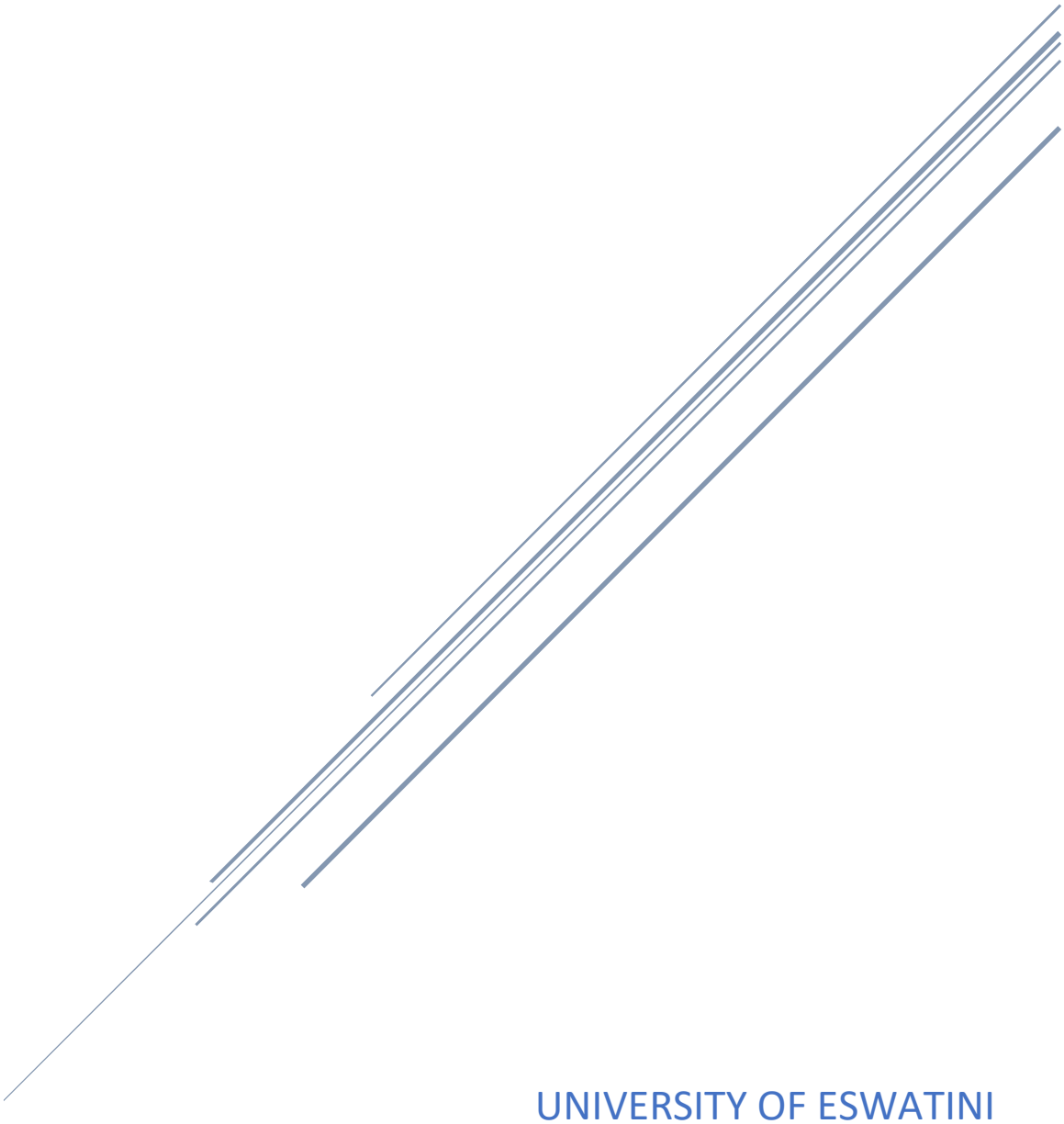
PROJECT PLAN

GROUP 2/ TEAM2



UNIVERSITY OF ESWATINI

PRACTICES IN SOFTWARE ENGINEERING

The University of Eswatini holds the title of being the largest tertiary institution in the country, eSwatini.
Disappointedly, the institution still uses traditional, or rather primitive methods for most of its operations. An example of such, in which my team and I are going to focus on, is the current use of the billing system, where the school's warden uses a spreadsheet for allocating students to their respective rooms and has to manually keep track of the number of days/duration for students. This, without a doubt, is outdated in the modern world since the method has proved to be erroneous. Students residing on camp are prone to paying extra money by simple miscalculations from the warden. Also, it becomes complex to keep track in cases where the University sends its students home in cases of emergency.

The team, therefore proposes use of a fairly modern system, which will be beneficial to both the school's warden and students.

The aims of this proposed modern billing system are:

1. To enable the warden to easily manage hostel rooms and be notified on already accommodated rooms.
2. To enable students to have remote access to the billing system, therefore adding benefits of both parties (warden and students) being able to keep track of the total duration for students. This minimizes error
3. To safely store data in the cloud for future access.
4. To enable students to have remote access to application for residency The Objectives:

1. To develop a user-friendly environment. This will be accomplished by designing the system as simple as possible, and should satisfy the user needs.
2. To test and run this software before making it public to its users (warden and students).

The team consists of:

- Mpumelelo Dlamini (team leader) 202002852
- Luyanda Zwane 202002531
- Bathandwa Mavuso 202003846
- Mandisa Nkumane 202004102
- Sakhile Dlamini 202003084

Summary

In a nutshell, the above people as a team, have a mutual goal of designing a software billing system for residency at the prestigious University of Eswatini. And as highlighted above, the aim is to introduce a billing system that syncs, or rather conducive for the modern world status. Further project details, including the estimated time the project will take to complete, the roles of assigned for the team members, etc., will be revealed in the below sections.

User Involvement

User 1: Warden

- For controlling residency intake, the warden needs the total available rooms for each block/compartment
- For tracking duration stay, the warden needs student's profiles that will have details like the student's name, days to be stayed, the payment method proposed by the student/s etc.
- For allocating special rooms to students with conditions/ disabilities, the warden needs a forum section where those affected students would have highlighted their conditions.

User 2: Students

- For access to the system, they need authentication to be provided by the warden. A login form, for this case will be used.
- Before applying for residency, they need to get notified on the availability of rooms when a new semester commences.
- They need clarity on how the billing system works, with regard to the rent per day
- Finally, students need to be able to successfully reserve a room for themselves when a new semester commences, and be able to opt out whenever they want.

Risks

1. Code: We may find it challenging to develop the code that serves the project aim. Even after development, the developed code could just not work.
2. Personal: One of the team members could struggle to deliver their part of the work.
3. Tight Schedule: Due to the minimal time allocated for this project, this could result in delayed project delivery.
4. Emergency closure of the university: On uncalled for occasions, the university may be compelled to send its student's home. This will have a negative impact on the team's communication, together with the time allocation.
5. Low productivity: Backlog of work by team members is one of the reasons that could lead to low productivity.

Standards, guidelines and procedures

Listed below are the team's standards, guidelines, together with procedures:

1. Standards and Guidelines

A. The team

- Playing an equal role in the project through fair allocation of tasks
- The team as a whole should instill a positive attitude towards working on the project at hand, and towards each other. This makes progress possible.
- Promptly relay all interpersonal concerns to the team leader
- The team leader should be rational at all times, and avoid biasness
- Team members should offer help towards another. This could come in the form of; offering ideas, volunteering for tasks, giving out constructive criticisms with suggestions for improvement, etc.
- Be honest with a team member who is not optimizing his/her work skill/s, and members should accept responsibility if such claims are made upon them.
- Delivering the work assigned on time
- Above all, respect should be a priority in the team.

NB: The team leader is obligated to ensure the above conditions are followed. Also, the team's leader has the power to dismiss team members who do not comply with the set guidelines.

B. The project

- Keeping a consistent coding style throughout the project
- Clearly outlining the layout/structure of the code to enable other people to work and modify the code with ease.
- All written codes should be: secure, reliable, testable and maintainable. • Avoid duplicating the functionality of a built-in library
- Avoid writing code that guesses functionality.
- Confronting issues directly and promptly
- Above all, the team should have dedication to a quality software

2. Procedures

A. Project meeting

- Be punctual for all team meetings, whether held virtually or live. A strict protocol of arriving 5 minutes prior should be followed.
- An assigned member should create and disseminate minutes after every team meeting

- Team members should concentrate and actively participate throughout the meeting duration.
- Do not disturb the meeting proceedings. That is, avoid informal talks in an ongoing team meeting.
- Be receptive and patient with alternative viewpoints. Remember, respect is a priority in the team's guidelines
- Team members should report prior if they will be unavailable for a meeting.

B. Team Location

- University of Eswatini

C. Project Delivery

- Seek the supervisor's feedback before handing in all deliverables
- Set deadlines for each deliverable in advance of due date to allow for collaborative revisions.
- In cases of delay, inform the team leader

D. Training Requirements of Personnel

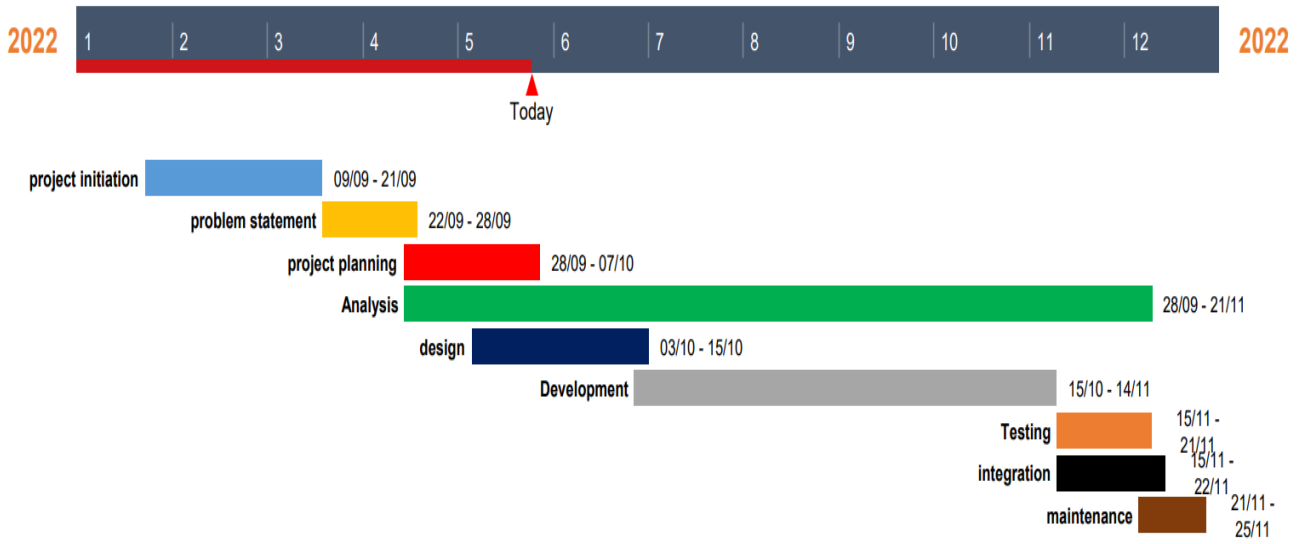
- Online platforms. This is to maximize the little time available for the project.

E. Project Decisions

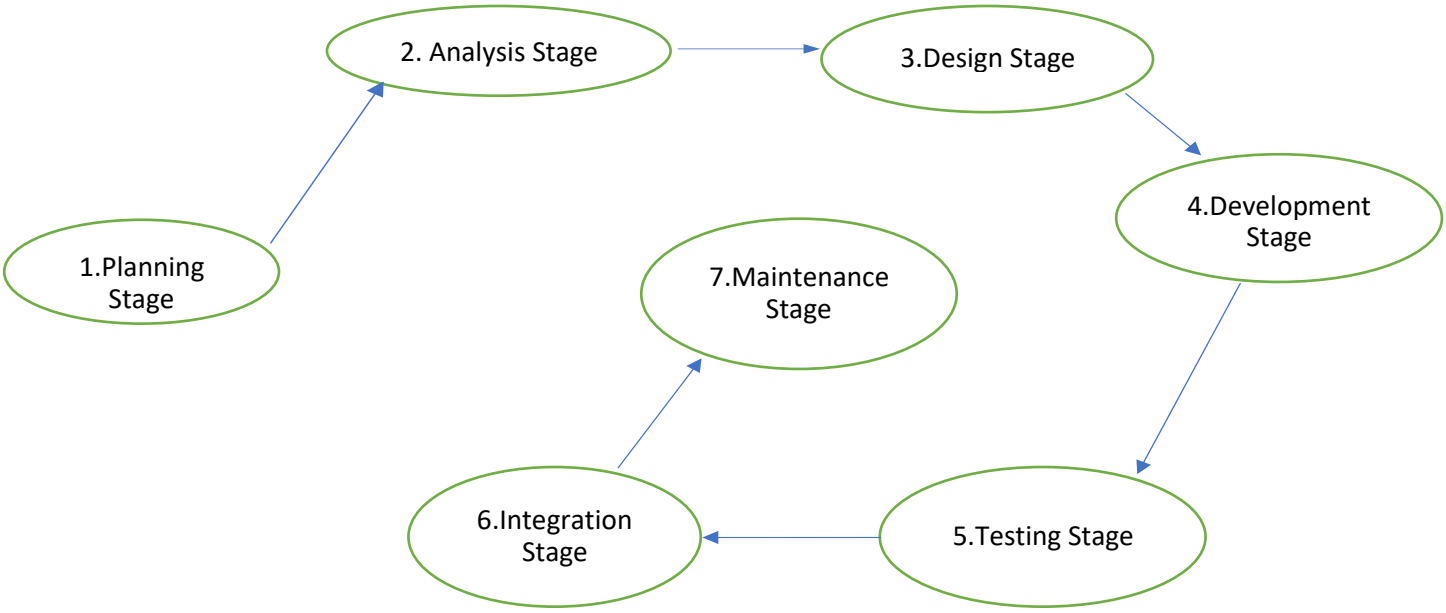
- Decisions should be made upon general agreements across all team members

Organization of the Project

1. Dlamini Mpumelelo – Manager
2. Zwane Luyanda – Designer
3. Mavuso Bathandwa – Developer
4. Nkumane Mandisa - Tester
5. Dlamini Sakhile – Analyst



PROJECT PHASES



Requirements Analysis and Design

- JSP method which goes along with the pseudo code, will used. The tool required for this is a personal computer.

Implementation

- Personal Computers

Testing

- Xamp will be used, since it is user-friendly, and its open source.

Resources:

- Personal Computers

Quality Assurance

The following are guarantees for the final product:

1. Is the software secure? YES

2. Is the software system accessible and usable? YES
3. Is the software efficient and meets the needs of both warden and students? YES

Changes

The following key steps are to be implemented in the case changes occur:

1. Identify the change:
   This includes: identifying the type of change, describing the change in detail using a set of features, noting the scope of the change, together with the project identifier. This makes the change easier to understand to everyone involved in the project.

2. Access the impact;
   This includes: evaluating all risks the change poses on the software, the impact on the team working on it, together with those who will be using the software(warden and students). In a nutshell, evaluate both the technical and general aspects of the change. Lastly, take into account the budget for those changes, the deadline, and the resources needed.

3. Decide on introducing the change:
   It is crucial to know who will make the change decision and how the change will be introduced. Such should be discussed from the very beginning so to make the members aware of handling change requests when they do arise. Upon making the change decision, ensure that the software developments team is well informed about every stage that is to be implemented.

4. Plan the introduction of the change:
   Modify the original project plan if needed, including other components which may be deleted or introduced. Include information on any new deadlines or backlog that the whole organization may face when the change is introduced.